201
1-2-80

DR. 461

LA-7396-M, Revised

Manual

MASTER

MCNP—A General Monte Carlo Code for Neutron and Photon Transport

University of California

LOS ALAMOS SCIENTIFIC LABORATORY
Post Office Box 1663 Los Alamos, New Mexico 87545

This report was not edited by the Technical Information staff.

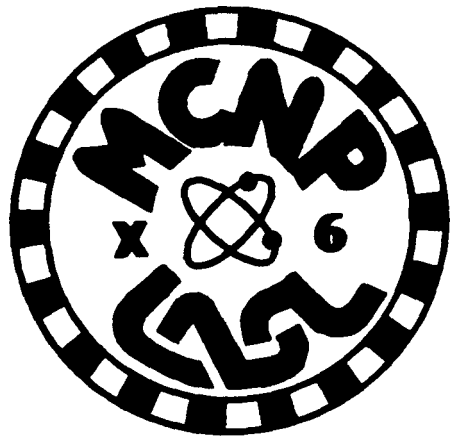
Work supported by the US Department of Energy, Office of Military Application.

LA-7396-M, Revised
Manual

UC-32
Issued: November 1979

MCNP—A General Monte Carlo Code for Neutron and Photon Transport

LASL Group X-6



FOREWORD

This manual is written as a practical guide for the use of our general-purpose Monte Carlo code MCNP. The intent is that the second chapter describe the mathematics, physics, and Monte Carlo simulation found in MCNP. However, this discussion is not meant to be exhaustive - details of the particular techniques and of the Monte Carlo method itself will have to be found elsewhere. The third chapter shows the user how to prepare input for the code. The fourth chapter contains several examples, and finally the fifth chapter explains the output. The appendices show how to use MCNP on a particular computer system at the Los Alamos Scientific Laboratory and also give details about some of the code internals that those who wish to modify the code may find useful.

Neither the code nor the manual is static. The code is changed from time to time as the need arises (about once a year), and the manual is changed to reflect the latest version of the code. This particular manual refers to Version 2 of MCNP that was released on September 26, 1979.

MCNP and this manual are the product of the combined effort of the people in Group X-6 of the Theoretical Applications Division (X Division) at the Los Alamos Scientific Laboratory.

Except for the figures, this manual was prepared entirely by the TRIX Report Editor and the REDPP post processor routines available through the LTSS operating system on the LASL CDC-7600 computers. The master for the manual is 35-mm film produced by the FR80 film recorders.

W. L. Thompson
X-6 Group Leader
505-667-4189

CONTENTS

APPENDICES

MCNP - A General Monte Carlo Code
for Neutron and Photon Transport

LASL Group X-6

ABSTRACT

The general-purpose Monte Carlo code MCNP can be used for neutron, photon, or coupled neutron-photon transport, including the capability to calculate eigenvalues for critical systems. The code treats an arbitrary three-dimensional configuration of materials in geometric cells bounded by first- and second-degree surfaces and some special fourth-degree surfaces (elliptical tori).

Pointwise cross-section data are used. For neutrons, all reactions given in a particular cross-section evaluation (such as ENDF/B-IV) are accounted for. Thermal neutrons are described by both the free gas and $S(\alpha,\beta)$ models. For photons, the code takes account of incoherent and coherent scattering, the possibility of fluorescent emission following photoelectric absorption, and absorption in pair production with local emission of annihilation radiation.

MCNP includes an elaborate, interactive plotting capability that allows the user to view his input geometry to help check for setup errors. Provisions are also made to translate and/or rotate surfaces from one coordinate system to another. Cell volumes and surface areas are automatically calculated for use by the tallies.

Standard features which are available to improve computational efficiency include geometry splitting and Russian roulette, weight cutoff with Russian roulette, correlated sampling, analog capture or capture by weight reduction, the exponential transformation, energy splitting, forced collisions in designated cells, flux estimates at point or ring detectors, deterministically transporting pseudo-particles to designated regions, track-length estimators, source biasing, and several parameter cutoffs.

Extensive summary information is provided to help the user better understand the physics and Monte Carlo simulation of his problem. The standard, user-defined output of MCNP includes two-way current as a function of direction across any set of surfaces or surface segments in the problem. Flux across any set of surfaces or surface segments is available. Similarly, the flux at designated detectors and the average flux in any cell (track length per unit volume) are standard tallies. Reactions such as fissions, tritium production, absorptions, or any product of the flux times any standard ENDF reaction cross sections plus several nonstandard ones may be obtained in any cell, at a surface, or at a point. The heating tallies give the energy deposition in designated cells. In addition, particles may be flagged when they cross specified surfaces or enter designated cells, and the contributions of these flagged particles to the tallies are listed separately. The user is allowed to modify any of the standard tallies almost any way desired. All quantities tallied also have their relative errors calculated. All tallies are a function of time and energy as defined by the user.

CHAPTER 1
A SUMMARY OF FEATURES

If you are reading this Manual for the first time with the intent to set up and run a job with MCNP as soon as possible, see the Quick and Easy section of Appendix A.

MCNP is a general-purpose, continuous-energy, generalized-geometry, time-dependent, coupled neutron-photon Monte Carlo transport code. It may be used in any of three modes: neutron transport only, combined neutron-photon transport, or photon transport only. The capability to calculate eigenvalues for critical systems is also a standard feature of MCNP.

The code compiles under the CDC FTN compiler that has been slightly modified at the Los Alamos Scientific Laboratory (LASL), is largely compatible with the FORTRAN 77 standard except for a few non-standard FTN features, and is developed and maintained by the LASL Group X-6 on the LASL CDC-7600 and CRAY-1 computers. MCNP has only 12697 lines of coding (this is with the COMMON blocks listed only once and not in every subroutine). Group X-6 also maintains MCNP on the Magnetic Fusion Energy (MFE) computers at the Lawrence Livermore Laboratory.

An attempt has been made to make MCNP as system independent as possible to enhance its portability. System dependencies that could not be avoided have been segregated as much as possible into system-dependent subroutines that will undoubtedly need to be replaced at other computer facilities. This manual is written for the CDC-7600 version of MCNP where the operating system is the Livermore Time Sharing System (LTSS).

The various features and capabilities of MCNP are summarized in the rest of this chapter. More detail concerning each topic is available in later chapters or appendices.

I. GEOMETRY

The geometry of MCNP treats an arbitrary three-dimensional configuration of arbitrarily defined materials (using up to 40 different isotopes chosen from the MCNP cross-section libraries) in geometric cells bounded by first- and second-degree surfaces and some special fourth-degree surfaces (elliptical tori). The cells are defined by the intersections and unions of the regions bounded by the surfaces.

MCNP does not have combinatorial geometry as it is commonly denoted. Rather than combining (i.e., taking the union of) several pre-defined geometrical bodies as in combinatorial geometry, MCNP gives the user the added flexibility to define his own geometrical bodies from all the available surfaces and then to combine them with a union operator.

Surfaces are defined by supplying coefficients to the analytic surface equations or for certain types of surfaces by supplying known points on the surfaces.

The code does extensive internal checking to find input errors. In addition, an elaborate plotting capability is available to aid the user.

II. NUCLEAR DATA AND REACTIONS

Pointwise cross-section data in considerable detail are used in MCNP. The data are tabulated in the MCNP cross-section libraries on an energy grid that is tailored to each isotope. Linear interpolation is used between energy points with a few hundred to several thousand points typically required. Cross sections are added at a sufficient number of points to insure that the linear interpolation constraint reproduces the original cross-section tabulation within a specified tolerance of a few percent (in fact, usually 0.1 to 0.5%). Resonance parameters, if they are given, are processed at several temperatures (but only at one temperature for a given cross-section evaluation) and added to the pointwise cross sections. Furthermore, the energies at which the cross sections are tabulated are shifted so that all reactions are given using the same energy grid – the grid on which the total cross section is tabulated. The total photon production cross section and neutron heating numbers are given on that same energy mesh.

MCNP is strictly a neutral particle transport code; any energy that would be carried by charged particles is deposited locally.

A. Neutrons

All reactions given in a particular neutron cross-section evaluation (such as ENDF/B-IV) are accounted for in the energy range from about 0.00001 eV to about 20 MeV. Users have the choice of three sources of neutron cross sections: ENDF/B,[1] Howerton's ENDL library from Livermore,[2] and the British (AWRE) library. Furthermore, there is the choice of using prompt or total fission $\bar{\nu}$ as well as the option of using discrete reaction cross sections in which the reaction cross sections are multigrouped.

The angular distributions for elastic and inelastic events are prescribed on a fine grid of incident neutron energies. Linear interpolation yields the angular distribution for the particular incoming neutron energy. This distribution is then sampled in a continuous fashion.

Similarly, the energy distributions for secondary neutrons from inelastic reactions (none needed for level scattering, of course) are also stored on a fine energy grid. These distributions are obtained from the laws prescribed in the particular cross-section evaluation. Linear interpolation yields the energy distribution for the incident neutron energy. Again, the distribution is sampled continuously.

If the total fission cross section alone is given, then that cross section is used with assumed behavior for the breakup into (n,n')f and (n,2n)f. However, if the reactions (n,f), (n,n')f and (n,2n)f are explicitly tabulated, then these cross sections along with the associated angular and secondary energy distributions are used directly.

The energy spectra for photons produced by neutron interactions are given in terms of 20 equally probable photon energies for each of 30 incident neutron energy groups.

There are two thermal neutron treatments in MCNP. One is the free-gas model in which, for elastic collisions, light atoms (for Z = 1 through 8) are assumed to be in a Maxwellian distribution with some thermal temperature that may be a function of time. Secondly, neutron thermal scattering can be modeled by the $S(\alpha,\beta)$ scattering model which includes chemical binding and crystalline effects that become important as the neutron's energy becomes sufficiently low. $S(\alpha,\beta)$ data are currently available for light and heavy water, beryllium metal, graphite, and polyethylene. Others will be added as they become available. Typically one will use the free-gas model from around 10 eV to 4 eV and will then switch to the $S(\alpha,\beta)$ model. $S(\alpha,\beta)$ effects are most significant below 2 eV.

B. Photons

The photon interaction cross sections come from the Storm and Israel[3] evaluation which covers the energy range 0.001 to 100.0 MeV. Below 0.001 MeV, MCNP allows only analog capture for photons which will rapidly terminate them.

MCNP takes account of incoherent (using an inverse fit rather than a rejection scheme on the Klein-Nishina distribution) and coherent scattering, the possibility of fluorescent emission following photoelectric

absorption, and absorption in pair production with local emission of the annihilation quanta.

III. SOURCES

MCNP has five standard sources in addition to the provision to allow the user to provide his own source subroutine.

The five standard sources (which all allow energy and directional biasing) are:

(1) point isotropic,
(2) outward cosine distribution on a spherical surface,
(3) inward cosine distribution on a spherical surface,
(4) uniform distribution in volume, and
(5) a plane-wave source.

IV. VARIANCE REDUCTION

Standard optional variance reduction schemes in MCNP include geometry splitting and Russian roulette, weight cutoff with Russian roulette, time and energy cutoff, correlated sampling, analog capture or implicit capture by weight reduction, the exponential transformation, energy splitting, forced collisions in designated cells, flux estimates at point or ring detectors, track-length estimators, and source biasing (both in energy and direction).

Flux estimates at point detectors are determined by two methods, the next-event estimator (or analog) scheme and the once-more-collided flux estimator (OMCFE). A scheme, DXTRAN, to improve sampling in the vicinity of detectors or other tallies is available. It involves deterministically transporting pseudo-particles on collision to some arbitrary, user-defined sphere in the neighborhood of a tally and then calculating contributions to the tally from these pseudo-particles. Contributions to the detectors or to the DXTRAN spheres can be controlled as a function of geometric cell or as a function of the number of mean free paths from a collision point to the detector or DXTRAN sphere.

V. TALLIES AND OUTPUT

The user-defined output of MCNP includes two-way current as a function of direction across any subset of surfaces (or surface segments) in the problem. Fluxes across any set of surfaces are available also. Similarly,

the flux at designated detectors (points or rings) and the average flux in a cell (track length per unit volume) are standard tallies. The heating tallies give the energy deposition in specified cells. In addition, particles may be flagged when they cross specified surfaces or enter designated cells, and the contributions of these flagged particles to the tallies are listed separately. Reactions such as fissions, absorptions, tritium production (or any product of the flux times the approximately one hundred standard ENDF reactions plus several nonstandard ones) may be tallied with any of the MCNP tallies. In fact, any quantity of the form

$$C \int \varphi(E)\ f(E)\ dE$$

may be tallied where $\varphi(E)$ is the energy-dependent flux and f(E) is any product of the quantities in the cross-section libraries or a function provided by the user. Tallies may be made for segments of cells and surfaces without having to build the desired segments into the actual problem geometry. All tallies are a function of time and energy and are normalized to be per unit starting particle.

Printed out with each tally is also its relative error corresponding to one standard deviation of the mean.

Standard summary information is also printed to give the user a better idea of how his problem ran. This information can give insight into the physics of the problem and the adequacy of the Monte Carlo simulation. If errors occur during the running of a problem, detailed debug prints are given.

VI. CRITICALITY

MCNP has the capability to calculate eigenvalues for both sub- and super-critical critical systems. The calculation is run as a series of generations of neutrons. At the end of each generation, k_{eff} is calculated for that generation as well as averaged over a specified number of preceding generations.

The source for the first generation is usually defined by the user at a set of spatial points in the system. The neutrons are then started at these points isotropically and with an energy sampled from standard fission distributions.

Three estimators (in various combinations) are used to calculate k_{eff}: absorption, collision, and track-length estimators.

VII. OTHER FEATURES

MCNP can be run interactively or in a batch mode.

An arbitrary cross-sectional view of the input geometry can be plotted on 35-mm film, 105-mm microfiche, on an interactive graphics display terminal, or on a 36-inch Versatec electrostatic plotter. A file, $PLOT$, of plotted points is produced to facilitate using other graphical devices.

A feature is available to allow the user to translate and/or rotate the entire geometry or a part of it from one coordinate system to another.

For use by some of the tallies, volumes of cells (regardless of axis of symmetry) are calculated if they are bounded by surfaces of revolution. Irregular volumes can be calculated stochastically. Surface areas of cells are also calculated.

Full restart capabilities, used for machine failure or continuing a run to obtain better statistics, are available. This includes periodically dumping to magnetic tape or disk automatically.

The user is allowed to specify almost any information desired to be written to a file for post-processing, such as for plotting results or to generate a source for a subsequent problem. Facilities also are included to read in information from a source or data file.

An event-log feature is available for debugging purposes that prints out the complete life history of a given particle.

CHAPTER 2
GEOMETRY, PHYSICS, AND MATHEMATICS

I. INTRODUCTION

This manual is written as a practical guide for the use of MCNP. This second chapter discusses in more detail the mathematics and physics of the MCNP geometry, cross-section libraries, variance reduction schemes, the Monte Carlo simulation of the neutron and photon transport, and the tallies. This discussion is not meant to be exhaustive; many of the details of the particular techniques and of the Monte Carlo method itself will have to be found elsewhere. Carter and Cashwell's book *Particle-Transport Simulation with the Monte Carlo Method*[4] is a good general reference on radiation transport by Monte Carlo and is based upon what is in MCNP.

A. History

Although the first use of Monte Carlo techniques is difficult to trace, the emergence of the Monte Carlo method as a radiation transport research tool springs from work done at Los Alamos during the Second World War. The credit for the so-called invention of Monte Carlo as a mathematical discipline is generally attributed to Fermi, von Neumann, and Ulam. Their ideas were developed by many followers in various laboratories, but the name of von Neumann seems to be attached to many, if not most, of the fundamental ideas and techniques. Metropolis and Richtmyer should also be credited with some of the early Monte Carlo development at Los Alamos.

In 1947 while at Los Alamos, Fermi invented a mechanical device to trace neutron movements through fissionable materials by the Monte Carlo method. The FERMIAC[5] is on display in the Bradbury Science Museum at LASL.

Monte Carlo at LASL has had, since its inception, a number of devoted and well-known disciples. In the early 1950's there was assembled at Los Alamos, under Ulam's guidance, a small band of scientists engaged in investigating transport methods and solving transport problems by Monte Carlo. At first the calculational effort was rudimentary, but with the completion of the MANIAC computer, the work gained momentum and rapidly became a factor in much of the research at Los Alamos. Much of the early work is summarized in the first book to appear on the subject of Monte

Carlo by Cashwell and Everett.[6] Shortly thereafter there appeared at LASL the Monte Carlo neutron transport code MCS[7] which incorportated a general geometry treatment.

The addition of several more features resulted in the Monte Carlo neutron code MCN[8] in 1965. These features included the improvement and expansion of the geometry treatment, provision for standard cross-section libraries, inclusion of standard variance reduction techniques, inclusion of general sources, treatment of thermalization of neutrons by the free-gas model, and more general tallies and output. The photon codes MCG and MCP[9] were then added to the LASL family of Monte Carlo codes. Both of these dealt with photon transport, MCG treating higher energy photons or gamma rays and MCP treating photons of energies down to 1 keV. MCN and MCG were merged to form MCNG (the predecessor of MCNP) in 1973. Gamma-production cross sections were added to enable the gamma rays produced by neutron interactions to be generated.

B. MCNP Structure

MCNP is the culmination of all the above work and codes plus many additions in mathematics, physics, and user-oriented features. The first version of MCNP was released in June, 1977. There have since been two revisions, versions 1A and 1B. Version 2 is the first major revision to MCNP, although as far as the user is concerned its input is almost completely backward compatible to the early MCN and MCG codes.

The general structure of MCNP is based on overlays. The main overlay MCNP calls up to four other overlays, depending upon the user's requirements. IMCN (overlay 1C0 and commonly called the initiation code) processes the problem input file and calculates volumes and areas. PLOT (overlay 2C0) plots, translates and/or rotates the problem geometry. XACT (overlay 3C0) processes cross sections for the specified materials. MCRUN (overlay 4C0 and commonly called the run code) does the actual particle transport. Reference is frequently made (mainly for historical reasons) to the MCRUN overlay (the run code) in three different ways: MCN refers to the neutron portion, MCG refers to the simple-physics photon treatment, and MCP refers to the detailed-physics photon treatment which includes fluorescence. The ERGP input card (see page 159) determines whether MCP or MCG is used. This terminology will be used in the rest of the manual. The criticality portion of MCRUN is frequently called the KCODE.

MCNP can be run in three modes: Mode 0 is neutron transport only, Mode 1 is combined neutron-photon transport, and Mode 2 is photon transport

only. In Mode 1, photon histories are followed as part of their parent neutron histories rather than being saved and followed later after all the neutron histories have been completed as was the case with MCNG.

Variants of MCNP are available that offer a geometrical perturbation feature; a multigroup adjoint capability; a state-space splitting capability; flux, importance, and contributon flux data generation with subsequent plotting; and a calculation of the variance of the variance. Some of these features may become a permanent part of MCNP at a later date.

Future work includes a geometrical lattice capability, the probability table method for unresolved resonances, photo-neutron reactions, improved variance reduction schemes such as better source biasing and a weight window to control high and low weighted tracks, and higher energy neutron cross sections (perhaps up to 60 MeV in some cases). Furthermore, X-6 always strives to be current with the best cross-section data such as ENDF/B-V.

A companion report, MCNP Applications, will be written soon and will give examples of a variety of problems and also examples of some of the more common changes made to MCNP by users for particular requirements.

C. MCRUN Flow

The basic flow of the MCRUN overlay for a coupled neutron-photon problem is as follows. At the beginning of MCRUN the computer field length is set to be the minimum required for the particular job. After other initializing, a particle is started by calling HSTORY. In HSTORY, the starting location in the random number sequence for the history is set, the number of the history NPS being run is incremented, and ISOS is called which generates a set of U,V,W direction cosines from an isotropic distribution regardless of what type of source is to be used - they are available for a user-provided source. Back in HSTORY, the flag IPT is set for the type of particle being run; 1 for a neutron and 2 for a photon.

Next SOURCE is called for a user-provided source (NSR = 0), or SOURCA is called for one of the five standard sources, or SOURCK (NSR = 71) is called if the KCODE is being used. In SOURCA, the initial parameters are set depending on the type of source requested (NSR = 1 to 5). All of the parameters describing the particle are set in the source routine including position, direction of flight, energy, weight, time, and starting cell (and possible surface). Subroutines called by SOURCA include CBIAS for direction biasing within a cone, ERGSMP to sample the starting energy from a distribution, ROTAS to rotate a direction vector through a polar and

azimuthal angle, NEWCEL to find the cell into which a particle has entered, and CHKCEL to make sure a new cell is legitimate. CHKCEL is also called after SOURCE for the first 100 particles to verify that the starting points are in the cell(s) expected.

At the next point in HSTORY, some of the summary information is incremented (see Appendix C for an explanation of these arrays), and then the actual particle transport is started (banked photons from neutron collisions also start here). First, TALLYD is called to score any detector contributions, and then DXTRAN is called (if used in the problem) to create pseudo-particles on the spheres and then bank them with BANKIT. Next, the energy of the particle is checked against energy cutoff. The track is terminated (NTER=3) if it is below the energy cutoff and logged with other terminations. (Most banked particles return to this point.) Energy splitting is then dome if required and the additional particles banked.

TRACK is next called in which the intersection of the trajectory of the particle with every bounding surface of the cell IA the particle is in is calculated. The minimum positive distance DLS to a surface indicates the next surface JA the particle is heading toward. For neutrons, ACETOT is called to calculate the cross sections for cell IA using a binary table lookup, and PHOTOT is called for photons where a linear table lookup is used (but only a maximum table length of forty is involved). The total cross section is modified in HSTORY by the exponential transformation if necessary. The distance PMF to the next collision is determined (if a forced collision is required, the uncollided part is banked). The track length D of the particle in the cell is found as the minimum of either the distance PMF to collision or distance DLS to the surface JA, and TALLY is called to increment any cell tallies. Some summary information is incremented, and the particle's parameters (time, position, and energy) are updated. If the particle exceeds the appropriate time cutoff, the track is terminated (NTER=2) and logged, and the bank is checked for additional tracks.

If the distance to collision PMF is less than the distance DLS to surface JA, the particle undergoes a collision. Otherwise, it crosses a surface and SURFAC is called to do any surfaces tallies (by calling TALLY) and process the particle through the surface into the next cell by calling NEWCEL. It is in SURFAC that reflecting surfaces, geometry splitting, Russian roulette (NTER=6), and loss to escape (NTER=1) are treated. For splitting, one bank entry of NPA particle tracks is made in BANKIT for an (NPA+1)-for-1 split. The bank is the IBNK array, and entries or retrievals

are made with the PBL array (the bank operates strictly on a last-in, first-out basis). The history is continued by calling TRACK in HSTORY.

If the particle did not make it all the way to surface JA, everything about the collision is determined in COLIDN for neutrons and COLIDP for photons. COLIDN determines which nuclide is involved in the collision, generates and banks any photons, handles analog capture or capture by weight reduction and plays the weight cutoff game (NTER=5), handles thermal collisions and then non-thermal collisions such as elastic or inelastic scattering. COLIDP for photons is similar, but it covers the MCG or MCP types of physics.

At this point, the history of a viable particle is continued by going back to the beginning of the actual transport of HSTORY where the detector tally is made. Otherwise, the bank is checked for any remaining progeny, and if none exists, the history is terminated. Appropriate summary information is incremented, the tallies of this particular history are added to the total tally data by TALSHF, and a return is made to MCRUN.

Back in MCRUN, checks are made to see if the job should be terminated because enough histories have been run or too little time remains to continue. For continuation, HSTORY is called again. Upon termination SUMARY is called to print the summary information, and then SUMMARY calls TALLYP to print the tally data. Details of the MCNP tally structure are found in Appendix D; this is the most frequently user-modified part of MCNP.

II. GEOMETRY

MCNP treats geometric cells bounded by first- and second-degree surfaces, as well as some fourth degree surfaces, in a Cartesian coordinate system. The surface equations recognized by MCNP are listed in Table 3.1 on page 122. The particular Cartesian coordinate system used is arbitrary and user defined, but most users at LASL choose the right-handed system shown in Figure 2.1. Any symmetry in the geometry is generally taken about the y-axis.

A. Cells

There are two basic ways to specify cells in MCNP: (1) by the intersection of regions defined by surfaces bounding a cell, or (2) by the

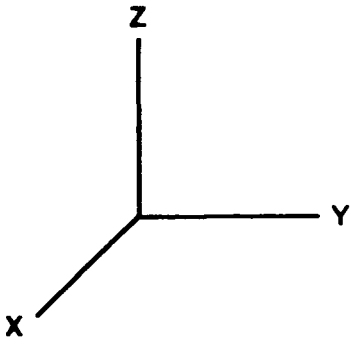


Figure 2.1

union of these regions. Combinations of the two are valid and, in practice, are the most frequent use. MCNP does not have combinatorial geometry as it is commonly denoted. Rather than combining (i.e., taking the union of) several pre-defined geometrical bodies as in combinatorial geometry, MCNP gives the user the added flexibility to define his own geometrical bodies from all the available surfaces in MCNP and then to combine them with the union operator.

The way to specify cells in MCNP is to list on a cell card the cell number, the material number and material density followed by a list of surfaces that bound the cell. After each bounding surface is a list of the cells on the other side of the surface, separated by commas. Each surface divides all space into two regions, one with positive sense with respect to the surface and the other with negative sense. The code interprets the geometry description as saying that the cell is the intersection of the listed regions. The union of these regions defined by the bounding surfaces may also be indicated on the cell card by the union operator, a colon.

A variation of the above cell-card format is not to specify otherside cells. The advantage of this is that there is less for the user to think about and less to input resulting hopefully in fewer mistakes. The disadvantage is that MCNP cannot make the same initial checks of the geometry's integrity but must rely on errors to become apparent (hopefully quickly) after the particle transport begins.

Each of the above cell specification techniques will now be discussed.

The subdivision of the physical space into cells is not necessarily governed by the different material regions occurring, but may take into consideration problems of sampling and variance reduction techniques (such as splitting and Russian roulette), the need to specify an unambiguous geometry, and the requirements of the tallies. However, using the cell union operator and the tally segmentation feature will reduce (if not eliminate) most of these requirements.

Caution should be exercised about making any one cell very complicated. For example, with the union operator and disjointed regions, a very large geometry can be set up with just one cell. The problem is that at each flight of a track in a cell, the intersection of the track's trajectory with each bounding surface of the cell is calculated. This can be costly. The geometry of Figure 2.2a is quite easy to set up; it is just a lot of parallel cylinders. However, the cell containing all the little cylinders is bounded by fourteen surfaces (counting a top and bottom). A much more efficient geometry is seen in Figure 2.2b where the large cell has been broken up. In an actual problem very similar to this, the job with the geometry of Figure 2.2b ran about four times faster than with the

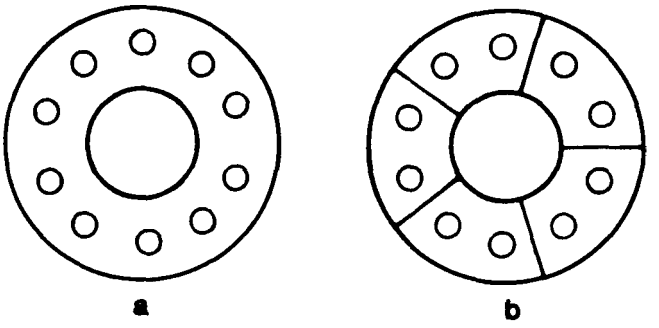


Figure 2.2

geometry of Figure 2.2a.

When defining cells by any method, an important concept is that of the *sense* of a cell with respect to a bounding surface. Suppose that f(x,y,z)=s=0 is the equation of a surface in the problem. For any set of points (x,y,z), if s=0 the point is on the surface. However, for points not on the surface, if s is negative the point is said to have a negative sense with respect to that surface and, conversely, a positive sense if s is positive. For example, a point at x=3 has a positive sense with respect to the surface x-2 = 0. That is, the equation x-D = 3-2 = s is positive.

It is essential that the description of the geometry of a cell be such as to eliminate any ambiguities as to which region of space is meant. That is, a particle entering a cell should be able to uniquely determine which cell it is in from the senses of the bounding surfaces. This eliminates a geometry such as shown in Figure 2.3 unless an *ambiguity surface* is specified. Suppose the figure is rotationally symmetric about the y-axis. A particle entering cell 2 from the inner spherical region might think it were entering cell 1 because a test of the senses of its coordinates would satisfy the description of cell 1 as well as that of cell 2. In such

cases, an ambiguity surface is introduced such as a, the plane y = 0. An ambiguity surface need not be a bounding surface of a cell, but of course it may be and frequently is. It can also be the bounding surface of some cell other than the one in question. In fact, it does not have to be a bounding surface at all. However, the surface must be listed among those in the problem and must not be a reflecting surface (see page 21). Referring to cells 1 and 2 in Figure 2.3, the description of each is augmented by listing its sense relative to surface *a* as well as that of each of its other bounding surfaces. A particle in cell 1 cannot have the same sense relative to surface *a* as does a particle in cell 2. More than one ambiguity surface may be required to define a particular cell.

The signed problem number of a surface that is to be used as an ambiguity surface for a cell is listed as an entry on that cell card with

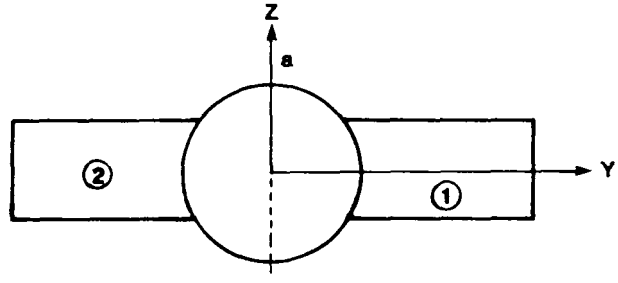


Figure 2.3

no otherside cell entries after it.

1. Cells by Intersections

The intersection operator in MCNP is implicit; it is simply the blank space between two sets of surface-cell relations on the cell card.

If geometric cells are specified as the regions delineated by the intersections of the spaces defined by a set of surfaces, MCNP requires that all points in a cell have the same sense with respect to a bounding surface of that cell. This means that all points in a cell must always remain on the same side of a bounding surface of that cell, and this must be true for each bounding surface of the cell. Another interpretation of this is that no concave corners can be in a cell. This requirement is illustrated in Figure 2.4 where the intersection of the regions bounded by the five surfaces form the cell (ignore surface 6 for the time being).

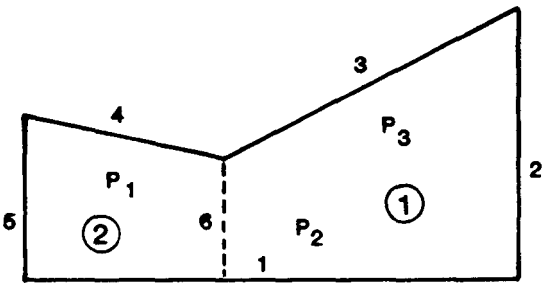


Figure 2.4

In Figure 2.4 surfaces 3 and 4 form a concave corner in the cell such that points p_1 and p_2 are on the same side of surface 4 (i.e., have the same sense with respect to 4) but point p_3 is on the other side of 4 (opposite sense). Points p_2 and p_3 also have an opposite sense to surface 3 than p_1 does. One way to remedy this (and there are others) is to add the surface 6 between the 3·4 corner and surface 1 to divide the original cell into two cells.

In Figure 2.4 with surface 6 added, call the cell to the right of surface 6 number 1; to the left, number 2; and the outside cell number 3. The cell cards (in two dimensions and all cells void) are

```
1     0    1,3     -2,3  -3,3  6,2
2     0    1,3     -6,1  -4,3  5,3
3     0   -1,1,2    2,1   3,1  4,2  -5,2
```

This says that cell 1 is the intersection of the region above surface 1 with the region to the left of surface 2 and that intersected with the region below surface 3 and finally that intersected with the region to the right of surface 6. The format of these cell cards is explained on page 118.

With these cell specifications, MCNP tracks particles through the geometry. At a collision point, MCNP knows of all bounding surfaces of the cell containing the collision. It then calculates the intersection of the track's trajectory with each bounding surface and finds the minimum distance to an intersection. If the distance to the next collision is greater than this minimum distance, the particle leaves its current cell. At the appropriate surface intersection, MCNP finds the correct cell on the other side of the surface by checking the sense of the intersection point for each surface of each cell listed on the other side of the surface in

the order listed. When a complete match is found, MCNP has found the correct cell on the other side and the transport continues.

Note that the outside cell 3 has concave corners and is not properly defined. However, for a combination of two reasons (both required) this is acceptable in this situation: (1) it is an outside cell that will be assigned an importance of zero so that particles will be killed when they enter the cell, and (2) MCNP does not verify the consistency of cell-surface relations when a particle leaves a cell if there is only one cell listed on the other side of a surface on that cell card. Thus an illegitimate outside cell is acceptable if it is the only cell listed on the other side of a real problem cell and has zero importance.

For example, in Figure 2.5a, this arrangement would fail (even though the outside cell has zero importance) because the cell card for cell 1 is

```
1      0   -3,2,3   etc.
```

Because there are two otherside cells listed, the code checks for consistency when a particle leaves cell 1 and finds cell 3 is not a legitimate cell (it has both positive and negative sense with respect to surface 3). One way around the problem is shown in Figure 2.5b when the large interior cell has been divided into two cells with the resulting cell card

```
1      0   -3,4   etc.
```

which has only one otherside cell listed. A second way around the problem is shown in Figure 2.5c where two outside cells are used; this is easier to set up than the method of Figure 2.5b. The outside cell 3 is in the niche above cell 1 and left of cell 2; the outside cell 4 is everyplace else. The communication between the interior and outside cell can then be described by listing only one otherside cell; the communication between the two outside cells can be ignored. For Figure 2.5c, the cell cards for a two-dimensional geometry are

```
1      0    1,4   -2,4   -3,2,3   4,4
2      0    3,1   -2,4   -6,4     5,3
3      0   -5,2    3,1
4      0   -4,1   -1,1    2,1,2   6,2
```

Cell 4 to the left of surface 5 in Figure 2.5c would also be acceptable.

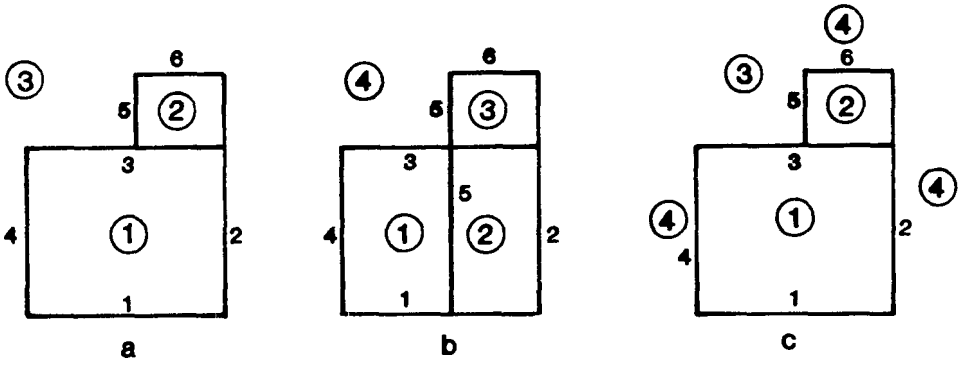


Figure 2.5

(Note that this is only an example; a complete three-dimensional geometry must always be specified for MCNP.)

There is an easy way to check the consistency of the cell-surface relationships. If cell 1 in the above example is bounded by surface 1 with positive sense and sees cell 4 on the other side, then it follows that cell 4 is bounded by surface 1 with negative sense with respect to cell 1. Any inconsistency will show that the geometry is not properly specified. MCNP makes the same check in the initiation overlay. However, be cautious in that this test does not guarantee a correct geometry. The geometry should always be plotted and examined carefully for other errors.

2. Cells by Unions

The union operator, signified by a colon on the cell cards, does allow concave corners in cells plus cells that are completely disjoint. These cells that are illegitimate when defined by the intersection of spaces are called Godfrey cells. The intersection and union operators are binary Boolean operators, and their use follows Boolean algebra methodology.

Spaces on either side of the union operator are irrelevant, but remember a space without the colon signifies an intersection. In the hierarchy of operations, intersections are performed first and then unions. There is no left to right ordering. Parentheses may be used to clarify operations and in some cases are required to force a certain order of operations. Spaces are optional on either side of a parenthesis. In fact, a parenthesis is equivalent to a space (after any ordering has taken place) and therefore signifies an intersection.

For example, let A and B be two regions of space. The region containing points which belong to both A and B is called the intersection of A and B. The region containing points which belong to A alone or to B

alone or to both A and B is called the union of A and B. The lined area in Figure 2.6a represents the union of A and B, and the lined area in Figure 2.6b represents the intersection of A and B. The only way spaces can be added is with the union operator. An intersection of two spaces always results in a region no larger than either of the two spaces. Conversely, the union of two spaces always results in a region no smaller than either

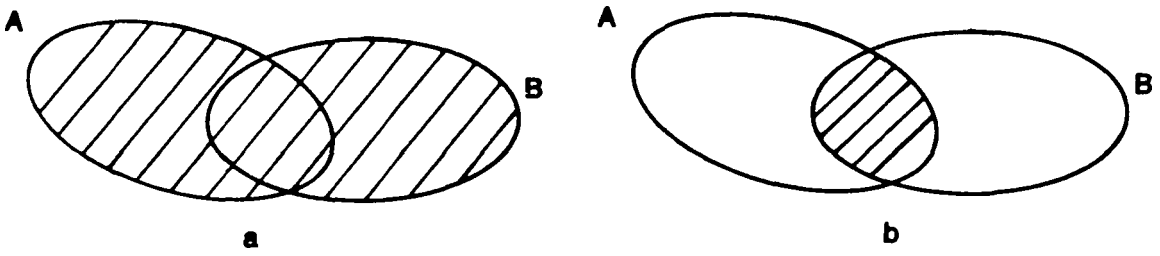


Figure 2.6

of the two spaces.

In Figure 2.4 on page 14, for one interior cell numbered 1 and one outside cell numbered 2, the MCNP cell cards in two dimensions are (assuming both cells are voids):

```
1     0    1,2      -2,2      (-3,2  :  -4,2)  5,2
2     0   -5,1  :  -1,1  :    2,1  :   3,1    4,1
```

This says that cell 1 is the space defined by the intersection of the region above surface 1 with the region to the left of surface 2 with the region below the union of the spaces below surfaces 3 and 4 and finally with the region to the right of surface 5. Note that cell 2 contains four concave corners (all but between surfaces 3 and 4), and its specification is just the converse of cell 1. Cell 2 is the space defined by the region to the left of surface 5 plus the region below 1 plus the region to the right of 2 plus the space defined by the intersections of the regions above surfaces 3 and 4.

A simple consistency check can be noted with the above two cell cards. For cell 1, wherever there is an intersection, it becomes a union for cell 2 and vice versa.

Note that in this example, all legitimate corners in a cell are handled by intersections and all illegitimate corners are handled by unions. This rule of thumb holds in all the following examples.

To illustrate some of the concepts about parentheses, assume an intersection is thought of as multiplication and a union is thought of as addition with multiplication being done first unless parentheses are present. The cell cards for the example cards above from Figure 2.4 may be written in the form

```
1      a*b*(c+d)*e
2      e+a+b+c*d
```

Note that parentheses are required for the first cell but not for the second, although the second could have been written as e+a+b+(c*d), (e+a+b)+(c*d), (e)+(a)+(b)+(c*d), etc.

Note that in Figure 2.5a the outside cell 3 is a Godfrey cell, and this geometry can be legitimately described with the union operator:

```
1     0   4,3      1,3      -2,3      -3,2,3
2     0   5,3      3,1      -2,3      -6,3
3     0  -4,1  :  -1,1  :   2,1,2  :   6,2   :  -5,2  3,1
```

In fact, cell 2 can be eliminated entirely and made part of cell 1 as seen in Figure 2.7. Then both cells 1 and 3 are Godfrey cells:

```
1     0   1,3      -2,3      -6,3     (5,3  :  -3,3)  4,3
3     0  -4,1  :  -1,1  :   2,1  :   6,1  :  -5,1   3,1
```

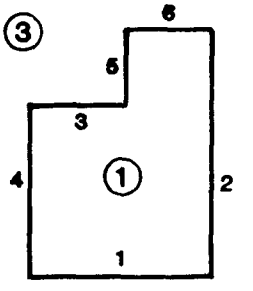


Figure 2.7

Several more examples with the union operator will be given in Chapter 4 with the intent that you can study them to get a better understanding of this powerful operator that can greatly simplify geometry setups.

3. Otherside Cells

A variation of the above ways to specify cells is not to list on a cell card the cells on the other side of bounding surfaces. Simply list all the surfaces bounding a particular cell and the relation between the regions bounded by a surface (i.e., union or intersection).

There are two advantages of not specifying otherside cells: (1) it's less taxing for the user and will hopefully reduce setup errors since there is less to think about, and (2) when the code finds the otherside cells during the course of running a problem, it usually lists them in an optimal order which may save some running time.

The main disadvantage of this method is that the check of surface-cell relations is bypassed in the initiation overlay; any geometry error of this type will result in a lost particle in the run overlay. A quick way to check a geometry is to remove the materials to create a void and then run particles through it.

There are two restrictions when otherside cells are not listed:

(a) The outside cell must be specified properly. It cannot be an illegitimate, zero-importance cell specified by only intersections; it can be a Godfrey cell specified with unions. In fact, it is simple to set up one legitimate outside cell with the union operator for any problem.

(b) You cannot mix listing and not listing otherside cells. You must be consistent throughout. Whether you use unions or intersections or a combination is irrelevant, however.

As far as the user is concerned, ambiguity surfaces are specified exactly the same way as before - simply list the signed surface number as an entry on the cell card. As far as MCNP is concerned, if a particular ambiguity surface appears on only one cell card, it is treated as a true ambiguity surface. If it appears on more than one cell card, it still functions as an ambiguity surface but the TRACK subroutine will try to find intersections with it thereby using a little more computer time.

As an example of not specifying otherside cells, the cell cards for the geometry in Figure 2.7 are

```
1     0   1   -2   -6   (5 : -3)   4
3     0   -4 : -1 : 2 : 6 : -5     3
```

B. Surfaces

The standard first- and second-degree surfaces plus even the fourth-degree elliptical and degenerate tori of analytical geometry are all available in MCNP. The surfaces are designated by mnemonics such as C/Z for a cylinder parallel to the z-axis. A cylinder at an arbitrary orientation is designated by the general quadratic GQ mnemonic. A parabaloid parallel to a major axis is designated by the special quadratic SQ mnemonic. There are twenty-nine mnemonics representing all types of surfaces.

Any surface (except an ambiguity surface) may be designated a reflecting surface by preceding its number with an asterisk. Any particle hitting a reflecting surface is specularly (mirror) reflected. Reflecting planes are valuable because they can simplify a geometry setup (and also tracking) in a problem. They can, however, cause difficulties (or even make it impossible) to get the correct answer. The user is cautioned to check the source weight and tallies to insure that the desired result is achieved. Any tally in a problem with reflecting planes should have the same expected result as the tally in the same problem without reflecting planes. Detectors should not be used with reflecting surfaces (see page 63).

Two surfaces, the cone and torus, require more explanation. The general equation for a cone describes a cone of two sheets (just like a hyperbola of two sheets) - one sheet is a cone of positive slope, and the other has a negative slope. A two-sheeted cone requires an ambiguity surface to distinguish between the two sheets as well as proper cell specification for both sheets. MCNP provides the option to select either of the two sheets which frequently simplifies geometry setups and eliminates any ambiguity. The +1 or the -1 entry on the cone surface card causes the one sheet cone treatment to be used. The sign of the entry becomes the sign of t (the slope, see Table 3.1) for the sheet desired.

The treatment of fourth degree surfaces in Monte Carlo calculations has always been difficult because of the resulting fourth order polynomial ("quartic") equations. These equations must be solved in order to find the intersection of a line of flight of a particle with a toroidal surface. In MCNP these equations must also be solved to find the intersection of toroidal surfaces with other surfaces in order to compute the volumes and surface areas of geometric regions of a given problem. In either case, the quartic equation,

$$x^4 + Bx^3 + Cx^2 + Dx + E = 0$$

is difficult to solve on a computer because of roundoff errors. For many years the MCNP toroidal treatment required 30 decimal digits (CDC double-precision) accuracy in order to solve quartic equations. Even then there were roundoff errors which had to be corrected by Newton-Raphson iterations. Schemes using a single-precision quartic formula solver followed by a Newton-Raphson iteration were inadequate because if the initial guess of roots supplied to the Newton-Raphson iteration is too inaccurate, the iteration will often diverge when the roots are close together.

A recent generalization of the MCNP geometry package has prompted a significant upgrade in the treatment of toroidal surfaces. Quartic equations are now solved in two different ways depending upon the application. In the volume and surface area calculation, where speed is unimportant because the quartic equation is only infrequently solved, a fully iterative single-precision scheme is used. This scheme uses a Newton-Raphson procedure applied to the Vieta formulas to solve all zeros of the fourth-order quartic equation simultaneously. It is very accurate but also very slow. In the transport portion of MCNP the speed of the quartic solver is very important. If the above iteration scheme is used then the MCNP running time may be as much as 10-100 times slower than the old double-precision scheme of MCNP. Therefore, a new single-precision (15 decimal digits) approach has been adopted.

The new single-precision quartic algorithm in MCNP, like its double-precision predecessor, basically follows the quartic solution of Cashwell and Everett.[10] When roots of the quartic equation are well separated, a modified Newton-Raphson iteration quickly achieves convergence. But the key to the new method is that if the roots are double roots or very close together, then they are simply thrown out. This is because a double root corresponds to a particle trajectory being tangent to a toroidal surface, and it is a very good approximation to assume that the particle then has no contact with the toroidal surface at all. In extraordinarily rare cases where this is not a good assumption (such a case has yet to be found in a realistic problem) the particle becomes "lost" and then the slower, full-iteration scheme is used as a backup. Additional refinements to the quartic solver include a carefully selected finite size of zero, the use of a cubic rather than a quartic equation solver whenever a particle is transported from the surface of a torus, and a gross quartic coefficient check to ascertain the existence of any real positive roots.

As a result, the new single-precision quartic solver is substantially faster than former schemes and also somewhat more accurate.

Treatment of toroidal surfaces in MCNP has been enhanced by other changes in addition to an improved quartic solver. Elliptical tori symmetric about any axis parallel to a major axis may now be specified. The volume and surface area of various tallying segments of a torus may also be calculated.

There are two ways to specify surfaces in MCNP: (1) by supplying the appropriate coefficients needed to satisfy the surface equation, and (2) for surfaces symmetric about an axis by specifying known geometrical points on the surface.

1. Coefficients

The first way to specify a surface is to use one of the equation mnemonics from Table 3.1 on page 122 and to calculate the appropriate coefficients needed to satisfy the surface equation. For example, a sphere of radius 3.62 with the center located at the point (4,1,-3) is specified simply by

```
S        4  1  -3  3.62
```

An ellipsoid whose axes are not parallel to the coordinate axes is defined by the GQ mnemonic plus up to ten coefficients of the general quadratic equation. Calculating the coefficients can be (and frequently is) nontrival, but the task is greatly simplified by centering the desired ellipsoid about the origin of the coordinate system with its axes on the coordinate axes. Then the simplified ellipsoid can be appropriately translated and rotated to find the necessary coefficients by using the MOVE option in the PLOT overlay of MCNP (see Appendix B).

The use of the SQ (special quadratic) and GQ (general quadratic) surfaces is determined by the orientation of the axes. One should never use a more complicated specification (i.e., a GQ instead of an S) unless it is required since it will unnecessarily slow down the calculations.

2. Points

The second way to specify a surface is not to specify the surface equation coefficients but to specify known points on the surface. This is very convenient if you are setting up a geometry from something like a

blueprint where you know the coordinates of intersections of surfaces or points on the surfaces. When one or more surfaces intersect at a point, this second method also guarantees the point of intersection if the common point is used in the surface specification. It is frequently difficult to get complicated surfaces to meet at one point if the surfaces are specified by the equation coefficients, and this results in lost particles.

There are, however, restrictions that must be observed when specifying surfaces by points that do not exist when specifying surfaces by coefficients. Surfaces described by points must be symmetric about the x,y, or z axes and they must be unique, real, and continuous. For example, points specified on both sheets of a hyperboloid are not allowed since the surface is not continuous. However, it is valid to specify points that are all on one sheet of the hyperboloid. (See the X,Y,Z input cards description on page 124.)

III. CROSS SECTIONS

MCNP uses continuous-energy data from six LASL-processed neutron cross-section libraries plus one photon cross-section library. A brief description of these libraries and a listing of their contents for neutrons are found in Appendix F. The format of the libraries is given in Appendix E in considerable detail. This appendix should be very useful for users making extensive modifications to MCNP involving cross sections or for users debugging MCNP at a fairly high level.

A. Neutrons

Neutron cross-section information in the Monte Carlo libraries comes principally from four sources: the Evaluated Nuclear Data File (ENDF/B), the Evaluated Nuclear Data Library (ENDL), the Atomic Weapons Research Establishment (AWRE) library, and special LASL evaluations. The ENDF/B evaluation is a national cross-section effort that has contributors from the thermal and fast nuclear reactor programs, the controlled thermonuclear reactor program, and the weapons program. ENDF/B evaluations change every few years. The ENDL evaluations come from Lawrence Livermore Laboratory and generally represent efforts connected with the weapons program at that laboratory. A much older evaluation is the AWRE library which comes from Great Britain. The AWRE library is no longer updated, since the British

now use ENDF/B; it is kept in the Monte Carlo libraries for reproducibility of old problems.

MCNP has a continuous energy treatment of nuclear data - even through the resonance region. All the reactions available in a particular data set, such as ENDF/B, are accounted for in MCNP.

The actual cross-section libraries used by MCNP are binary files in the ACE (A Compact ENDF) format. The differences between ENDF/B data and the more compact binary ACE data are (1) all interpolation is linear-linear, (2) resonance cross sections are represented as pointwise data which are Doppler broadened to a specific temperature, (3) pointwise heating numbers are added, (4) secondary angular distribution data are stored in tabular form as equally probable cosine bins (5) secondary neutron energy distribution data are described in any of eleven different ways prescribed by distribution laws, (6) scattering is assumed to be isotropic in the center-of-mass system if no angular data are provided for an inelastic collision, (7) fission $\bar{\nu}$'s are represented either in tabular form or as polynomial functions, and (8) photon production data are added in tabular form.

Processing of ENDF/B-formatted data into ACE format is usually done with the NJOY-MCNJOY code maintained by Group T-2 at LASL.

The neutron cross sections and angular distributions for the Monte Carlo library have been generated from the ENDF/B, ENDL and AWRE libraries under the constraint that the resulting library is accurate with linear-linear interpolation to within a relative error E_1 for the cross sections and E_2 for the angular distributions (both errors typically 0.1%). Linear interpolation is used between energy points with a few hundred to several thousand tabulated energy points, depending on the isotope. Cross sections are added at a sufficient number of points to insure that the linear-linear constraint reproduces the original cross-section evaluation to within E_1. Furthermore, cross sections for all reactions are given in the same energy grid. This faithful preservation of the ENDF/B, ENDL, and AWRE libraries has the advantage that the Monte Carlo cross-section libraries can be used with confidence for general applications throughout the laboratory. It suffers from the disadvantage that the cross-section sets are much larger than necessary for some applications.

The angular distributions for secondary neutrons from inelastic reactions are stored on a 33-bin coarse energy grid with linear interpolation between adjacent angles. These distributions are obtained from the laws prescribed in the particular cross-section evaluations and are preserved in the Monte Carlo libraries. Linear interpolation from data

given at discrete energies yields the secondary energy distribution for an incident neutron energy. If no angular data are provided for an inelastic collision, MCNP assumes the scattering is isotropic in the coordinate system in which the energy is given.

The fission reaction is treated with one of two methods. If the total fission cross section alone is given, then that cross section is used with assumed behavior for the breakup into (n,f), (n,n')f, and (n,2n)f. However, if the reactions (n,f), (n,n')f, and (n,2n)f are explicitly tabulated, they are used. In both cases the associated angular, neutron/fission and secondary energy distributions of the ENDF/B, ENDL, and AWRE libraries are preserved and are sampled continuously in the Monte Carlo codes.

Photon production cross-section data (as determined from secondary energy distribution information) are included in the library for some isotopes. These production cross sections are given on the same energy grid as the total cross section. The energy spectra for photons produced by neutron interactions are given in terms of 20 equally probable photon energies for each of 30 incident neutron energy groups. Presently, MCNP selects one of the 20 equally probable energies and does not interpolate. This forms twenty (n,γ) lines, each of the same height. The photons are assumed to be born with an isotropic angular distribution.

When a neutron collides with an isotope, heat (energy) is deposited locally by the recoil of charged particles and residual nuclei. The average local heating number is tabulated in the cross-section file at each energy point. The total particle heating is the sum of the local heating and the total photon energy deposited in each cell after the photon has been properly transported. If there are no photon production data for the isotope, the photon energy is included in the local heating number.

The only alteration of data performed by MCNP is the expunging of data outside the energy range of interest for a given problem by the EXPUNG subroutine and the thermal treatment applied to low-energy cross sections. These modifications are problem dependent (for example, a function of energy cutoff).

Appendix F indicates which nuclides do not have cross sections available in the entire range from 0.00001 eV to 20 MeV. Neutrons of energy greater than the upper limit of the cross sections are transported with cross sections at that upper limit. Below the lower energy limit of a cross-section table, neutrons are transported with the cross sections of that lower limit.

The user has the option to use discrete reaction neutron cross sections (pseudo-multigroup) for which only the neutron reaction cross sections have been put into 240 groups on the DRMCCS library. The secondary angular and energy distributions are identical to those in the regular continuous energy cross-section files, so this is not a true multigroup treatment. The advantage of using these discrete cross sections is that computer cross-section storage requirements are reduced for better performance in a timesharing environment. Accuracy is degraded by the discrete cross sections only when resonance self-shielding is important.

Use of the discrete treatment reduces average computer storage requirements per isotope by a factor of two and for some isotopes by a factor of seven. However, there are a few isotope evaluations that have fewer points than the 240 of their discrete counterpart - in which case computer storage will increase slightly.

When a threshold reaction is encountered and the particular energy is less than that of the threshold but still in the threshold group, the reaction is abandoned. To conserve neutrons the reaction selection process is repeated. Experience has shown that this rarely occurs, but how many times it does happen is indicated in the problem output. If the reaction is rejected 100 times, the problem is aborted and a debug message *energy was not within the band for the reaction 100 times* is printed in the output file.

Also available to the user is the choice of total or prompt fission $\bar{\nu}$. Generally, the total fission $\bar{\nu}$ is desired for reactor-type or steady state calculations where the effect of delayed neutrons may be important. However, if the lifetime of neutrons is so short that a steady-state condition has not been reached, prompt $\bar{\nu}$ is appropriate (see the TOTNU card).

A final neutron library is TMCCS that contains the $S(\alpha,\beta)$ data.

B. Photons

Only one set of photon cross sections (MCPLIB) is available for use in MCNP. In this file, the natural logarithms of the total and partial cross sections are tabulated as a function of the natural logarithm of the incident photon energy. Average photon heating is also tabulated on this same logarithmic energy grid. Incoherent and coherent atomic form factors as a function of momentum transfer are included for the calculation of secondary angular distributions. Fluorescence data are also available. The principal source for the photon data on this file is the compilation of

Storm and Israel (below 15 MeV) and the ENDF/B photon interaction files (above 15 MeV). The PXSEC (see Appendix E) code was used to access the ENDF/B cross sections, calculate the heating values, and merge the two data sources. PXSEC is currently being updated to process all photon interaction data available in ENDF/B.

Any photon with energy below 0.001 MeV is allowed to undergo only analog capture which will quickly terminate the history.

IV. PHYSICS

Two concepts are important in the following discussion for setting up the input and for understanding the output. These are particle weight and particle tracks.

As far as Monte Carlo simulation is concerned, one particle is equivalent to two particles if the two particles have the same properties (position, energy, etc.) but each has a weight of half the original particle. In other words, a particle of weight w may be replaced by any number of k identical particles of weights such that $w_1 + \ldots + w_k = w$. This is the basis for cell importances and surfaces where the games of splitting and Russian roulette are played in order to keep the original track population built up to provide a good sample size. Upon tallying, if a particle of weight = 1 scores x, an otherwise identical particle of weight = w scores wx to the tally.

When a particle starts out from a source, it is said to create a track. If that track is split 2 for 1 at a splitting surface, then a second track is created and there are now two tracks from the original source particle. If one of the tracks has an (n,2n) reaction, then one more track is started for a total of three. A track refers to each component of a source particle along its history. Some tallies use the length of a track in a given cell to determine a quantity of interest while others tally at each collision or surface crossing.

A. Neutron Interactions

When a particle (representing any number of neutrons depending upon the particle weight) collides with a nucleus, the following sequence of events occurs:

1. the collision nuclide is identified;

2. photons are optionally (Mode 1) generated for later transport;
3. neutron capture is modeled;
4. either thermal, elastic, or an inelastic collision is selected.

1. Selection of Collision Nuclide

If there are n different nuclides comprising the material in which the collision occurred, and if ξ is a random number on the unit interval (0,1), then the k^{th} nuclide is chosen to participate in the collision if

$$\sum_{i=1}^{k-1} \Sigma_{t\,i} < \xi \sum_{i=1}^{n} \Sigma_{t\,i} \leq \sum_{i=1}^{k} \Sigma_{t\,i}$$

where Σ_{ti} is the macroscopic total cross section of nuclide i.

2. Optional Generation of Photons

Once the collision nuclide is chosen, photons are generated if the problem being run is a combined neutron/photon run (Mode 1) and if the collision nuclide has a nonzero photon-production cross section. The number of photons produced is a function of neutron weight, photon weight limits (entries on the PWT card), photon-production cross section, neutron total cross section, and cell importance. No more than ten photons may be born from any neutron collision. Because of the large number of low weight photons created by neutron collisions, Russian roulette is played for particles with weight below the bounds specified on the PWT card. This results in fewer particles with larger weights. The photon weight of each particle before Russian roulette is

$$WGT_p = \frac{WGT_n\ \sigma_\gamma}{\sigma_T\ N_p}$$

where WGT_p = photon weight
WGT_n = neutron weight
N_p = number of emitted photons
σ_γ = photon production cross section
σ_T = total neutron cross section.

This Russian roulette game is played according to neutron cell importances. For a photon produced in cell IA where the minimum weight set

on the PWT card is WM_{IA}, let IN_{IA} be the neutron importance in cell IA and let IN_{SRC} be the neutron importance in the source cell. If $WGT_p > WM_{IA}*IN_{SRC}/IN_{IA}$, the photon survives with weight WGT_p and is saved for further transporting. If $WGT_p < WM_{IA}*IN_{SRC}/IN_{IA}$, the photon plays Russian roulette and survives with probability $WGT_p*IN_{IA}/(WM_{IA}*IN_{SRC})$ and is given the weight $WM_{IA}*IN_{SRC}/IN_{IA}$. These surviving photons are stored in the bank to be transported later during the history.

If the weight of the starting neutrons is not unity, setting all the WM_{IA} on the PWT card to a negative value will make the photon minimum weight relative to the source weight. This will make the number of photons being created roughly proportional to the biased collision rate of neutrons. It is recommended for most applications that such a negative number be used and chosen to produce from one to four photons per source neutron. The default values for WM_{IA} on the PWT card are -1 which should be adequate for most problems.

3. Capture

Once the collision nuclide is identified and any photons are stored in the bank, capture is treated in either one of two ways: analog or implicit. In analog capture the particle is killed with probability σ_a/σ_T where σ_a and σ_T are the absorption and total cross sections of the collision nuclide. The absorption cross section is specially defined for MCNP as the (n,x) cross section where x is anything except neutrons. Thus σ_a is the sum of σ_γ, σ_α, σ_D, ... etc.

For implicit capture, the neutron weight WGT is reduced to WGT' as follows:

$$WGT_n' = \left(1 - \frac{\sigma_a}{\sigma_T}\right) WGT_n$$

If the new weight, WGT', is below the problem weight cutoff (specified on the CUTN card) then Russian roulette is played resulting overall in fewer particles with larger weight.

For all particles killed by analog capture the entire particle energy and weight are deposited in the collision cell. For implicit capture, a fraction σ_a/σ_T of the incident particle weight and energy is deposited in the collision cell corresponding to that portion of the particle which was captured. This is the default method of neutron capture by MCNP.

The fraction of a particle surviving implicit capture, or any particle

surviving analog capture is scattered by either thermal, elastic, or inelastic processes. These scattering modes will now be discussed.

4. Thermal Collision

Thermal collisions are calculated only when requested by the user and under the following conditions:

1. the collision nuclide is hygrogen, deuterium, a nuclide specified by the user on a TI card, or a nuclide in an $S(\alpha,\beta)$ material specified on an MT card.
2. the incident neutron energy is less than the thermal cut-in energy specified by the user on the ERGN card.

When these conditions are met, scattering is determined by either a free-gas model or by the $S(\alpha,\beta)$ scattering law. In the free-gas model neutrons are assumed to be transported in a monatomic gas that has an isotropic Maxwellian distribution of velocities. In the more rigorous $S(\alpha,\beta)$ treatment the chemical binding and crystalline effects of the collision nucleus are also considered. Regardless of the thermal threatment used, the particle undergoing thermal scattering emerges from the collision with a new weight, energy, and direction which are used until the next collision.

a. Free-gas treatment:

The free-gas kernel is a thermal interaction model that results in a good approximation to the thermal flux spectrum in a variety of applications and can be sampled without tables. The effective scattering cross section in the laboratory system for a neutron of kinetic energy E is

$$\sigma_s^{eff}(E) = \frac{1}{v_n}\int\int \sigma_s(v_{rel})v_{rel}p(V)dVd\mu_t \qquad (2.1)$$

Here, v_{rel} is the relative velocity between a neutron moving with a scalar velocity v_n and a target nucleus moving with a scalar velocity V, and μ_t is the cosine of the angle between the neutron and the target direction-of-flight vectors. The scattering cross section for this relative velocity is denoted by $\sigma_s(v_{rel})$, and p(V) is the probability density function for the Maxwellian distribution of target velocities,

$$p(V) = \frac{2}{\pi^{1/2}} \beta^3 V^2 e^{-\beta^2 V^2}$$

with β defined as

$$\beta = \left(\frac{A}{2kT}\right)^{1/2}$$

In Equation 2.1, A is the mass of a target nucleus in units of mass of a neutron, k is the Boltzmann constant, and T is the equilibrium temperature of the target nuclei.

The most probable scalar velocity V of the target nuclei is β which corresponds to a kinetic energy of kT for the target nuclei. This is not the average kinetic energy of the nuclei which is 3kT/2. The quantity on the TEMPn input card that MCNP expects is kT and not just T (see page 160). This quantity kT is the kinetic energy in MeV of the nuclei at their most probable velocity for temperature T. Note that kT is not a function of the particle mass and is therefore the kinetic energy at the most probable velocity for particles of any mass.

b. S(α,β) treatment:

The thermal scattering model generally referred to in this manual by the name S(α,β) is, in reality, a complete representation of thermal neutron scattering by molecules and crystalline solids. Two processes are allowed: (1) inelastic scattering with cross section σ_{in} and a coupled energy-angle representation derived from an ENDF/B S(α,β) scattering law, and (2) elastic scattering ($\Delta E=0$) for solids with cross section σ_{el} and an angular treatment derived from lattice parameters. For those materials with $\sigma_{el} \neq 0$, the scattering treatment is chosen with probability $\sigma_{el}/(\sigma_{el}+\sigma_{in})$. This thermal scattering treatment also allows the representation of scattering by multiatomic molecules (e.g., BeO).

For the inelastic treatment, the distribution of secondary energies is represented by a set of equally probable final energies (typically 16 or 32) for each member of a grid of initial energies from an upper limit of typically 4 eV down to 10^{-5} eV along with a set of angular data for each

initial and final energy. The selection of a final energy E′ given an initial energy E may be characterized by sampling from the distribution

$$p(E'|E_i < E < E_{i+1}) = \frac{1}{N}\sum_{i=1}^{N}\delta\left[E'-\rho E_{i,j}-(1-\rho)E_{i+1,j}\right]$$

where E_i and E_{i+1} are adjacent elements on the initial energy grid

$$\rho = \frac{E_{i+1} - E}{E_{i+1} - E_i}$$

N is the number of equally probable final energies, and E_{ij} is the j^{th} discrete final energy for incident energy E_i.

There are four allowed schemes for the selection of a scattering cosine following selection of a final energy and final energy index j. In each case, the $(i,j)^{th}$ set of angular data is associated with the energy transition $E = E_i \rightarrow E' = E_{i,j}$.

(1) The data consist of sets of P_3 Legendre coefficients, and μ is obtained by sampling from the continuous (and normalized) distribution constructed with the coefficients interpolated on incident energy linearly by parameter ρ between the (i,j) and (i+1,j) coefficient sets.

(2) The data consist of sets of double-P_1 discrete probabilities which are linearly interpolated with ρ to obtain probabilities for the five allowed scattering processes: forward isotropic ($0 \le \mu \le 1$), backward isotropic ($-1 \le \mu < 0$), $\mu = \pm 1$, and $\mu = 0$.

(3) The data consist of sets of equally probable discrete cosines $\mu_{i,j,k}$ for $k=1,\ldots,\nu$ with ν typically 4 or 8. An index k is selected with probability $1/\nu$, and μ is obtained by the relation

$$\mu = \rho\mu_{i,j,k} + (1-\rho)\mu_{i+1,j,k}$$

(4) The data consist of bin boundaries of equally probable cosine bins. In this case, random linear interpolation is used to select one set or the other, with ρ being the probablility of selecting the set correspoinding to incident energy E_i. The subsequent procedure consists of sampling for one of the equally probable bins and then choosing μ uniformly in the bin.

For elastic scattering, the above four angular representations are allowed for data derived by an incoherent approximation. In this case, one set of angular data appears for each incident energy and is used with the interpolation procedures on incident energy described above.

For elastic scattering, when the data have been derived in the coherent approximation, a completely different representation occurs. In this case, the data actually stored is the set of parameters D_k, where

$$\sigma_{el} = D_k/E \qquad \text{for } E_{Bk} \leq E < E_{Bk+1}$$
$$\sigma_{el} = 0 \qquad \text{for } E < E_{B1}$$

and E_{Bk} are Bragg energies derived from the lattice parameters. For incident energy E such that $E_{Bk} \leq E \leq E_{Bk+1}$,

$$P_i = D_i/D_k \text{ for } i = 1,\ldots,k$$

represents a discrete cumulative probability distribution which is sampled to obtain index i, representing scattering from the i^{th} Bragg edge. The scattering cosine is then obtained from the relationship

$$\mu = 1 - 2E_{Bi}/E$$

5. Elastic and Inelastic Scattering

If the conditions for a thermal collision are not met, then the particle undergoes either an elastic or inelastic collision. The selection of an elastic collision is made with probability

$$\frac{\sigma_{el}}{\sigma_{in} + \sigma_{el}} = \frac{\sigma_{el}}{\sigma_T - \sigma_a}$$

where σ_{el} is the elastic scattering cross section.
σ_{in} is the inelastic scattering cross section.
σ_a is the absorption cross section.
σ_T is the total cross section.

The selection of an inelastic collision is made with probability

$$\frac{\sigma_{in}}{\sigma_T - \sigma_a}$$

For both elastic and inelastic scattering the direction of exiting particles is usually determined by sampling angular distribution tables. This process will be described shortly. For elastic collisions the exiting

particle energy is then determined from two body kinematics based upon the center-of-mass cosine of the scattering angle. For inelastic scattering the energy of exiting particles is determined from secondary energy distribution laws from the cross-section files which vary according to the particular inelastic collision modeled.

a. Sampling of Angular Distributions: The direction of emitted particles is sampled in the same way for most elastic and inelastic collisions. The center-of-mass system cosine of the angle between incident and exiting particle directions, μ_{cm}, is sampled from angular distribution tables in the collision nuclide's cross-section library. The angular distribution tables consist of 32 equiprobable cosine bins and are given at a number of different incident particle energies. If E is the incident neutron energy, if E_n is the energy of table n, and if E_{n+1} is the energy of table n + 1, then a value of μ_{cm} is sampled from table n + 1 with probability $(E - E_n)/(E_{n+1} - E_n)$ and from table n with probability $(E_{n+1} - E)/(E_{n+1} - E_n)$. A random number ξ on the interval (0,1] is then used to select the i^{th} cosine bin such that $i - 1 < 32\ \xi \leq i$. The value of μ_{cm} is then computed as

$$\mu_{cm} = \mu_i + (32\ \xi - i)(\mu_{i+1} - \mu_i)$$

If no angular distribution table is given for the incident neutron energy (i.e., $E < E_1$) then μ_{cm} is chosen isotropically:

$$\mu_{cm} = \xi$$

where ξ is a random number on the interval [-1,1].

Once μ_{cm} is chosen, the incident particle direction cosines, (u_o,v_o,w_o), are rotated to new laboratory system cosines, (u,v,w), through a polar angle whose cosine is μ_{cm}, and through an azimuthal angle sampled uniformly. For random numbers ξ_1 and ξ_2 on the interval [-1,1] with rejection criterion $\xi_1^2 + \xi_2^2 \leq 1$, the rotation scheme is

$$u = u_o\mu_{cm} + \frac{\sqrt{1 - \mu_{cm}^2}\ (\xi_1\ u_o w_o - \xi_2 v_o)}{\sqrt{(\xi_1^2 + \xi_2^2)(1 - w_0^2)}}$$

$$v = v_o\mu_{cm} + \frac{\sqrt{1 - \mu_{cm}^2}\ (\xi_1\ v_o w_o + \xi_2 u_o)}{\sqrt{(\xi_1^2 + \xi_2^2)(1 - w_0^2)}}$$

$$w = w_o\mu_{cm} - \frac{\xi_1\sqrt{(1 - \mu_{cm}^2)(1 - w_0^2)}}{\sqrt{(\xi_1^2 + \xi_2^2)}}$$

If $1 - w_o^2 \sim 0$, then

$$u = \xi_1\sqrt{\frac{1 - \mu_{cm}^2}{\xi_1^2 + \xi_2^2}}$$

$$v = \xi_2\sqrt{\frac{1 - \mu_{cm}^2}{\xi_1^2 + \xi_2^2}}$$

$$w = \mu_{cm}$$

If the scattering angle is isotropic, often it is possible to use an even simpler formulation where advantage is taken of the fact that the exiting direction cosines, (u,v,w), are independent of the incident direction cosines, (u_o,v_o,w_o). In this case,

$$u = 2\xi_1^2 + 2\xi_2^2 - 1$$

$$v = \xi_1\sqrt{\frac{1 - u^2}{\xi_1^2 + \xi_2^2}}$$

$$w = \xi_2\sqrt{\frac{1 - u^2}{\xi_1^2 + \xi_2^2}}$$

b. Elastic Scattering: Elastic scattering is treated in two different ways depending upon the incident particle energy. If the incident energy is within the thermal range (e.g., below the thermal cut-in energy specified by the user on the ERGN card), the particle is scattered isotropically with no loss in energy. This collision is not with a thermal scattering isotope. If the incident energy is above the thermal range, the particle direction is sampled from the appropriate angular distribution tables, and the exiting energy, E_{out}, is dictated by two body kinematics:

$$E_{out} = \frac{1}{2} E_{in}\left[(1 - \alpha)\mu_{cm} + 1 + \alpha\right]$$

$$= E\left[\frac{1 + A^2 + 2A\,\mu_{cm}}{(1 + A)^2}\right]$$

where E_{in} = incident neutron energy
μ_{cm} = center-of-mass cosine of the angle between incident and exiting particle directions
A = mass of collision nuclide in units of the mass of a neutron (atomic weight ratio)

$$\alpha = \left(\frac{A - 1}{A + 1}\right)^2$$

c. Inelastic Scattering: The treatment of inelastic scattering differs from one inelastic reaction to another depending upon the particular inelastic reaction chosen. Inelastic reactions are defined as (n,x) reactions such as (n,n'), (n,2n), (n,f), (n,n'α) in which x includes at least one neutron. Each inelastic reaction, σ_x, is sampled with frequency σ_x/σ_{in}.

For many inelastic reactions, such as (n,2n), more than one neutron may be emitted for each incident neutron. The weight of each exiting particle is always the same as the weight of the incident particle. The energy of exiting particles is governed by various scattering laws which are sampled independently for each exiting particle from the cross-section

files. Which law is used is prescribed by the particular cross-section evaluation used. In fact, more than one law may be specified, and the particular one used at a particular time is decided with a random number. In an (n,2n) reaction, for example, the first particle out may have an energy sampled from one or more laws while the second particle out may have an energy sampled from one or more different laws depending upon specifications in the nuclear data library. Whereas the emerging energy and scattering angle are sampled independently there is no correlation between the emerging particles. Hence energy is not conserved in each individual reaction since, for example, a 14 MeV particle could conceivably produce two 12 MeV particles in a single reaction. But the net effect of many particle histories is unbiased since on the average the correct amount of energy is emitted. Results are biased only when quantities that depend upon the correlation between the emerging particles are being estimated.

The exiting particle energy and direction are always given in the laboratory coordinate system. However, the nuclear data in the cross-section library may be specified in either the laboratory or the center-of-mass system. Furthermore, the calculation of exiting energy and/or direction may be performed in either lab or center-of-mass system. If the data or calculations are in the center-of-mass coordinate system, the appropriate conversion to the lab system is always made.

d. Nonfission Inelastic Scattering: Nonfission inelastic reactions are handled differently from fission inelastic reactions. For each nonfission reaction N_p particles are emitted where N_p is an integer quantity specified for each reaction in the cross-section data library of the collision nuclide. The direction of each emitted particle is independently sampled from the appropriate angular distribution table as was described earlier. The energy of each emitted particle is independently sampled from one of six scattering laws:

Law 1 (ENDF law 1): Tabular energies out
Exiting energies are sampled from a table of equiprobable energies completely analogously to the sampling of scattering cosines (described in the "Sampling of Angular Distributions" section)

Law 3 (ENDF law 3): Level scattering

$$E_{out} = \left(\frac{A}{A+1}\right)^2 \left\{E_{in} - \frac{|Q|\ (A+1)}{A}\right\} \quad \text{cm system}$$

$$E_{out} = \frac{A^2+1}{(A+1)^2} \left\{E_{in} - \frac{|Q|\ (A+1)A}{A^2+1}\right\} \quad \text{lab system}$$

Law 5 (ENDF law 5): General evaporation spectrum

$$f(E_{in} \rightarrow E_{out}) = g\left(\frac{E_{out}}{T(E_{in})}\right)$$

This density function is sampled by
$E_{out} = \chi(\xi)\ T(E_{in})$
where $T(E_{in})$ is a tabulated function of the incident energy
$\chi(\xi)$ is a tabulated function of the random number ξ

Law 9 (ENDF law 9): Evaporation spectrum
$f(E_{in} \rightarrow E_{out}) = C^* E_{out} e^{-E_{out}/T(E_{in})}$
where $T(E_{in})$ is a tabulated function of the incident energy and E_{out} is restricted as $0 < E_{out} < E_{in} - U$ with U being the reaction threshold energy. In MCNP this density function is sampled by
$E_{out} = -T(E_{in})\ ln(\xi_1 \xi_2)$
where ξ_2 and ξ_2 are random numbers on the unit interval.

Law 21 (UK law 1): Discrete tabular energies out
Tables of P_{ij} and E_{ij} are given at a number of incident energies, E_i. If $E_i \le E_{in} < E_{i+1}$ then the i^{th} P_{ij} and E_{ij} tables are used. $E_{out} = E_{ik}$ where k is chosen according to

$$\sum_{j=1}^{k} P_{ij} < \xi \le \sum_{j=1}^{k+1} P_{ij}$$

and where ξ is a random number on the unit interval.

Law 22 (UK law 2): Tabular linear functions of incident energy out
Tables of P_{ij}, C_{ij}, and T_{ij} are given at a number of incident energies E_i. If $E_i \le E_{in} < E_{i+1}$ then the i^{th} P_{ij}, C_{ij} and T_{ij} tables are used.
$E_{out} = C_{ik}\ (E_{in} - T_{ik})$

where k is chosen according to

$$\sum_{j=1}^{k} P_{ij} < \xi \leq \sum_{j=1}^{k+1} P_{ij}$$

and where ξ is a random number on the unit interval.

Whereas laws 21 and 22 are merely different formats of law 1, there are really only 4 inelastic nonfission scattering laws governing exiting particle energy:

1. table lookup: $E_{out} = g(E_{in})$
2. level scattering: $E_{out} = C_1(E_{in} - C_2)$
3. general evaporation spectrum: $f(E_{in} \rightarrow E_{out}) = g\left(\frac{E_{out}}{T(E_{in})}\right)$
4. evaporation spectrum: $f(E_{in} \rightarrow E_{out}) = C_3 E_{out} e^{-E_{out}/T(E_{in})}$

e. Fission Inelastic Scattering: There are two ways of handling fission reactions: type 19 fission which is used by ENDF/B format data libraries and a pseudo-type 18 fission which is used by other format data libraries. For both fission formats, N_p particles are emitted according to the value of $\bar{\nu}(E_{in})$. The average number of neutrons per fission, $\bar{\nu}(E_{in})$, is either a tabulated function of energy or a polynomial function of energy. If I is the largest integer less than $\nu(E_{in})$, then

$N_p = I$ if $\xi \leq \nu(E_{in}) - I$

$N_p = I+1$ if $\xi > \nu(E_{in}) - I$ where ξ is a random number.

For type 19 fission the direction of each emitted neutron is sampled independently from the appropriate angular distribution table by the procedure described earlier. For type 18 fission the direction of each emitted neutron is sampled isotropically.

For type 19 fission the energy of each emitted neutron is determined from the appropriate scattering law. In addition to the laws described earlier for nonfission inelastic scattering, three more fission scattering laws are available:

Law 7 (ENDF law 7): Simple Maxwell Fission Spectrum

$$f(E_{in} \rightarrow E_{out}) = C_0 * \sqrt{E_{out}}\, e^{-E_{out}/T(E_{in})}$$

where $T(E_{in})$ is a tabulated function of incident energy, E_{in}, and E_{out} is restricted by $0 \leq E_{out} < E_{in} - U$ with U being the reaction threshold energy. In MCNP this density function is sampled by the rejection scheme

$$E_{out} = -T(E_{in}) \left[\frac{\xi_1^2 \ln \xi_3}{\xi_1^2 + \xi_2^2} + \ln \xi_4 \right]$$

where ξ_1, ξ_2, ξ_3, and ξ_4 are random numbers on the unit interval. ξ_1 and ξ_2 are rejected if $\xi_1^2 + \xi_2^2 > 1$.

Law 10 (ENDF law 10): Watt Spectrum

$$f(E_{in} \rightarrow E_{out}) = C_0\, e^{-E_{out}/a} \sinh \sqrt{b\, E_{out}}$$

The quantities b, L, and M are given in the nuclear data library where

$$L = a(k+\sqrt{k^2-1})$$
$$M = k+\sqrt{k^2-1} - 1$$
$$k = 1 + ab/8$$

For random numbers ξ_1 and ξ_2 on the unit interval the density function is sampled by the rejection scheme
$E_{out} = -L \ln \xi_1$
where E_{out} is accepted whenever
$[-\ln \xi_2 - M(1 - \ln \xi_1)]^2 \leq - bL \ln \xi_1$

Law 11 (ENDF law 11): Energy Dependent Watt Spectrum

$$f(E_{in} \rightarrow E_{out}) = C_0 \, e^{-E_{out}/a(E_{in})} \sinh \sqrt{b\,(E_{in})\,E_{out}}$$

This law is sampled in the same way as Law 10 except that b, L, and M are now dependent upon the incident energy and are looked up in a table. (Law 11 applies to ENDF/B-V.)

For type 18 fission special MCNP fission routines are invoked by calling ficticious law 18. These routines use the probability density functions and sampling schemes of laws 7 and 9. However, the data for these laws come from data statements in the MCNP code rather than from the collision nuclide cross-section data library.

Once the energy and direction of the emerging particles are determined, MCNP proceeds to model the transport of the first particle out after stashing all other particles in the bank. Type 18 fission is an exception since the exiting energies and direction for all but the first emerging particle are calculated when these later particles are retrieved from the bank rather than before they are put in the bank.

B. Photons

The photon portion of MCNP is composed of two main routines: (1) the MCG simple physics treatment that assumes photoelectric absorption is a terminal event but models it by implicit capture, and (2) the MCP detailed physics treatment that accounts for fluorescent photons after photoelectric absorption. The MCP routine is the default, but the user can make a choice on the ERGP card. The following discussion is based primarily on Reference 8.

1. MCG

The physical processes treated are photoelectric effect, pair production, and Compton scattering on free electrons. Since this treatment is intended primarily for higher energy photons, the photoelectric effect is regarded as an absorption (without fluorescence), scattering (Compton) is regarded to be on free electrons (without use of form factors), and the

highly forward coherent Thomson scattering is ignored. Thus the total cross section σ_t is regarded as the sum of three components:

$$\sigma_t = \sigma_{pe} + \sigma_{pp} + \sigma_s$$

a. Photoelectric Effect: This is treated as a pure absorption by implicit capture with a corresponding reduction in the photon weight WGT, and hence does not result in the loss of a particle history. On every collision, the weight $WGT \cdot \sigma_{pe}/\sigma_t$ and energy $E \cdot WGT \cdot \sigma_{pe}/\sigma_t$ are tallied in the appropriate bins. The non-captured weight $WGT(1 - \sigma_{pe}/\sigma_t)$ is then forced to suffer either pair production or Compton scattering with the proper dependent probabilities.

b. Pair Production: In a collision resulting in pair production [probability $\sigma_{pp}/(\sigma_t - \sigma_{pe})$], it is assumed that the kinetic energy $WGT(E - 1.022)$ MeV of the electron-positron pair produced is deposited as thermal energy at the time and point, with isotropic production of one gamma of energy 0.511 MeV, and weight $2 \cdot WGT$. This 0.511 MeV photon is then followed to termination.

c. Compton Scattering: The alternative to pair production (when both are possible) is Compton scattering on a free electron, with probability $\sigma_s/(\sigma_t - \sigma_{pe})$. In the event of such a collision, the objective is to determine the energy E' of the scattered photon, and $\mu = \cos\theta$ for the angle θ of deflection from the line of flight. This yields at once the energy $WGT(E - E')$ deposited at the point of collision and the new direction of the scattered photon.

The differential cross section for the process is given by the Klein-Nishina formula

$$K(\alpha,\mu)d\mu = \pi r_o^2 \left(\frac{\alpha'}{\alpha}\right)^2 \left[\frac{\alpha'}{\alpha} + \frac{\alpha}{\alpha'} + \mu^2 - 1\right] d\mu$$

where r_o is the classical electron radius, α and α' are the incident and final photon energies in units of 0.511 MeV [$\alpha = E/(mc^2)$, where m is the mass of the electron and c is the speed of light], and

$$\alpha' = \alpha/[1 + \alpha(1 - \mu)] .$$

Changing variables from μ to $x = 1/[1 + \alpha(1 - \mu)]$ on $\delta \equiv (1 + 2\alpha)^{-1} \leq x \leq 1$, one finds the probability density function for x to be

$$p(x) = g(x)/G(\delta) ,$$

where $g(x) = x + x^{-1} + \mu^2 - 1$, $\mu = 1 + \alpha^{-1} - (\alpha x)^{-1}$, and $G(x) = \int_x^1 g(x)dx$. Thus a random number ξ determines x by the implicit relation

$$\xi = G(x)/G(\delta)$$

and consequently the required $\mu = 1 + \alpha^{-1} - (\alpha x)^{-1}$ and $\alpha' = \alpha x$, $E' = 0.511\ \alpha'$.

An accurate approximation for the inverse $x = H(y)$ of the function $y = G(x)$ allows rapid determination of $x = H[\xi G(\delta)]$.

2. MCP

The MCP treatment, photons of energies 1 keV to 100 MeV, provides for fluorescent emission and the modification of Thomson and Klein–Nishina differential cross sections by appropriate form factors which take electron binding effects into account.

A cross-section library MCPLIB has been prepared, incorporating all constants required by this treatment, for elements Z = 1, ..., 94, in a form designed to expedite computation.

a. Free Path: MCPLIB contains, for each nuclide, a table of the logarithms $L_i(Z) = ln[E_i(Z)]$ of suitable energies, including the photoelectric edges, and a matrix $L^j_i(Z) = ln\ \sigma^j_i(Z)$, listing for j = 1, 2, 3, 4, the logarithms of corresponding cross sections (when the latter are non-zero) for incoherent scattering, coherent scattering, photoelectric effect, and pair production, respectively. The compilation of data by Storm and Israel was used for all listed energies $E_i \leq 100$ MeV. In the case of scattering (j = 1,2) the cited total cross sections were obtained by numerical integration, based on the same form factors used in the Monte Carlo treatment of such collisions, and referred to below.

In the collision routine, a photon of energy E, starting from a point in a particular medium, has a free path

$$\lambda = \frac{1}{\sum_{z} N(z) \sum_{1}^{4} \sigma^{j}{}_{i}(z)}$$

where Z runs over all elements present in the medium, N(Z) is the corresponding numerical density, and $\sigma^j(Z)$ is the cross section for process j, each log-log interpolated to energy E. A random number ξ on (0,1) then determines the (infinite-medium) distance to collision, $d = -\lambda \ln \xi$.

In the event of collision, two random numbers ξ_1 and ξ_2 serve to designate the element Z hit, and the process j responsible. The former results from a comparison of ξ_1/λ with the partial Z-sums obtained above, and present in the memory. The latter process j is determined by a similar comparison of

$$\xi_2 \sum_{1}^{4} \sigma^j(z)$$

with the partial sums involved, the individual $\sigma^j(Z)$ being also retained from the λ computation.

b. Note on Interpolation: Log-Log interpolation for the partial cross sections σ^j, at an energy E between tabulated energies $E_{i-1} < E_i$, leads to the result

$$\sigma^j = (\sigma^j{}_{i-1})^a \, (\sigma^j{}_i)^b \ ,$$

where $a = (\ln E_i - \ln E)/(\ln E_i - \ln E_{i-1})$, $a + b = 1$, $a,b > 0$. It is expedient to regard as the total cross section and as the probability of process j at energy E, the values of σ and σ^j/σ, where σ is the sum $\sum_{j=1}^{4} \sigma^j$ of the σ^j so found, and not the log-log interpolated value σ' of the total cross section. For, the relation

$$\sigma = \sum_{j} \sigma^j$$

$$= \sum_j \left(\sigma^j_{i-1}\right)^a \left(\sigma^j_i\right)^b$$

$$\leq \left(\sum_j \sigma^j_{i-1}\right)^a \left(\sum_j \sigma^j_i\right)^b = \sigma'$$

is an obvious consequence of Hölder's inequality,

$$\sum x_i y_j \leq \left(\sum x_j^{1/a}\right)^a \left(\sum y_j^{1/b}\right)^b$$

which is strict unless $y_j^{1/b} \equiv kx_j^{1/a}$.

Hence, in practice one has $\sigma < \sigma'$, and use of σ' in place of σ may lead to absurdities; e.g., pair production, determined above by default (after the other three processes are tested), would occur at all energies $E \geq 1$ keV.

This shows that adoption of log-log interpolated <u>partial</u> cross sections is inconsistent with a log-log interpolated <u>total</u> cross section.

<u>c. Incoherent Scattering</u>: The objective, in the event of such a process ($j = 1$), is to determine the angle θ of scattering from the incident line of flight (and thus the new direction), the new energy E' of the photon, and the local energy deposition $E - E'$ (the recoil kinetic energy of the electron).

Incoherent scattering is assumed to have the differential cross section $\sigma^1(Z,\alpha,\mu)\, d\mu = I(Z,\nu)\, K(\alpha,\mu)\, d\mu$, where $I(Z,\nu)$ is an appropriate scattering factor, modifying the Klein-Nishina cross section

$$K(\alpha,\mu)d\mu = \pi r_o^2 \left(\frac{\alpha'}{\alpha}\right)^2 \left[\frac{\alpha'}{\alpha} + \frac{\alpha}{\alpha'} + \mu^2 - 1\right] d\mu$$

As is customary, α and α' denote the incident and scattered photon energies, respectively, in units of electron rest energy mc^2,

$$\alpha' = \alpha/[1 + \alpha(1 - \mu)],$$

$\mu = \cos\theta$, and $r_0 = e^2/mc^2 = 2.81776 \times 10^{-13}$ cm, the "classical electron radius."

Qualitatively, the effect of I(Z,v)/Z is to decrease the Klein–Nishina cross section (per electron) more extremely in the forward direction, for low E and for high Z independently. For any Z, I(Z,v) increases from I(Z,0) = 0 to I(Z,∞) = Z. The parameter v = v(α,μ) is a given function of α and μ which, for a particular incident energy α, increases from v(α,1) = 0 at μ = 1 to a maximum value $\bar{v}$ = v(α,−1) at μ = −1. The

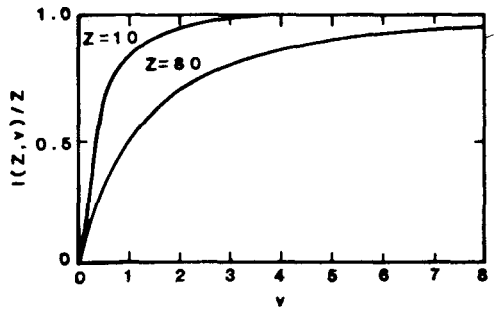


Figure 2.8

essential features of I(Z,v) are indicated in Figure 2.8.

Having obtained μ and α′ = αx from the direct evaluation of an inverse function rather than from a rejection technique, the final energy of the photon is E′ = $mc^2\alpha'$, and one deposits the energy E−E′ locally.

For the point detector routine of the general code, one requires, for a given μ (determined by the detector position), the probability of (incoherent) scattering to the angular range (μ,μ + dμ), $p^1(\mu)d\mu = I(Z,v)K(\alpha,\mu)d\mu/\sigma_t^1(Z,\alpha)$. The values of πr_0^2 and of $\alpha'/\alpha = 1/[1 + \alpha(1 - \mu)]$ are needed in K(α,μ); I(Z,v) is obtained by linear interpolation at the computed value of v = v(α,μ); and $\sigma_t^1(Z,\alpha) = \sigma^1(Z)$, at the incident energy E, is recoverable from the free path routine.

d. Coherent Scattering: This process (j = 2) involves no energy loss, only the scattering angle θ being required.

The differential cross section is now $\sigma^2(Z,\alpha,\mu)d\mu = C^2(Z,v)T(\mu)d\mu$, where C(Z,v) is a form factor modifying the (energy independent!) Thomson cross section $T(\mu) = \pi r_0^2(1 + \mu^2)d\mu$. (Superscripts on σ's denote process number j, not an exponent.)

The general effect of $C^2(Z,v)/z^2$ is to decrease the Thomson cross section, more extremely for backward scattering, high E, and low Z, being opposite in these respects to the effect of I(Z,v)/Z on K(α,μ) in Section c above. The a given Z, C(Z,v) decreases from C(Z,0) = Z to C(Z,∞) = 0. The

parameter is here the $v = K\alpha\sqrt{1-\mu}$ of that section, with maximum $\bar{v} = \sqrt{2}\,K\alpha$ for

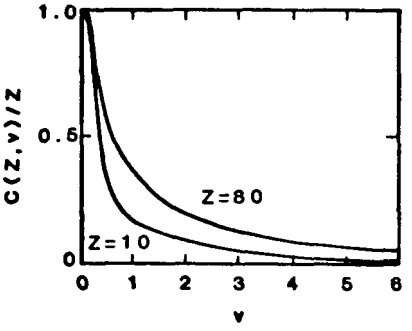


Figure 2.9

given α. Tne qualitative features of C(Z,v) are shown in Figure 2.9.

For the point detector routine, one must evaluate the density $p^2(\mu) = \pi r_0^2(1 + \mu^2)\ C^2(Z,v)/\sigma_t^2(Z,\alpha)$ for given μ. Although $\sigma_t^2(Z,\alpha) = \sigma^2(Z)$ is recoverable from the λ routine, the value of $C^2(Z,v)$ at $v = K\alpha$ SQRT(1 − μ) must be interpolated in the original $C^2(Z,v_i)$ tables, separately stored on MCPLIB for this purpose.

e. Photoelectric Effect: A collision of this type (j = 3) involves the disappearance of the incident photon of energy E, the ejection from some written energy level $e \leq E$ of an orbital electron with kinetic energy E−e, and the transition of a second electron from a level $e' < e$ to the e−level vacancy. There are two possibilities.

(1) Up to two fluorescence photons of energy $E' = e-e'$ may be emitted. In such a case, the photon energy difference $E - E' = (E - e) + e'$ consists of the kinetic energy of the first ejected electron, plus a residual excitation energy e′ which is ultimately dissipated by further processes, with additional fluorescence of still lower energy. This is ignored, depositing all of E−E′ locally, and returning to the mean−free−path routine with the (isotropically emitted) fluorescence photon of energy E′, provided of course that $E' \geq 1$ keV. Otherwise the event is "terminal," by which is meant that the incident photon's history terminates, its energy E being locally deposited.

(2) The electron transition $e' \rightarrow e$ may not be accompanied by $E' = e - e'$ fluorescence, but by the ejection of an Auger electron, resulting from internal conversion. In this event, the entire incident energy E is tallied as energy deposition, and the collision is terminal.

The energy levels e are called "edge energies" because, regarded as a function of increasing E, the photoelectric cross section $\sigma(E)$, elsewhere

decreasing continuously, shows a sharp discontinuity (edge) at each E = e, jumping from its lower, limiting value $\sigma(e^-)$ to its value $\sigma(e) > \sigma(e^-)$ as the photon energy E becomes sufficient to activate the e–level.

A photoelectric event is regarded as terminal for elements $Z < 12$, the possible fluorescence energy being below 1 keV. For elements $Z \geq 12$, fluorescent emission above 1 keV is possible and allowed for.

f. Pair Production: This process (j = 4) is considered only in the field of a nucleus. Although the threshold is technically $2mc^2[1 + (m/M)] \cong 1.022$ MeV, M being the nuclear mass, $\sigma_i^4(Z)$ becomes positive only for $E \geq 1.5$ MeV in the tables used.

In the event of such a collision, the incident photon, of energy E, vanishes; the kinetic energy of the created positron–electron pair, assumed to be $E - 2mc^2$, is deposited locally at the collision point; the positron is considered to be annihilated with an electron at the point of collision; and a single photon of weight twice that of the incoming photon and energy mc^2 is given a new isotropic direction and is transported further.

g. Energy Range: If all other effects (bremsstrahlung, etc.) are ignored, nothing prevents extension of the code to the upper limit (100 MeV) of the Storm–Israel tables, since the approximation for the inverse to the Klein–Nishina scattering distribution remains good to that energy. Below 0.001 MeV, photons are allowed to have only analog capture which quickly eliminates them because of the very large photoelectric cross section.

V. TALLIES

MCNP provides six standard neutron tallies and five standard photon tallies, but these basic tallies can be easily modified by the user almost any way desired. All tallies are normalized to be per starting particle unless modified by the user. They are also a function of energy and time (plus direction in the case of current). Finally all standard tallies are integrated over the azimuthal angle φ from 0 to 2π.

Tally Mnemonic	Description
F1 and F11	Current across any designated subset of surfaces in the problem (units: particles)
F2 and F12	Flux across any designated subset of surfaces in the problem (units: particles/cm^2)
F4 and F14	Track length per unit volume, or average flux, in any designated subset of cells (units: particles/cm^2)
F5a and F15a	Flux at a designated set of points or spatial rings symmetric about the a = x,y, or z axis (units: particles/cm^2)
F6 and F16	Track-length estimate of neutron energy deposition (or charged particle heating) including fission (F6), or photon energy deposition (F16) for any designated subset of cells (units: MeV/gm)
F7	Track-length estimate of energy deposition due to fissions in any designated subset of cells. This tally is a subset of Tally F6. (units: MeV/gm)

The above six tally categories represent the basic MCNP tally types. Photon tallies are identified by a number that is ten higher than the equivalent neutron tally (but there is no F17 tally). Any basic tally number may be changed by increments of 20 up to three digits; for example, F1 and F221 or F11 and F131.

Thought should be given to the application of these tallies and to the comparison of one tally with another. For example, if the flux is varying as $1/R^2$ in a cell, an average flux in the cell determined by the F4 tally will be higher than the flux at a point in the center of the cell determined by a detector. This same consideration applies to the average flux provided by DXTRAN spheres (see page 95).

Standard summary information to give the user a better insight into the physics of his problem and the adequacy of the Monte Carlo simulation includes a complete accounting of the creation and loss of all tracks and their energy; the number of tracks entering and re-entering a cell plus the track population in the cell; the number of collisions in a cell; the average weight, mean free path, and energy of tracks in a cell; the activity of each nuclide in a cell; a complete weight balance for each cell; plus more.

The following MCNP definitions of current and flux come from reactor theory but are related to similar quantities in radiative transfer theory. The MCNP particle angular flux multiplied by the particle energy is the same as the intensity in radiative transfer theory. The MCNP particle total flux at energy E multiplied by the particle energy is equivalent to the integrated energy density times the speed of light in radiative transfer theory, and the MCNP particle current multiplied by the particle energy is analogous to the radiative flux crossing an area in radiative transfer theory. The MCNP particle current uses $|\mu|$ in the definition, while the radiative transfer flux uses μ in its definition. MCNP current is not net current nor a positive nor a negative current; it is the number of particles crossing a surface in a particular direction. The MCNP particle fluence multiplied by the particle energy is the same as the fluence in radiative transfer theory.

A. Current and Flux across a Surface

It will be useful to review the definitions of flux, current, and the relation between the two to help clarify what MCNP calculates.

Consider a one-speed, steady-state problem with particles all of unit

weight and velocity v crossing surface S in the increment dS as in

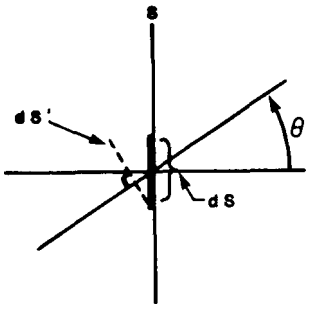


Figure 2.10

Figure 2.10.

Let

$$N(\bar{r},\mu,\varphi)\ d\mu\ \frac{d\varphi}{2\pi}$$

be the number of particles N per unit volume at the point $\bar{r}$ traveling with angles θ and φ in $d\mu$ and $d\varphi$ where $\mu = \cos\theta$ (θ is the polar angle), and φ is the azimuthal angle. The number of particles crossing dS′ (which is an area perpendicular to dS) is the same as the number of particles crossing dS. The number of particles crossing dS′ per second with angles in $d\mu$ and $d\varphi$ is

$$N(\bar{r},\mu,\varphi)\ Vd\mu\ \frac{d\varphi}{2\pi}\ dS'$$

However, $dS' = \mu dS$ so that the number of particles crossing dS per second with angles in $d\mu$ and $d\varphi$ is

$$N(\bar{r},\mu,\varphi)\ v\mu d\mu\ \frac{d\varphi}{2\pi}\ dS$$

This is the MCNP F1/11 current tally if μ is replaced by $|\mu|$. Note that scoring unity at each crossing between μ_1 and μ_2 through a surface S has the mean

$$\int_S dS \int_{\mu_1}^{\mu_2} d\mu \int_0^{2\pi} \frac{d\varphi}{2\pi} N(\bar{r},\mu,\varphi)v\mu \qquad (2.2)$$

which is the current between μ_1 and μ_2.

The particle angular flux is defined as $\Phi(\bar{r},\mu,\varphi) = N(\bar{r},\mu,\varphi)v$. The scalar or total flux Φ at $\bar{r}$ is defined as

$$\int_{-1}^{1} d\mu \int_0^{2\pi} \frac{d\varphi}{2\pi} N(\bar{r},\mu,\varphi)v$$

The scalar flux has units of particles/cm^2. The integral of Φ over a time interval is called fluence. The fluence can be thought of as track length because it is the total distance traveled during the time interval by all of the particles in a unit volume. The flux integrated over a surface S is

$$\int_S dS \int_{-1}^{1} d\mu \int_0^{2\pi} \frac{d\varphi}{2\pi} N(\bar{r},\mu,\varphi)v \qquad (2.3)$$

Equations (2.2) and (2.3) are the same except that Equation (2.2) has a μ in the integrand and the range of integration over $d\mu$ is not from -1 to 1. Note that scoring $1/\mu$ at each crossing in Equation (2.2) and integrating over all μ has the mean

$$\int_S dS \int_{-1}^{1} d\mu \int_0^{2\pi} d\varphi \left\{N(\bar{r},\mu,\varphi)v\mu\right\} \frac{1}{\mu}$$

which is just the flux integrated over S.

More specific to MCNP, the flux tallies are really fluence since they are integrated over time intervals:

$$F2,4,or5 = \int_{-1}^{1} \int_{t_{m-1}}^{t_m} \int_{E_{n-1}}^{E_n} \phi(E,t,\mu)dE\ dt\ d\mu$$

where μ is the cosine of the trajectory's angle, t_m are the user-specified time boundaries, and E_n are the user-specified energy boundaries.

The current tally F1/11 is the number of particles crossing a surface and is related to the surface flux tally F2/12 by

$$F1 = \int_A \int_{\mu_{l-1}}^{\mu_l} \int_{t_{m-1}}^{t_m} \int_{E_{n-1}}^{E_n} |\mu| \phi(E,t,\mu)\, dE\, dt\, d\mu\, dA$$

where A is the surface area and μ_l are the user-specified cosine bin boundaries. Note that for neutrons this differs from the net neutron current of nuclear engineering reactor theory because $|\mu|$ rather than μ is used.

MCNP determines the F1/11 current in the TALLY subroutine by summing the weights WGT of tracks crossing a specified surface in the cosine bins defined on the C1/11 input card.

The MCNP tally sequence begins by setting the tally score T to the weight of the scoring track in the TALLY subroutine. The score is then modified in the WTMULT subroutine if there is an FM card. Back in TALLY, a call to TALLYX is made to allow any user modifications if an FU card is present. The score is next sorted into the appropriate angular bin and then summed with the previous scores. Final norming (i.e., for the EM, TM, CM cards, etc.) is done at the end of the job in the TALLYP subroutine.

The F2/12 surface flux tally is determined simply by dividing the F1/11 score by the cosine of the polar angle between the track's trajectory and the normal to the surface crossed. MCNP sets a lower value of this cosine to 0.1 to avoid a singularity. If the cosine is less than 0.1, it is reset to 0.05 which can still give rise to relatively large contributions to the tally. The sequence is almost identical to the above current tally. The cosine division is done just before the summing with previous scores in TALLY. The division by surface area is done in TALLYP.

Under certain circumstances these tallies may be related to each other. In the case of a point isotropic source at the center of a hollow sphere (where there are no collisions and $\mu = 1$), the F1 and F2 tallies on the surface of the sphere should be equal if the F1 result is divided by the surface area of the sphere.

B. Average Flux in a Cell

For the F4/14 cell flux tally, the total track length of all tracks in a designated cell is determined. Dividing this by the cell volume results in the average flux (actually fluence) in the cell. The track-length estimate of flux is in general quite reliable since there is frequently a large number of tracks in a cell (compared to the number of collisions) leading to a large number of contributions to this tally.

The F4/14 cell tally sequence is similar to that of the surface tallies. In particular, the score T is initially equated to the track weight WGT. Next calls to WTMULT (where any tally multiplier cards are taken into account) and TALLYX are made, and then the score is multiplied by the track's velocity VEL (if it is an energy deposition tally by the velocity times the energy deposition H). Finally in subroutine TALLY the score is multiplied by (T2-T1) which is the length of time the track is in the cell or the time between collisions. At this point the score T is the track length (unless the user has modified it). In the TALLYP subroutine a division by cell volume to convert to flux plus final norming is done.

C. Flux at a Detector

All detector schemes may yield anomalous statistics and must be used with caution. The following discussion should be read before using a detector.

The F5a/15a detector flux tally provides for the estimation of flux by three methods at a prescribed set of distinct points or rings in space: (1) The analog point detector should be used in regions of little scattering or in voids. (2) The ring detector flux is the average flux at a point on a ring. The ring detector (which is based on the analog detector) can reduce running time significantly from a point detector calculation for all types of problems having sufficient symmetry. (3) The once-more-collided flux estimator (OMCFE) point detector is suited for use in a scattering medium and should not be used elsewhere since the analog detector will work just as well and is cheaper to use.

Contributions to a detector are not made through or in a region of zero importance.

The contribution from source particles (the uncollided flux) is tabulated separately in addition to the total flux at detectors. If a user-provided source subroutine is required for a problem and the source is not isotropic, the user needs to specify the probability density function

PSC for emitting a particle directly at the detector in a user-provided subroutine, SRCDX. PSC must be defined for source particles for each detector used. Looking at the uncollided flux results and doing some hand calculations is a convenient way to verify that the PSC values were correctly set. See Chapter 4 for cautions and examples of SRCDX.

Although the source contribution to a detector could usually be calculated analytically, MCNP does it by determining the individual contribution to each detector for each source particle as it is emitted.

Basically, MCNP calculates the contribution to a detector directly from the source plus the probability of a contribution to a detector every time a particle has a collision. This assumes that the particle will reach the detector from the collision point with no more collisions and is called the last-flight estimator. The particle does not actually get to the prescribed point. This statistical estimation at each collision is made by computing the probability the particle will scatter at just the correct angle to hit a point in the center of a unit area at the detector location normal to the line joining the collision point and the detector. These individual contributions at each collision are summed over the life of the particle for a total contribution to the detector.

This MCNP detector scheme is also commonly called the next-event estimator (or analog estimator) which has an $1/R^2$ singularity (R is the distance between collision and detector) for a detector in a scattering medium which makes the theoretical variance of the estimator infinite even through the first moment is finite. The technique is still valid and unbiased but convergence is slower than if the variance were finite.

The once-more collided detector has a 1/R singularity and the ring detector also exhibits a 1/R behavior. The variance of these two schemes is therefore finite. One would like to have as many collisions in close proximity to the detector by particles of the same weight as possible, because a large-weighted particle interacting arbitrarily close to the detector (by the $1/R^2$ singularity) can severely perturb both the detector result and its variance. The two methods do indeed sample the region near the detector point better.

Any use of the detectors should be done with the PRDMP and/or DD card (see pages 165 and 152) to watch how the result and relative error are varying as a function of the number of histories.

The detector diagnostics (DD) input card provides information about isolated, large-weighted contributions to a detector. Studying this information can give insight into the appropriateness of using a detector

in the first place or how well (or poorly) you chose parameters like R_o or source biasing for it.

For an example of such problems with a detector, consider the concrete shell problem illustrated in Chapter 5. For 100000 histories the neutron flux at an analog point detector on the outer surface had a relative error of 23% but was lower than the true known value by almost a factor of two. Examination of the detector variance every ten thousand histories with the PRDMP input card revealed large fluctuations in the variance and hence the results could not be believed. Furthermore, results from the DD input card indicated that there were just not enough particles having interactions close to the detector point but a few large-weighted particles were interacting very close to the detector. This means that the region of space that was the most important contributor to the result was not being sampled well. A ring detector produces much better results in this particular problem.

As an exercise in the basic calculation of flux at a detector and in determining PSC for an SRCDX subroutine, suppose one wishes to estimate the contribution to the flux at a point in space from a particle entering a collision, scattering, and reaching the detector with no further collisions. The same considerations apply to the contribution to the flux at a point from a source supplied in a Monte Carlo problem or from an (n,xn) event, the emitted source particle and the scattered particle behaving in the same manner. A simple expression may be derived for the contribution as follows: suppose $p(\nu,\varphi)d\Omega$ is the probability of the particle scattering or being born into the solid angle $d\Omega$ about the direction (ν,φ). If R is the distance to the detector from the collision or source point, then

$$p(\theta,\varphi)d\Omega \cdot e^{-\int_0^R \Sigma_t(s)ds}$$

yields the probability of scattering into $d\Omega$ about (ν,φ) and arriving at the detector point with no further collisions. This situation is illustrated in Figure 2.11. The attenuation of a beam of monoenergetic particles passing through a material medium is given by $\exp[-\int_0^R \Sigma_t(s)ds]$ where s is measured along the direction from the collision or source point to the detector and $\Sigma_t(s)$ is the macroscopic total cross section at s.

This expression may be altered somewhat. If dA is an element of area

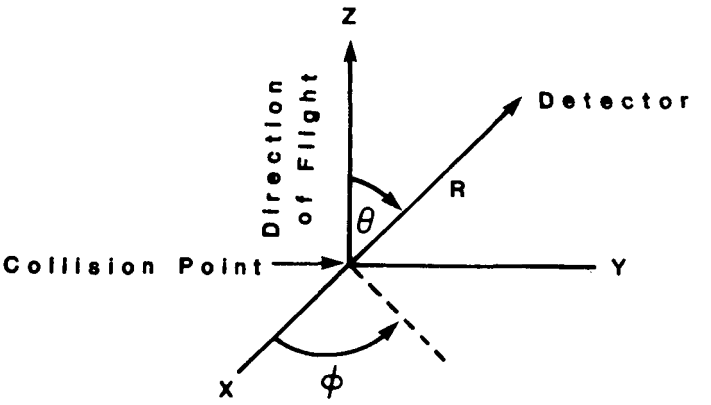


Figure 2.11

normal to the scattered line of flight to the detector, $d\Omega = dA/R^2$ and therefore

$$p(\theta,\varphi)\ \frac{dA}{R^2}\ e^{-\int_0^R \Sigma_t(s)ds}$$

is the expression giving the probability of scattering toward the detector and passing through the element of area dA normal to the line of flight to the detector. Since the contribution to the flux is by definition the number of particles passing through a unit area normal to the scattered direction, the general expression for the contribution to the flux is given by

$$\frac{p(\theta,\varphi)}{R^2}\ e^{-\int_0^R \Sigma_t(s)ds}$$

In all the MCNP scattering distributions and in most (but not all) of the standard sources, the density function $p(\theta,\varphi)$ is uniform (or isotropic) in φ – another way of saying this is that the angles θ and φ are independent variables so that $p(\theta,\varphi)$ can be written as the product of a probability density function in θ times a probability density function in φ. Illustrating this,

$$\int_\Omega p(\theta,\varphi)d\Omega = \int_0^{2\pi}\int_0^{\pi} p(\theta,\varphi)\ \sin\theta\ d\theta\ d\varphi$$

$$= \int_0^{2\pi} \int_{-1}^{1} p(\mu,\varphi) d\mu \; d\varphi$$

Now, since μ and φ are independent, $p(\mu,\varphi) = p(\mu)/2\pi$. Here $p(\mu) = \int_0^{2\pi} p(\mu,\varphi)d\varphi$, the marginal density function in μ and therefore $\int_{-1}^{1} p(\mu)d\mu = 1$. Clearly $d\varphi/2\pi$ is the uniform density of φ [which is the marginal density of φ, given by $\int_{-1}^{1} p(\mu,\varphi)d\mu = \int_{-1}^{1} p(\mu)\,\Phi(\varphi)\,d\mu = \Phi(\varphi) =$ constant. But $\int_0^{2\pi} \phi(\varphi)d\varphi = \phi(\varphi)\cdot 2\pi = 1$ so $\Phi(\varphi) = 1/2\pi$].

Now if $p(\theta,\varphi) = p(\mu,\varphi) = p(\mu)/2\pi$ is substituted in the expression for the flux, the expression used in MCNP is arrived at:

$$\frac{p(\mu)}{2\pi R^2} \, e^{-\int_0^R \Sigma_t(s)ds}$$

The quantity $p(\mu)$ is a probability density function in μ so that $\int_{-1}^{1} p(\nu)d\nu = 1$. The only case when a user must provide his own values for $p(\mu)$ is for the direct (unscattered) contribution to a detector or DXTRAN sphere from a user-specified source with anisotropic angular distribution. MCNP assumes that particle emission from a user-defined source is isotropic and has no way to calculate anisotropies in the source. Another subroutine, SRCDX, is provided by the user to specify $p(\mu)$ for <u>each</u> detector.

The quantity most often specified in SRCDX is PSC which is $p(\mu_o)$ where μ_o is the cosine of the angle between the direction defining the polar angle for the subroutine SOURCE and the direction to a detector or DXTRAN sphere point in the laboratory system. MCNP includes the 2π in the previous expression automatically. Other quantities may also need to be specified in SRCDX. A discusssion of these, user cautions, and examples of three actual SRCDX subroutines are given in Chapter 4.

Four different source examples are given below for calculating $p(\mu)$. PSC is then $p(\mu_o)$ for any $\mu=\mu_o$. PSC must always be non-negative and may be greater than one.

(1) The above considerations can be illustrated by using the important but simple example of an isotropic source. That is done automatically in the code and requires nothing from the user. The function $p(\mu,\varphi)\,d\Omega$ can now be written

$$p(\mu,\varphi)\ d\Omega = \frac{d\Omega}{4\pi} = \frac{d\mu d\varphi}{4\pi}$$

$$p(\mu)d\mu = d\mu \int_0^{2\pi} \frac{d\varphi}{4\pi} = \frac{1}{2}\, d\mu$$

Hence $p(\mu) = 1/2$.

(2) Another interesting case is that of a cosine source distribution in half-space. Here

$$p(\mu,\varphi)\ d\Omega = 2\mu d\mu \cdot \frac{d\varphi}{2\pi}$$

Integrating over φ yields $p(\mu) = 2\mu$. Actually one can often ignore the angle φ (and can always do so where the distribution of φ is uniform in φ); e.g., a cosine distribution means that the density of particles between μ and $\mu + d\mu$ is proportional to $\mu d\mu$. Therefore, $p(\mu)\ d\mu = k\mu d\mu$ and since $\int_0^1 k\mu d\mu = 1$, $k=2$. Hence $p(\mu) = 2\mu$.

(3) As a very important example in practice let us consider the case where the source information is given in the form of a table. Suppose the data are given as frequencies versus angular bins. Without loss of generality it can be assumed that ω_i is the probability of being emitted in the i^{th} angular bin (μ_{i-1},μ_i) and that the probability density function $p(\mu)$ is constant in μ within the bin (μ_{i-1},μ_i). Suppose i runs from 1 to N. The data can be represented in graphical form as a step-function, $p(\mu)$ versus μ, where $p(\mu) = \omega_i/(\mu_i - \mu_{i-1})$ in the interval (μ_{i-1},μ_i). In order to sample for μ the cumulative distribution function $P(\mu)$ is formed where

$$P(\mu) = \int_{-1}^{\mu} p(\mu)\ d\mu$$

and $p(\mu)$ is the probability density function in μ. Here $\mu_o = -1$ and

$\mu_N = +1$, if the source is emitted in all directions. Setting a random number ξ equal to $P(\mu)$, one needs to solve for μ as a function of ξ.

Graphically, for the distribution function plotted against μ, a broken line curve is obtained such as the one shown in Figure 2.12. For each random number ξ,

$$\frac{\xi - P_{i-1}}{P_i - P_{i-1}} = \frac{\mu - \mu_{i-1}}{\mu_i - \mu_{i-1}}$$

is obtained from which μ can be calculated. Here

$$P_i = \int_{-1}^{\mu_i} p(\mu)\, d\mu$$

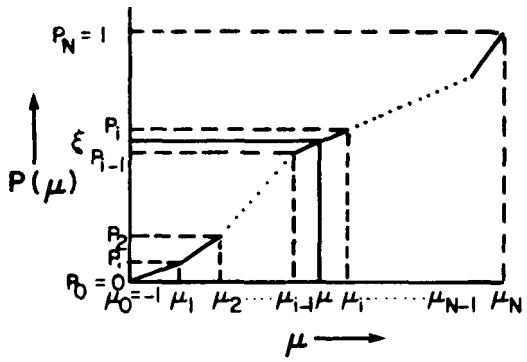


Figure 2.12

In order to obtain $p(\mu)$, the derivative of $P(\mu)$ is needed, which for the case at hand is easily seen to be given by

$$p(\mu) = \frac{P_i - P_{i-1}}{\mu_i - \mu_{i-1}} \cdot \mu_{i-1} < \mu < \mu_i$$

Summarizing, the function $p(\mu)$ is a probability density function in μ and consequently must satisfy the relation $\int p(\mu)d\mu = 1$ over the range of values of μ occurring in the problem. In practice one passes directly from the ω_i's to the P_i's in setting up the source, since $P_i = \sum_{j=1}^{i} \omega_j$. That is, the value of the cumulative distribution function at μ_i is simply the sum of the probabilities of being emitted in all bins up to and including the

i^{th} bin. The calculation of $p(\mu) = (P_i - P_{i-1})/(\mu_i - \mu_{i-1})$ is not required in sampling from the source unless there are point detectors in the problem.

(4) As a last example, suppose a plane wave of particles is the source impinging perpendicularly on an area A at the detector. If the detector can be hit by a source particle the flux contribution is given by

$$\frac{1}{A} \cdot e^{-\int_0^R \Sigma_t(s)ds}$$

per source particle where $p(\mu,\varphi)d\Omega = p(\mu,\varphi)/A$ and $p(\mu,\varphi) = \delta(\mu-\mu_o)\delta(\varphi-\varphi_o)$. In this manner each source particle contributes to the detector. (If the source is monoenergetic one calculation is sufficient, but in the general case of an energy-dependent source this will reduce the variance. It is assumed that the energy distribution is independent of the position of entry).

To compute the above probability PSC for a particular collision process, MCNP must use the available information to determine PSC and the new particle energy for a scattering through μ_o where μ_o is the cosine between the line of flight and the direction to the detector. For example, in the simplest case of neutron elastic scattering through the angle $\cos^{-1}\mu$ in the center-of-mass system, the angular scattering probability $P_c(\mu)$ may be stored for the center-of-mass coordinate system, where $\nu = \cos\theta$ refers to the angle θ of scattering in the laboratory system. However, there are the well-known relations

$$\nu = \frac{1+A\mu}{\sqrt{1+A^2+2A\mu}}$$

where A is the ratio of the mass of the target atom to that of the neutron, and

$$E' = E\left(\frac{1+\alpha}{2} + \frac{1-\alpha}{2}\mu\right)$$

where E is the incoming neutron energy, E' is the energy of the scattered neutron, and

$$\alpha = (A-1)^2/(A+1)^2$$

Using these formulas,

$$PSC = P_c[\mu(\nu)]\ (d\mu/d\nu)$$

may be computed. Knowing ν from the position of the detector point, $\mu(\nu)$ may be computed and then PSC and the outcoming energy E' of the scattered neutron determined.

In treating the various inelastic processes, one must allow for scattering and energy distributions given in either the laboratory or center-of-mass systems. In the first case, the evaluation of PSC is simple, but in the second case, one must use the appropriate formulas linking the incoming laboratory energy of the neutron, the outgoing laboratory energy, the outgoing energy in the center-of-mass system, the scattering angle in the center of mass system, and the scattering angle in the laboratory system. These formulas will not be given here, but they are readily derived from considerations of the collision process.

When a neutron from a free-gas thermal collision can tally at a detector at either of two energies, to save computer time one energy is randomly selected and the tally contribution is doubled. A flow diagram of the point-detector estimator in conjunction with the free-gas thermal treatment can be found in Reference 1. This same technique of multiplying the neutron weight by $\bar{\nu}$ or x and then making only one detector calculation is also done for fission and (n,xn) events even though in the actual non-detector transport the actual number of particles is followed.

Currently theory does not exist in MCNP for contributions to a detector from $S(\alpha,\beta)$ thermal collisions.

Detectors should not be used with reflecting surfaces. Consider Figure 2.13 with a point detector and eight source cells. The imaginary cells and point detector are also shown on the other side of the mirror. The source contribution from the indicated cell is shown as the solid line. MCNP does not (at present) allow for the dashed-line contribution on the other side of the reflecting surface. The result is that contributions to the detector will always be from the solid path instead of from a mixture of solid and dashed contributions. This same effect will also occur at every collision. Therefore, the detector tally will be higher than the correct answer and thus should not be used. This effect is even worse for problems with more than one reflecting plane.

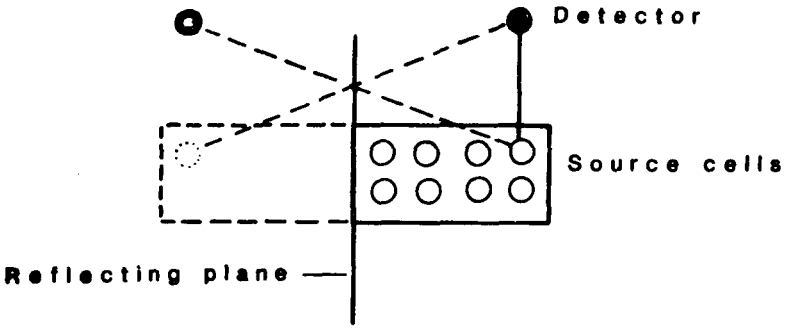


Figure 2.13

1. Analog Point Detector

As pointed out previously for the analog detector in a scattering medium, the theoretical variance of the estimator is infinite because of the $1/R^2$ difficulty even though the first moment is finite. The variance is proportional to

$$\int R^2 dR \left(\frac{1}{4\pi R^2}\right)^2$$

When the detector point is outside the scattering region (such as in a void), this analog technique of estimating flux is in general quite reliable. To cope with this $1/R^2$ singularity, the code contains the simple technique of computing an average contribution for collisions within a spherical neighborhood of the detector. To be more precise, if one assumes that the flux is isotropic and uniform in a spherical region surrounding the point, one can devise the formula

$$\frac{PSC \cdot (1-e^{-\Sigma_t R_0})}{\frac{2}{3}\pi R_0{}^3 \Sigma_t}$$

for the average contribution to the flux at the detector for particles colliding in the spherical region, where PSC and σ_t are defined as before, and R_0 is the radius of the fictitious sphere about the point. Using this expression does not cure all difficulties, but it can help prevent the rare collision very close to the detector from seriously perturbing the calculation.

The choice of R_0 may require some experimentation, since the sphere should be large enough to enclose a reasonable number of collisions, but not so large that the above assumptions are violated. For a typical problem, R_0 may be chosen as a fraction of a mean free path (on the order of 1/8 to 1/2), but it is most important that a good sample should be obtained in the vicinity of the point detector. Otherwise either the estimate of the flux will be too low, or the occasional collision in the vicinity of the detector will carry too much weight, leading to large variances in the result. For a detector in a void or a region with very few collisions (such as air), R_0 may be set to zero.

The radius of the sphere R_0 can be specified in centimeters or in mean free paths. Mean free path is more straightforward for the user, but it requires more machine time since at every collision the cross section of the material in the sphere must be determined – plus it will increase the variance. If the input is in centimeters, that should correspond to the mean free path for some average energy in the sphere.

Caution must be observed when defining R_0 such that the sphere it defines does not encompass more than one material unless you understand the consequences. This is especially true when using R_0 in terms of mean free path since R_0 becomes a function of energy and can vary widely. In particular, if R_0 is defined in terms of mean free paths and if a detector is on a surface that bounds a void on one side and a material on the other, the contribution to the detector from the direction of the void will be zero even though the importance of the void is non-zero. The reason is simply that the volume of the artificial sphere is infinite in a void. Contributions to the detector from the other direction (i.e., across the material) will be accounted for.

For the analog point detector, at each collision and for each detector MCNP calculates a contribution to each detector with a probability set on the PDn card or within λ_{max} specified on the DD card. The sequence of the calculation begins in the main transport subroutine HSTORYN where a call is made to subroutine TALLYD on collision or with a source particle. In TALLYD, the contribution T to the detector is determined by one of the above appropriate formulas. The call to TALLYX is then made for any possible user modifications, and final norming takes place in subroutine TALLYP.

2. OMCFE Detector[11]

This is a finite variance estimator for the flux at a point. The procedure involves the sampling of an imaginary intermediate collision point from a real collision. By choosing the probability density function of this intermediate point in a suitable way, an estimator with only a $1/R$ singularity can be derived. The analog flux estimator has a $1/R^2$ singularity, where R is the distance from collision point to detector. The OMCFE thus has an advantage over the analog scheme, especially when the detector is embedded in a scattering medium. The variance is proportional to

$$\int R^2 dR \left(\frac{1}{4\pi R}\right)^2$$

and is therefore finite.

In order to speed up the calculation, the sampling of an intermediate collision point is not done at every real collision. An imaginary sphere of specified radius R_o is drawn around the detector. The once-more-collided game is played if the real collision occurs within this sphere. It is also played if the collision is outside the sphere but the direction after collision is inside the cone defined by the collision point and the sphere. Otherwise, an analog contribution is made.

The effectiveness of the OMCFE is also very sensitive to the radius R_o of the sphere. If R_o is chosen too small, this in effect reduces the technique to the analog case. If R_o is too large, the running time increases considerably. The scheme is also relatively ineffective if the collision point is a long way from the detector. A rule of thumb that a sphere radius of about 1.5 to 3 mean free paths at the source energy seems to be the most effective. As with the analog detector, the sphere radius R_o can be specified in centimeters or in mean free paths.

The OMCFE does not work miracles with the point detector tally variance. It does generate more pseudo-collision points closer to the detector than real collision points. Some thought has to be given, as in any point detector problem, to source biasing, cell splitting, or other biasing schemes in order to increase the sampling of real collision points reasonably close to the detector.

The PDn and DD cards are also available for use with the OMCFE.

Furthermore, as with the analog detector, the PRDMP card should be used to get an idea of the variance of the variance.

There are several restrictions and cautions. At present, the OMCFE is not applicable to neutrons thermalized according to the $S(\alpha,\beta)$ treatment (this is also true for the analog detectors). Also, the scheme does not apply to a ring detector. Variance problems can result if the detector sphere includes materials of greatly different mean free paths. One moderating effect is to specify the sphere radius in centimeters and not in mean free paths. If the change in total cross section at a material boundary is too drastic (such as an air/water interface), however, the flux may have to be calculated some other way - such as with DXTRAN. The detector sphere can include regions that are void or have zero importance, but as with the analog scheme contributions are not made through a region of zero importance. Finally, as with the analog scheme, the OMCFE does not handle reflecting surfaces.

3. Ring Detector[12]

The ring detector tallies are modified point detector tallies, and everything said in the above discussion applies to the ring detector. The ring detector allows enhanced computing efficiency for problems symmetric about a coordinate axis. For example, if a geometry and spatial source distribution are symmetric (or at least the asymmetry is insignificant) about a coordinate axis and a point detector flux estimate at radius r from the symmetry axis is desired, the ring detector will yield a smaller relative error than the point detector for a given number of histories. Ring detectors may also be used for non-axisymmetric problems where one is interested in the average flux at a point on a ring about a coordinate axis.

If the radius of the ring is very large compared to the dimensions of the scattering media (such that the detector sees essentially a point source in a vacuum), the ring detector is still more efficient than a point detector. The reason for this unexpected behavior is that the individual scores to the ring detector for a specific history have a mean which is closer to the true mean than the regular point detector contributions. The ring detector contributions from one history to the total will be much more distributed than those for the point detector and therefore the contribution to the variance for each history is reduced. Another way to view this is that the point detector contributions from one history will

tend to cluster about the wrong mean because the history will not have collisions uniformly in volume throughout the problem.

Furthermore, although the ring detector is based on the analog detector which has a $1/R^2$ singularity and an unbounded variance, the ring detector has a finite variance and only a $1/R_{min}$ singularity where R_{min} is the minimum distance between the contributing point and the detector ring.[13]

In a cylindrically symmetric system, the flux is constant on a ring about the axis of symmetry. Hence, one may sample uniformly for positions on the ring to determine the flux at any point on the ring. The efficiency of the ring detector is enhanced in MCNP by biasing the selection of point detector locations to favor those near the point of contributing collision or source point. This results in the same total number of detector contributions but the large contributions are sampled more frequently resulting in a reduced relative error.

The theoretically "perfect" biasing function to select a detector position on a ring is the range of detector contributions from a single collision. For isotropic scattering in the lab system, this is proportional to $e^{-P}R^{-2}$, where P is the number of mean-free-paths and R is the distance from the collision point to the detector point. For most practical applications, using a biasing function involving P presents prohibitive computational complexity except for homogeneous medium problems. For air transport problems, a biasing function resembling e^{-P} has been used with good results. It was desired to select a biasing function that would be applicable to problems involving dissimilar scattering media and would be effective in reducing variance. The function R^{-2} meets these requirements.

In Figure 2.14, consider a collision point, (x_o,y_o,z_o) at a distance R from a point detector location (x,y,z). The point (x,y,z) is to be selected from points on a ring of radius r. The ring is symmetric about the y-axis in the discussion. A selection of the angle φ is made according to a probability density function proportional to

$$R^{-2}(\varphi) = [a + b\cdot\cos\varphi + c\cdot\sin\varphi]^{-1} ,$$

where

$$a = r^2 + x_o^2 + (y - y_o)^2 + z_o^2$$
$$b = -2rx_o ,$$

and

$$c = -2rz_o .$$

With ξ a random number on the unit interval, it follows that

$$\tan\frac{\varphi}{2} = \frac{A\cdot\tan\left[\pi\left(\xi-\frac{1}{2}\right)\right]-c}{(a-b)}$$

where $A = (a^2 - b^2 - c^2)^{1/2}$. The detector coordinates are then given by

$$x = r\left(\frac{1-t^2}{1+t^2}\right) = r\cdot\cos\varphi$$

$$y = y \text{ (fixed)}$$

$$z = r\left(\frac{2t}{1+t^2}\right) = r\cdot\sin\varphi$$

where $t = \tan(\varphi/2)$. Following the selection of the point detector location on the ring, the appropriate adjustment is made to the particle weight. The procedure is invoked for all source particle births and for all collisions.

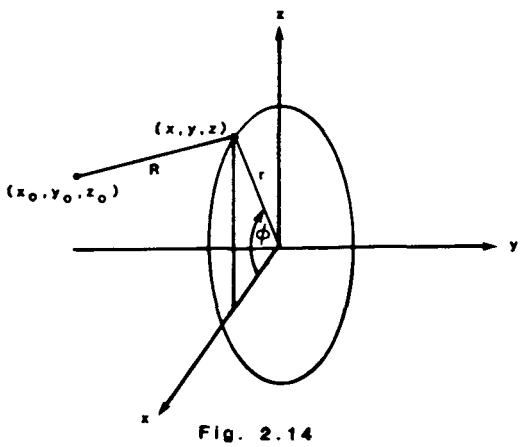


Fig. 2.14

D. Energy Deposition in a Cell

The heating and energy deposition tallies F6, 7, and 16 are simply

$$F6,7 = C \int_{-1}^{1} \int_{t_{m-1}}^{t_m} \int_{E_{n-1}}^{E_n} \phi(E,t,\mu)\ H(E)\ dE\ dt\ d\mu$$

and are track-length estimators equivalent to the F4/14 tallies but multiplied by various parameters:

C = atoms/gram of cell material

H = $\sigma_{tot}(E) \cdot H_{ave}(E)$ for neutron tally F6 where $\sigma_{tot}(E)$ is the material microscopic total cross section (cm^2) and $H_{ave}(E)$ is the average heating number (MeV/collision) of the material. Both quantities are in the MCNP cross-section libraries.

= $\sigma_f(E) \cdot Q_{fis}$ for neutron tally F7 where $\sigma_f(E)$ is the material microscopic fission cross section (cm^2) and Q_{fis} is the fission Q (MeV/fission). Each fissionable nuclide has a $Q_{fis} \sim 180$ MeV/fission; the values of Q_{fis} are carried in an MCNP data statement.

= photon heating (MeV·cm^2/collision) for photon tally F16. This quantity is calculated by MCNP and assumes local energy deposition by secondary electrons.

Important:

Note that the F6 tally determines energy deposition from neutron scattering and fission. Any energy lost to the production of gamma rays is not scored. To account for the gamma-ray energy deposition, a coupled Mode 1 problem must be run using both the F6 and F16 tallies. The total energy depostion is the sum of these two tallies. In the F16 tally, secondary electron transport is not accounted for, and the gamma-ray energy loss is assumed to be deposited locally. A Mode 0 problem, as does a Mode 1 problem without an F16 tally, totally neglects any neutron energy converted to gamma rays that may be subsequently deposited. The neutron F7 tally is a subset of the F6 tally and scores only energy deposition due to fission.

VI. ESTIMATION OF ERRORS

All standard MCNP tallies are normalized to be per starting particle and are printed out accompanied by a second number which is the relative error corresponding to one standard deviation of the mean. If the tally mean is $\bar{x}$ and the relative error is $\bar{\sigma}(\bar{x})/\bar{x}$, then the true result (for an unbiased estimator) should lie within the band $[\bar{x} \pm \bar{\sigma}(x)]$ about 68% of the time. The following discussion explains the meaning of the term relative error.

Suppose that, in a Monte Carlo calculation, the independent sample values $x_1, x_2, \ldots, x_n$ are drawn from a probability distribution that may be unknown. Thus even the mean $E(x)$ and the variance $\sigma^2(x)$ may have to be approximated by their sample values. Certainly this is the case for most of the quantities tallied by MCNP.

The sample mean is defined as

$$\bar{x} = \frac{1}{N} \sum_{i=1}^{N} x_i$$

and the sample variance of x by

$$\bar{\sigma}^2(x) = \frac{1}{N-1} \sum_{i=1}^{N} \left[x_i - \bar{x} \right]^2$$

$$= \frac{N}{N-1} \left[\frac{1}{N} \sum_{i=1}^{N} x_i^2 - \bar{x}^2 \right]$$

or

$$\bar{\sigma}^2(x) = \frac{N}{N-1} \left[\overline{x^2} - \bar{x}^2 \right]$$

where N represents the total sample drawn from the population. For example, in MCNP, N represents the number of particles started from the source, and x_i represents the <u>total</u> contribution to x from the i^{th} starting

particle. This latter definition of x_i is important, for in the various methods of importance sampling (see page 75) and even in treating physical processes such as fission or (n,2n) reactions leading to the creation of neutrons, the i^{th} particle and its offspring may contribute many times to a category x.

In MCNP, contributions to a tally and its second moment are not made individually for each of a particle's offspring but collectively for all offspring at the end of the particle's complete history. This accounts for the fact that the various contributions from the i^{th} particle are correlated.

MCNP estimates the error of the sample mean x. It is well-known that if one draws a sample of size N from a population with true mean E(x) and variance σ^2, then for the sample mean

$$\bar{x} = \frac{1}{N} \sum_{i=1}^{N} x_i$$

one has the two relations

$$E(\bar{x}) = E(x)$$

and variance $(\bar{x}) = \sigma^2/N$

This leads to the common notion that the error goes as the inverse of the square root of the sample size. However, in MCNP the true mean is not known, and consequently the estimate of the variance of the sample mean is

$$\frac{\bar{\sigma}^2(\bar{x})}{N} = \frac{1}{N-1} \left[\overline{x^2} - \bar{x}^2 \right]$$

Because N is usually sufficiently large that the error is negligible in replacing N-1 by N, MCNP uses the following formula for the standard deviation $\bar{\sigma}(\bar{x})$ of the sample mean $\bar{x}$:

$$\sqrt{\frac{\bar{\sigma}^2(x)}{N}} \equiv \bar{\sigma}(\bar{x}) = \sqrt{\frac{\overline{x^2} - \bar{x}^2}{N}}$$

Note that infrequent large contributions to the sample mean can cause fluctuations in $\bar{\sigma}(\bar{x})$. The standard deviation is not a smoothly varying decreasing function of N. In practice, most Monte Carlo calculations involve some form of importance sampling (i.e., variance reduction schemes) that causes the importance (or weight) of a relatively large number of particles to be small while the smaller number of remaining particles have a large weight. When these infrequent, large-weighted particles score at a tally, the tally variance can be severely jolted. It is very good practice to print the tally results at relatively frequent intervals (see the PRDMP card, page 165) to be able to see how the tally and the relative error vary with increasing histories. You may find that an indicated 10% error can suddenly jump to 20%. Unless the error is less than just a few percent, you may be fooling yourself if you just look at one final result and believe it. This is especially true when using detectors. With an analog detector, a tally with an indicated error of 10% can easily be accurate to within only a factor of a few.

In applying the formula for the standard deviation, $\bar{\sigma}(\bar{x})$, to the sample values obtained by Monte Carlo, one uses the Central Limit Theorem of probability theory, which states that

$$\lim_{N\to\infty} \text{Prob}\left[E(x) + \alpha \frac{\sigma}{N^{1/2}} < \bar{x} < E(x) + \beta \frac{\sigma}{N^{1/2}}\right] = \frac{1}{\sqrt{2\pi}} \int_{\alpha}^{\beta} e^{-t^2/2}\, dt$$

In terms of the sample variance, this may be restated in the following approximation for large N:

$$\text{Prob}\left[\alpha\bar{\sigma}(\bar{x}) < \bar{x} - E(x) < \beta\bar{\sigma}(\bar{x})\right] \sim \frac{1}{\sqrt{2\pi}} \int_{\alpha}^{\beta} e^{-t^2/2}\, dt$$

In this form, results from Monte Carlo calculations are readily interpreted from tables of the normal distribution function. For example, a result is within one standard deviation of the sample mean 68% of the time or within two standard deviations 95% of the time.

As an example, let T_n be a flux tally score for a particular source particle n. Then after n histories, the flux printed by MCNP is simply

$$\frac{1}{N} \sum_{n=1}^{N} T_n$$

and the standard relative error is

$$\left[\frac{\sum_{n=1}^{N} T_n^2}{\left(\sum_{n=1}^{N} T_n \right)^2} - \frac{1}{N} \right]^{1/2}$$

Consider two cases:

(1) There are 10^4 histories each scoring with a weight of 1.

$$\text{Tally} = \frac{1}{10^4} \cdot 10^4 = 1$$

$$\text{Variance} = \left[\frac{10^4}{(10^4)^2} - \frac{1}{10^4} \right]^{1/2}$$

$$= [0]^{1/2}$$

$$= 0$$

(2) There are 10^4 histories making 7.5×10^3 scores each with a weight of 1. The rest of the scores are zero.

$$\text{Tally} = \left(\frac{1}{10^4} \right) \left(7.5 \times 10^3 \right) = 0.75$$

$$\text{Variance} = \left[\frac{7.5 \times 10^3}{(7.5 \times 10^3)^2} - \frac{1}{10^4} \right]^{1/2}$$

$$= 0.58 \times 10^{-2}$$

Note that the error in the two cases is less than the common notion of $1/\sqrt{n} = 10^{-2}$. The reason is the weight uniformity of scoring histories. Conversely, when scores have widely varying contributions, the error can be greater than $1/\sqrt{n}$ and is not necessarily a monotonically decreasing function of n. This is a very important concept, especially for detectors.

The relative error is used to estimate how precise the result $\bar{x}$ is. Precision is the uncertainty in a result caused by the random errors of the process involved. Accuracy of $\bar{x}$ is how close $\bar{x}$ is to the true x. MCNP results for $\bar{x}$ are biased by cross sections and physical models used in the calculations. This bias is not known and therefore not accounted for in the relative error estimate. Thus the relative error is a measure of precision and not necessarily of accuracy.

VII. VARIANCE REDUCTION TECHNIQUES

Frequently in Monte Carlo calculations, analog sampling involves prohibitively long running times to determine some quantity of interest with acceptable precision. Consequently, one tries to improve the efficiency of the sampling techniques. A class of schemes to alter or bias the probability density function, so as to sample the particles important to a particular problem more effectively, is called importance sampling.

The basic idea may be demonstrated by considering the evaluation of the following one-dimensional integral:

$$F = \int_a^b f(x)p(x)dx$$

where p(x) is a probability density function with

$$\int_a^b p(x)dx = 1$$

In analog sampling, one would choose points $x_1, \ldots, x_n$ from the density function p(x) and form the mean value

$$\bar{f} = \frac{1}{N} \sum_{i=1}^{N} f(x_i)$$

This yields the Monte Carlo estimate for the value of the integral F. The variance σ^2 of the random variable f(x) is given by

$$\sigma^2 = \int_a^b \Big[f(x) - F \Big]^2 \, p(x)dx = E(f^2) - F^2$$

In general smaller variances occur when scoring particles have weights which are as nearly equal as possible.

Now suppose the density function $\tilde{p}(x)$ is sampled instead of p(x). To each point x_i selected from $\tilde{p}(x)$, the weight $w(x_i) = p(x_i)/\tilde{p}(x_i)$ is assigned and the contribution of particles x_i is scored as $w(x_i)\, f(x_i)$. The expected score is then given by

$$\int_a^b w(x) f(x) \tilde{p}(x) dx = \int_a^b f(x) p(x) dx$$

so that the mean value is again F. However, the variance of the variable w(x) f(x) is given by

$$\int_a^b \Big[w(x) f(x) - F \Big]^2 \, \tilde{p}(x) dx$$

and is not usually the same as the variance in f(x) when p(x) is sampled. Hence it may be possible through judicious choice of $\tilde{p}(x)$ to decrease the variance in a calculation (while leaving the mean unchanged, of course). The decrease in variance is most often the primary reason for altering the probability density function, although one may do so in case the density function p(x) is difficult to sample.

In solving the Boltzmann transport equation, as MCNP does, it is possible to show that if the various density functions entering the equation are altered in just the right way, then the sampling procedure has zero variance. The solution of the adjoint transport equation must be

known, and it is not possible to achieve a zero variance scheme in a practical case, but the very existence of such a scheme encourages one to seek better sampling techniques.

The statistical uncertainty of an MCNP calculation is a function of the distribution of tally scores per source particle. This distribution will usually include a large delta function at the origin on a plot of tally score versus weight of scoring particle representing particles that contribute zero to the tally mean. For non-analog calculations, this distribution will reflect the distribution of particle weights. For track length tallies, the tally score distribution will reflect the distribution of track lengths (or flight paths) through a cell, and where energy response data are folded with the flux data, the tally score distribution will also reflect these response data. When using source biasing, geometric splitting, the exponential transformation, or in general any importance sampling procedure, they will all affect the distribution of tally scores per source particle.

A salutory effect is noted when variance reduction procedures decrease the variance per source particle at a faster rate than the computing time per source particle is increased. Some procedures such as Russian roulette are known to increase the variance per source particle but are employed with the intention of decreasing the computing time per source particle at a faster rate than the increase in variance per source particle. In all cases, the product of variance and computing time is a measure of performance which is independent of either variance or computing time. In other words, one can always reduce the variance without altering the calculation by simply running more source particles, and the variance should then vary as the inverse of the number of source particles which are sampled.

The following simple analysis should give some insight into the effect of importance sampling on variance reduction. Consider first a purely analog simulation without any particle multiplication reactions and in which a surface current tally is employed. The distribution of tally scores will then appear as a delta function at the origin representing non-scoring particles and another delta function at unity as seen in Figure 2.15:

$$f(x) = C_1\delta(x) + C_2\delta(x-1)$$

$$\int f(x)dx = 1$$

$$C_1 + C_2 = 1$$

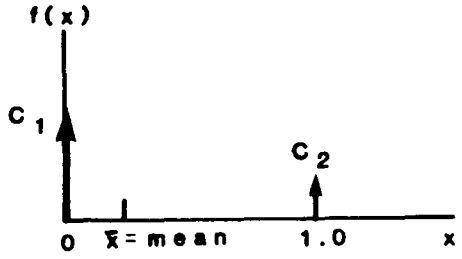


Figure 2.15

The sample mean and variance can be easily evaluated:

$$\bar{x} = \int xf(x)dx = C_2$$

$$\sigma^2 = \int x^2 f(x)dx - \bar{x}^2 = C_1C_2$$

$$= (1-C_2)C_2$$

These are the well-known results for the binomial distribution where C_2 is equal to the probability of a successful trial, or in other words, the efficiency of scoring.

The use of non-analog collisions and other types of importance sampling results in a distribution of tally scores with considerably more structure than for the analog case. In general, this distribution may be

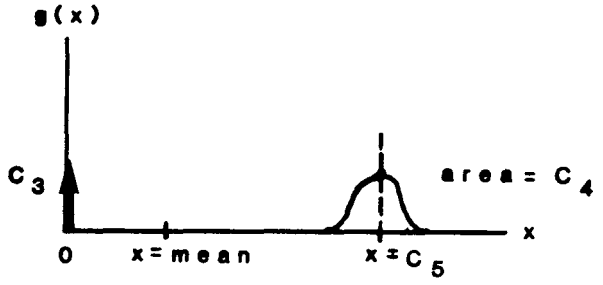


Figure 2.16

represented schematically as follows from Figure 2.16. Because the scoring efficiency is less than 100%, the delta function at the origin will remain, albeit at reduced magnitude from the purely analog case. The remainder of

the distribution is drawn as a peak around a non-zero mean value at $x = C_5$ and with total normalized area (scoring efficiency) equal to C_4. In general, the peak will not be narrow, symmetric, or unimodal, and may not look at all as pictured above. In all cases, however, the sample variance may be rigorously decomposed into a component arising solely from the inefficiency of scoring and a component arising from the intrinsic spread of the non-zero values about the non-zero mean at $x = C_5$:

$$\sigma^2_{\text{total}} = \sigma^2_{\text{efficiency}} + \sigma^2_{\text{intrinsic}}$$

$$\sigma^2_{\text{efficiency}} = C^2_5 C_4(1-C_4)$$

$$= \bar{x}^2(1-C_4)/C_4$$

$$\sigma^2_{\text{intrinsic}} = C_4 S^2$$

where S^2 = variance of non-zero distribution about non-zero mean

$$\bar{x} = C_4 C_5$$

Importance sampling can reduce the total variance by decreasing either or both components of the variance. Geometry splitting, for example, can be used to increase the probability that a source particle will contribute to a tally score and hence increase the scoring efficiency. The magnitude of the delta function at the origin will then decrease, the distribution of non-zero scores will shift to the left toward the mean value, and the area under the non-zero part of the distribution will increase. Note that the distribution of non-zero tally scores may be located at tally score values considerably in excess of the tally mean. If the scoring efficiency is 1%, the typical tally scores will be larger than the tally mean by a factor of 100. The distribution of particle weights at some surface will be even more weakly related to the tally mean because one or more progeny of a source particle may contribute to the total tally score per source particle.

The methods available in MCNP are described below. They should all be used with care and understanding. The stronger the biasing you use with one of these techniques, the more unreliable your results are likely to be. One should avoid as much as possible having particles in the same geometrical region with widely varying weights (it places a greater burden

on adequate sampling). The more biasing you do, the greater the weight variation in most cases. Finally, be aware that using a particular scheme to help get one item of interest may exclude getting some other item. For example, using the exponential transformation to get particles through a thick shield probably excludes a reliable calculation of tritium production within the shield.

Finally, even though the following schemes are all put under the heading of variance reduction techniques, that is not strictly true. Some of the techniques do indeed reduce the variance while others reduce computational time. The goal of all of the techniques is to maximize the function $1/\sigma^2 t$ where σ is the variance and t is the computer time. This function can be thought of as a figure of merit and is inversely proportional to the cost to the actual monetary cost of the job.

A. *Geometry Splitting with Russian Roulette*

This technique is perhaps the simplest and most reliable of all the variance reducing techniques used in general geometry codes. When used judiciously in deep penetration problems, it can save substantial machine time. Basically, as particles migrate in an important direction, they are increased in number to provide better sampling, but if they head in the opposite direction, they are killed in an unbiased manner to avoid wasting time on them. Care must be taken, however, not to over-split; this can lead to a substantial wasting of machine time.

Each cell in the problem geometry setup is assigned an importance I by the user on the IN and/or IP input cards, a number which should be proportional to the estimated value that particles in the cell have for the quantity being scored. When a particle of weight WGT passes from a cell of importance I to one of higher importance I′, the particle is split into a number of identical particles of lower weight according to the following recipe. If I′/I is an integer n ($\geq$ 2), the particle is split into n identical particles, each of weight WGT/n. Weight is preserved in the integer splitting process. If I′/I is not an integer but still greater than 1, splitting is done probabilistically so that the expected number of splits is equal to the importance ratio. Denoting n = [I′/I] to be the largest integer in I′/I, p = I′/I−n is defined. Then with probability p, n+1 particles are used, and with probability 1 − p, n particles are used. For example, if I′/I is 2.75, 75% of the time split 3 for 1 and 25% of the time split 2 for 1. The weight assigned to each particle is WGT·I/I′, which is the expected weight, in order to minimize dispersion of weights.

On the other hand, if a particle of weight WGT passes from a cell of importance I to one of lower importance I', so that I'/I < 1, then Russian roulette is played and the particle is killed with probability 1-(I'/I), or followed further with probability I'/I and weight WGT·I/I'. Weight may not be preserved with the Russian roulette game since there is the possibility of poor sampling and over the lifetime of the problem more histories may win the game than lose - or vice versa. The expected value is, however, unbiased. An indication of how well (or poorly) the Russian roulette game is played and the extent of net weight gain or loss can be found by looking at the creation and loss summary. In the MCNP output in the Creation and Loss table, *Import. Sampling* refers to geometry splitting with Russian roulette on surfaces.

Geometry splitting with Russian roulette is a very reliable variance reduction technique because the weights of all particle tracks are the same in a cell no matter which geometrical path the tracks have taken to get to the cell. (This assumes that there is no weight biasing in the source and ignores the effect of implicit capture on the weights.) The variance of any tally will be reduced when the possible contributors all have the same weight. It is interesting to note that geometry splitting could be done without Russian roulette and vice versa; however, the variance will be larger than with both together because the weights will no longer be the same in a cell.

The assigned cell importances can have any value - they are not limited to integers. However, adjacent cells with greatly different importances place a greater burden on reliable sampling. Once a sample track population has deteriorated and lost some of its information, large splitting ratios (like 20 to 1) can build the population back up but nothing can regain the lost information. It is generally better to keep the ratio of adjacent importances small (like a factor of a few) and have cells with optical thicknesses in the penetration direction less than about two mean free paths. Also, when the importance ratio is not an integer, more sampling is required to achieve an accurate representation as pointed out above, and furthermore, a weight dispersion is introduced which may increase the variance.

Generally, in a deep penetration shielding problem the sample size (i.e., number of particles) diminishes to almost nothing in an analog simulation, but splitting helps keep the size built up. A good rule of thumb is to keep the population of tracks traveling in the desired direction more or less constant - that is, approximately equal to the number of particles started from the source. A good initial recipe is to

split the particles 2 for 1 wherever the track population drops by a factor of 2. Near optimum splitting can usually be achieved with only a few iterations, and conversely, additional iterations show strongly diminishing returns. Note that in a combined neutron-photon problem, importances will probably have to be set individually for neutrons and for photons.

An indication of the track population throughout the geometry is found in the MCNP output summary sheet called *Problem Activity* on a per cell basis. This *Population* is not the same as the *Tracks Entering* information. The latter refers to tracks crossing a surface to enter a cell plus any source particles. However, if a track leaves a cell and later re-enters that cell, it is counted again. A comparison of these two quantities will give an indication of the amount of backscattering. The average energy of particles in a cell and the corresponding number of mean free paths for the cell and energy are printed out in the MCNP summary sheets. This information may be useful in deciding where to put splitting surfaces.

B. *Particle Cutoffs*

When a particle reaches a point, for whatever reason, that it can no longer make a significant contribution to tallies, continuing that history does no more than waste computer time. Histories must be terminated one way or another. MCNP provides several mechanisms for killing particles; Russian roulette on surface crossings has already been mentioned. A common way to kill particles is to define zero-importance cells (see IN or IP card) so that when a particle reaches these cells it is killed. Frequently one *outside-world* cell of zero importance is defined around the geometry of interest so that when a particle leaves the actual geometry it is killed. In addition, histories can be terminated by energy, time, weight cutoff, or by any combination of these. The cutoffs are set on the CUTN and CUTP input cards.

1. Energy Cutoff

Neutron calculations sometimes are required to go down to zero energy. If this is not required for a particular problem, however, an energy cutoff as high as possible should be used since low-energy neutrons typically require a relatively large amount of computer time since the number of collisions increases. For example, consider a 14 MeV neutron source at the center of a concrete shell, inner radius 360 cm and outer radius 390 cm. The number of neutrons processed per minute of computer time and number of

collisions per source neutron as a function of four different cutoff energies are shown in Table 2.1. The number of particles processed per minute is inversely proportional to the collisions per particle.

Energy cutoff for photons is not as critical as it is for neutrons because not that much more time-consuming physics is required for photons. However, the lower the energy goes, the more collisions are involved which does require computer time. Remember that the photon cross sections are valid only above 0.001 MeV.

Table 2.1
Demonstration of Energy Cutoff

Cutoff Energy MeV	Neutrons Processed per minute	Collisions per neutron
10.0	6.27×10^4	2.27
1.0	2.10×10^4	6.58
0.1	1.10×10^4	12.70
0.0	4.50×10^3	32.15

An indication of average particle energy in each cell can be found in the MCNP output summary pages.

Furthermore, a lower energy cutoff requires more cross sections so that computer storage requirements go up and interactive computing with a timesharing system is degraded. The four examples in Table 2.1 used 98651, 169186, 188843, and 202548 $words_{10}$ of storage, respectively. Adding the DRXS input card to the example with no energy cutoff reduced the storage to 110911 words and increased the processing rate to 5.15×10^3 neutrons per minute. The time cutoff for all the above runs was 10^5 shakes.

2. Time Cutoff

A time cutoff is available for both neutrons and photons but it is generally not as useful in reducing computer time for photons since they move so fast. For a neutron with energy in MeV, however, the time in shakes to travel a certain distance in centimeters is calculated in MCNP as

$$\text{Time} = \frac{\text{Distance}}{13.83\sqrt{\text{Energy}}}$$

To cover a 1 meter straight-line distance, a 14 MeV neutron requires 1.93 shakes while a 0.001 MeV neutron requires 228.65 shakes. MCNP does not include any relativistic effects in the velocity of neutrons.

The velocity of photons is the speed of light, 299.7925 cm/shake.

When the last example in Table 2.1 with a zero energy cutoff was run with a time cutoff of 10^4 shakes, 5.04×10^3 neutrons were processed per minute with an average of 28.76 collisions per neutron. With no time cutoff, 4.31×10^3 neutrons were processed per minute with 34.00 collisions per neutron. That is a 14% decrease in running time. The current crossing the outer surface was 3.95×10^{-1} neutrons per starting neutron with a 0.9% standard error for 10000 histories and no time cutoff. With the 10^4 shake time cutoff, the current was 3.79×10^{-1} with a 1.0% standard error. The different results are significant since the two runs are correlated (see page 100) and since they are well outside the error bounds. Therefore, to use a time cutoff (or any other cutoff) to decrease computer time, care must be taken not to kill particles that can contribute to the result.

An indication of particle lifetime can be found in the MCNP output summary pages.

3. Weight Cutoff

Perhaps the most frequently used variance reduction scheme other than geometry splitting is the weight cutoff. It is also the most user-abused cutoff. It is a very effective, simple way to kill particles of little importance (i.e., weight) to keep them from wasting computer time. Unfortunately weight cutoff is one of the most difficult of all Monte Carlo subjects to discuss. That is why you will not find much (if anything) written about it. It is very problem dependent and its setting is an art. If anything very specific is said at all, it will undoubtedly get you into trouble before the day is over. Hopefully the discussion that follows will be some improvement to throwing salt over your left shoulder followed by the two step. The MCNP weight-cutoff parameters are entered on the CUTN and CUTP cards, pages 162 and 163.

For photon problems, the weight-cutoff game is played only in the MCG (simple physics) treatment, whereas the game as not played at all in the MCP (detailed physics) treatment. The reason is that in the MCP treatment, analog capture is required to produce fluorescent photons.

Of the two weight cutoff values inserted into the code on the appropriate cutoff card, the second and smaller value WC2 is the actual so-called "weight cutoff" and defines the lower bound of weights tolerated by MCNP; the larger value WC1 is the new weight the particle receives if it survives Russian roulette. The default values in MCNP are -0.50 and -0.25 (the negative signs make the values relative to the source weight) but are not to be taken as values recommended by X-6. In order to explain how these values are used, assume a problem with all cell importances identical, say they are 1.0 in value. Then if a particle's weight WGT falls below WC2, with probability WGT/WC1 the weight is increased to WC1, and with probability (1 - WGT/WC1) it is killed. The expected surviving weight is WGT. In the code a random number ξ is compared with WGT/WC1; if ξ < WGT/WC1, the particle survives with weight WC1, otherwise it is killed. The survival probability is the ratio WC2/WC1 and as a rule of thumb should be about a half. An indication of how well the Russian roulette game was played (and maybe how well or poorly the cutoff values were set) can be found in the MCNP summary pages in the Creation and Loss table. The weight created on the *Weight Cutoff* line should equal the corresponding weight lost.

In a problem with varying cell importances, the values WC1 and WC2 are modified by MCNP in a given cell by multiplying these values by the ratio of the importance of the source cell to the importance of the given cell (the ratio of these importances multiplied by the source weight is the weight particles would have in the given cell just due to the splitting process). If R is the ratio of the source cell importance to the collision cell importance, if a particle's weight WGT falls below WC2*R, then with probability WGT/(WC1*R) the particle survives and is assigned weight WGT = WC1*R. In a Mode 1 problem, R based on neutron importances is used in the photon weight cutoff game to account for low-weighted photons born from low-weighted, split neutron tracks.

Some care should be exercised by the user in selecting WC1 and WC2. If the particles in a problem have a starting weight of 1.0 or if the average starting weight is 1.0 (and many problems fit into one of these categories), then WC1 is frequently chosen to be 1.0. The choice of WC2 is generally more difficult and should take into consideration as many of the physical aspects of the problem as possible. For example, if capture is negligible in a particular problem and no biasing schemes (other than splitting which is accounted for, as described above) are used which causes particles to have unequal weights in a given cell, then the choice of WC2 is easy - it may be chosen to be any value less than WC1, say WC2 = 0.5.

However, usually the choice is not so simple. If a cell can contain particles with a wide variation in weights, then the user has to consider that the low weighted particles cannot contribute as heavily to any tally as those with larger weights, and therefore some of them should be eliminated (to save machine time) in favor of fewer particles with larger weights. A very important consideration here is that the error is generally reduced in a calculation if the particles contributing to the tally have weights which are as nearly equal as possible. However, naturally one does not want to play Russian roulette so frequently that a sample size deteriorates seriously.

Consideration of the other cutoffs, such as time and energy, may help the user to decide upon an appropriate weight cutoff. Certainly a few short runs of a problem may be helpful, even necessary, to arrive at the weight cutoff. It is particularly important to determine that the cutoff is not prejudicing the scores for the various tallies.

As mentioned, a good way to set the cutoffs is simply by trial and error - run a few short jobs with different weight cutoffs and see what the effect is. In particular, look at the MCNP summary pages (see Chapter 5) for the number of tracks lost to weight cutoff. If a large number is lost to weight cutoff then the second weight cutoff WC2 is probably too high or the cell importances are not appropriate to the problem. On the other hand, the second cutoff is probably too low if it has been reduced and the tallies are about the same but the running time increases.

The weight cutoff game can be avoided by setting the second value to zero (but this can lead to significantly longer running times) or by setting the first value to zero. With the first value zero, capture is analog versus implicit by weight reduction. See page 93 for a discussion of capture treatments along with advantages and disadvantages of each method. In a system with a high degree of capture, the weight cutoff will have to be lowered to maintain the low-weighted particles.

A low weight cutoff can be misleading when a problem is near criticality because low-weighted neutrons can still cause a large number of fission events. The MCNP summary sheets will show the average weight and number of particles in the problem cells which will indicate how important the fission particles really are.

Caution should be exercised when calculating system multiplication (see page 106) when using weight cutoffs. If the weight balance from Russian roulette in the weight cutoff game is not very small (and also from Russian roulette in surface importance sampling), the system multiplication may be in error especially if you are interested in multiplication to

several decimal places. It probably will be more accurate in this case to enter the weight cutoff parameters as zero to cause analog capture (see page 93) and avoid the weight cutoff game entirely.

To illustrate the effect of weight cutoffs and that there is no clear-cut formula to set them, seven problems running about five minutes each on the CDC-7600 were run with different weight cutoffs. The geometry of the problem is very complicated, consisting of 170 different cells and 18 different materials. The runs are correlated (see page 100), each run having been done with 20946 histories. The job was originally run for about an hour with WC1 set to $WS*10^{-7}$ and WC2 set to $WS*10^{-8}$. A result of interest to the problem determined by the long run was 8.66 with a standard deviation of 3.6%. About 3.2% of the started tracks were lost to weight cutoff. Results of the seven runs are found in Table 2.2.

In runs 1, 2, and 3, the weight cutoffs entered on the CUTN card are the starting weight times the fraction indicated in the table. Run 4 treats capture implicitly by weight reduction but neutrons are never killed because their weight never reaches 0. In run 5 the weight cutoff game is not played, and capture is treated explicitly in the analog fashion. Run 3 is the MCNP default - you will get this if you don't specify anything.

Table 2.2
Effect of Weight Cutoff

Run	WC1	WC2	Neutrons per min.	% Tracks Lost to Weight Cutoff	Result	% Standard Deviation	$1/\sigma^2 t$
1	WS*1.0	WS*1/2	5.06×10^3	11.2	7.45	12.2	16.2
2	WS*2/3	WS*1/3	4.84×10^3	10.0	8.19	11.9	16.3
3	WS*1/2	WS*1/4	4.78×10^3	9.3	7.75	10.9	19.2
4	$WS*10^{-7}$	0	3.72×10^3	0	9.39	13.9	9.19
5	0	0	5.17×10^3	12.1*	6.00	13.9	12.8
6	-1.0	-0.5	4.86×10^3	10.3	7.00	12.3	15.3
7	-.01	-.005	4.24×10^3	6.3	7.90	11.5	15.3

*12.1% of tracks lost to analog capture

In runs 6 and 7, the weight cutoffs are entered as negative numbers which makes them relative to the starting weight WS. With a biased source the cutoff is relative to the smallest starting weight after biasing.

The recommended cutoffs for this particular problem are in run 3, mainly because the error in the result is the smallest and the number of neutrons processed per minute is very good. If a *figure of merit* is defined to be the inverse of the square of the percent relative error times the running time, $1/\sigma^2 t$, for the seven jobs in the above table it is clear the best is run 3 and the worst is run 4. This figure of merit is inversely related to the actual monetary cost of the job.

Note that in run 5 with analog capture, the result is not in the same ball park as the other results. This is apparently so because analog capture in general requires larger samples (i.e., more histories) than capture by weight reduction, and 20946 histories just aren't enough in this problem. Also note that in runs 6 and 7 with the relative weight cutoffs the errors are higher than in run 3 (probably due to inadequate sampling). The error in run 7 is not bad, but note the decrease in neutrons processed per minute.

In all of these short MCNP runs the second weight cutoff is half the first weight cutoff. This says that the survival probability in the Russian roulette part of the weight cutoff game is 0.5 versus 0.1 in the hour-long run. There is nothing wrong mathematically with lower survival probability; it however places a greater burden on the sampling.

The above recommended weight cutoffs apply only to this particular problem. The precedure used to get the cutoffs is the same for any problem; use your knowledge of the physics of the problem, run two or three short jobs, and study the results. There is no magic formula. The seven runs made here were done mainly to illustrate various properties of weight cutoffs. Only the first three at most were needed for the choice, and with a little experience and forethought, two runs (2 and 3) would be sufficient.

Note that weight cutoff is used only to reduce computing time when other mechanisms such as escape, energy cutoff, zero importance, or time cutoff are not adequate in limiting excess collisions. The Russian roulette game will not reduce the tally variance but can only have the opposite effect.

C. *The Exponential Transformation*

The exponential transform, alias "path-length stretching," appears to be a useful variance reduction technique in only a very limited class of transport problems. Although many impressive results are stated for the exponential transform, it should be remembered that these results are

usually obtained for one-dimensional geometries and quite often for energy-independent problems. A review article by Clark[14] gives theoretical background and sample results for the exponential transform. Sarkar and Prasad[15] have done a purely analytical analysis for the optimum transform parameter for an infinite slab and one energy group.

It is suggested that MCNP be used without the exponential transform except in the simplest of problems because of difficulties in using the transform reliably and effectively. When the transform is improperly used, the sample mean may be unreliable while the sample variance may erroneously indicate an acceptable precision. Furthermore, geometric splitting and Russian roulette can be almost always used instead of the transform and still achieve good results with a more reliable error estimate.

If you need to use a weight cutoff, you should not use the exponential transform because MCNP does not currently have a weight cutoff game designed for use with the exponential transform. The weight cutoff game used is independent of the particle position (assuming no geometry splitting is used); this is clearly inappropriate since the exponential transform assumes the particle weight to decrease exponentially with increasing penetration in the preferred direction. If you really do need the transform, you should consider patching MCNP with something like a cell-dependent weight-window patch to get around the problem. Contact X-6 for an existing patch to accomplish this.

The exponential transform allows particle walks to move in a preferred direction by artifically reducing the macroscopic cross section in the preferred (+y in MCNP) direction and increasing the cross section in the opposite direction according to

$$\Sigma_{ex} = \Sigma_t \ (1 - p\mu),$$

where

Σ_{ex} = fictitious transformed cross section
Σ_t = true total cross section
p = the exponential transform parameter used to vary the degree of biasing $|p| < 1$
μ = cosine of the angle between the preferred direction and the particle's velocity. $|\mu| \leq 1$

At a collision a particle's weight is multiplied by a factor w_c (derived below) so that the expected weight colliding at any point is

preserved; that is, the particle's weight is adjusted such that the weight multiplied by the probability that the next collision is in ds about s remains constant.

The probability of colliding in ds about s is

$$\Sigma e^{-\Sigma s} ds$$

(where Σ is either Σ_t or Σ_{ex}) so that preserving the expected collided weight requires

$$\Sigma_t e^{-\Sigma_t s} ds = w_c \Sigma_{ex} e^{-\Sigma_{ex} s} ds,$$

or

$$w_c = \frac{\Sigma_t e^{-\Sigma_t s}}{\Sigma_{ex} e^{-\Sigma_{ex} s}} = \frac{e^{-p\Sigma_t \mu s}}{1 - p\mu}$$

If the particle reaches a cell surface instead of colliding the particle's weight is adjusted so that the weight, multiplied by the probability that the particle travels a distance s to the cell surface, remains constant. The probability of traveling a distance s without collision is

$$e^{-\Sigma s} ,$$

so that preserving the expected uncollided weight requires

$$e^{-\Sigma_t s} = w_s e^{-\Sigma_{ex} s},$$

or

$$w_s = \frac{e^{-\Sigma_t s}}{e^{-\Sigma_{ex} s}} = e^{-p\Sigma_t \mu s}$$

For one-dimensional, deep penetration through highly absorbing media, the variance will typically decrease as p goes from zero to some p′, and then increase as p goes from p′ to one. For $p < p'$, the solution is said to be "underbiased" and for $p > p'$, the solution is "overbiased."

Choosing p′ is usually a matter of experience, although some insight may be gleaned by understanding what happens in severely underbiased and severely overbiased calculations. For illustration, apply the variance analysis of page 77 to a deep penetration problem when the exponential transform is the only non-analog technique used. In a severely underbiased calculation ($p \rightarrow 0$), very few particles will score, but those that do will all contribute unity. Thus the variance in an underbiased system is due to a low scoring efficiency rather than a large dispersion in the weights of the penetrating particles. In a severely overbiased system ($p \rightarrow 1$) a large number of particles will score, but there will be a large dispersion in the weights of the penetrating particles with a resulting increase in variance.

A variant of the exponential transformation may be useful in some cases. If one is interested in studying some collision process in a relatively thin material, a fictitious total cross section larger than the actual cross section may be obtained by using a negative value for the parameter p. This will have the effect of artifically creating more collisions in the material, with the weight of the colliding particle adjusted accordingly.

To use this variant, however, the total cross section should be increased uniformly in all directions. Since MCNP provides for a directionally dependent transformation (along the y-axis), the code must be modified by the user. It is suggested to use forced collisions rather than this approach.

Furthermore, no guidelines are available on appropriate values for a negative p other than it must not be unbounded. The next section on forced collisions discusses a better approach to thin cells.

D. *Forced Collisions*

Sometimes it is desired to sample a relatively thin cell (a fraction of a mean free path) to accurately calculate quantities like a reaction rate or energy deposition or to cause interactions that are important to some other part of the problem. It may then be necessary to force particles to have one or more collisions in these cells. Furthermore, since contributions to a detector come only from collisions, a forced collision may be required in some situations.

Basically, the particle entering the cell in question is separated into two components: (1) an uncollided part weighted by the probability of transmission which is stored in the bank until later when its track is resumed at the boundary on the other side of the cell, and (2) the collided

part, derived by sampling the distance to collision on the path length up to the boundary on the other side of the cell, and weighting the particle by the probability of having a collision along that length.

If DLS is the distance across the cell in the direction a particle of weight WGT is traveling and QPL is the macroscopic cross section of the cell material, then the particle is transmitted with probability exp(-DLS*QPL) so the uncollided part referred to above is assigned a new weight WGT*exp(-DLS*QPL). The remaining component that has the collision then has a weight WGT*[1 - exp(-DLS*QPL)]. The position PMF of the collision must be determined on the interval $0 \leq PMF \leq DLS$ within the cell according to the formula $\xi = P(PMF)/P(DLS)$ where $P(PMF) = 1-\exp(-PMF*QPL)$ and ξ is a random number. Solving for PMF, one obtains

$$PMF = -\frac{1}{QPL} \ln\left\{1-\xi\left[1-\exp(-DLS*QPL)\right]\right\}$$

The disadvantage of using forced collisions versus the exponential transformation for thin cells is that it may be more time consuming because it generates an extra particle to be followed and also requires the relatively slow evaluation of an exponential. The advantage is that it is a standard feature of MCNP and that it guarantees a given number of collisions (which may be an advantage when using a detector) whereas the exponential transformation will provide a given number of collisions only on the average.

Because forced collisions can lead to the collided part of the particle having a very small weight, the weight-cutoff game (see page 84) is not played in cells in which collisions are forced. However, it is turned back on in the other cells. If you are interested in tracking the collided part in the other cells, you may have to set the weight cutoff very low (maybe even zero) to keep the track alive.

Computer time can increase significantly as the number of forced collisions increases. A rule of thumb is to force as few per cell as possible. Start with one for a short run and then go to two for another short run if the first run doesn't produce the desired effect. The number of collisions to force is reset to its initial value when the particle leaves the cell.

In principle, forced collisions can be used for deep penetrations because the uncollided component is transported through the thick cell determininstically. However, because the collided component is in a thick

cell in which the weight-cutoff game is not played, the particle can rattle around inside the cell for a long time. Computer time can increase significantly in this case.

E. *Capture by Weight Reduction*

Capture can be treated in two ways: (1) explicit analog capture in which a particle is absorbed and its track is terminated, or (2) implicitly by reducing the particle's weight according to the capture probability and allowing the track to continue. This second way is frequently called survival biasing. With implicit capture of a particle of weight WGT, a weight of $WGT*\Sigma_a/\Sigma_t$ is deposited at the point of capture and a particle of weight $WGT(1-\Sigma_a/\Sigma_t)$ continues (Σ_a is the absorption cross section and Σ_t is the total cross section). Implicit capture is frequently called survival biasing.

In neutron problems, if WC1 on the CUTN card is set to zero, capture is treated in the analog fashion. In photon problems, the MCG treatment allows only for capture by weight reduction (and thus the weight-cutoff game), and MCP allows only for analog capture (and no possibility of the weight-cutoff game). The MCP treatment requires analog capture in order to have fluorescent emission following photoelectric absorption (see page 48).

For photon problems where the detailed physics of MCP is required, there is no choice but to use analog capture. However, where there is a choice (as with MCN or using MCG rather than MCP - see the ERGP input card), the implicit capture has distinct advantages. In deep penetration problems particularly, if a particle works very hard to get through a thick cell and is absorbed explicitly just before it gets out, a lot of computer time has been used with nothing to show for it. With capture by weight reduction, however, the particle's weight is reduced and the track keeps going. Thus the sample size is not reduced and the statistics of the problem is thereby improved. A second advantage of implicit capture is that the number of collisions in a history is increased which is an advantage when using detectors since a contribution to a detector is made at every collision point (see page 56).

Problems with analog capture may (and generally do) run with a greater particle processing rate than with implicit capture simply because there are not as many collisions per history. However, it will probably require more histories with analog capture to obtain the same statistical error that can be obtained with fewer histories and implicit capture. See Table 2.2 and its discussion for an example of this.

F. *Flux at a Detector*

Because flux contributions to a detector are made at every collision point the large sample size generally yields results with a lower variance than results from a tally of the actual flux at (or in the close proximity of) a detector. However, because many time-consuming calculations are made, detectors are generally quite expensive compared to other tallies, and the number of detectors should be kept as small as possible.

MCNP provides two input cards (PDn and DD cards, see pages 131 and 152) that can reduce the number of collisions from which tally scores are made and bias the locations of these contributing collisions. When used judiciously, these cards can significantly speed up detector calculations. The PDn card allows the user to set in each cell a probability P for making a detector contribution calculation for each collision. At each collision in cell I, the detector tallies are made with probability P_I ($0 \le P_I \le 1$). The tally is then increased by the factor $1/P_I$ to obtain unbiased results for all cells except those where $P_I=0$. In this case, no contribution will even be made from cell I. This enables the user to increase the problem efficiency by setting $P_I<1$ for cells many mean free paths from the detectors. It also enables the selective suppressing of contributions from cells by setting the P_I's to zero.

The DD card provides a very easy to use and effective way to speed up detector calculations and is in general preferable to the PDn card. The first entry on the DD card is the number of mean free paths beyond which Russian roulette is played for contributions to a detector. However, to use the DD card effectively a short run must first be made with this first entry set to zero which means to accept all contributions. The special printout from the DD card will then indicate where the detector contributions are coming from.

See page 55 for a more complete discussion of detectors.

Rather than using a detector, consideration should be given to determining average flux in a cell by a track-length tally F4/14. These tallies are quite reliable in that the sample size is generally large and there are no singularity problems as with detectors. The following DXTRAN scheme can also be used in many cases in place of detectors (for example, an F2 tally across a segmented surface inside a DXTRAN sphere).

G. *DXTRAN*

In a geometry region which is difficult to sample adequately, the DXTRAN routine can frequently be of value. At each collision, scattered particles are deterministically transported to a neighborhood of interest. These particles will later be transported in the ordinary random walk manner. To explain how the scheme works, consider the neighborhood of interest to be a spherical region surrounding a designated point in space. In fact, consider two spheres of arbitrary radii about the point (x_o,y_o,z_o). Further assume that the particle having direction (u,v,w) collides at the point (x,y,z), as shown in Figure 2.17.

Note that the average flux provided by a DXTRAN sphere is not necessarily equal to the actual flux at a point inside the sphere and is frequently higher. This is the case if the actual flux at the location of the sphere is not constant but varies as a function of distance through the

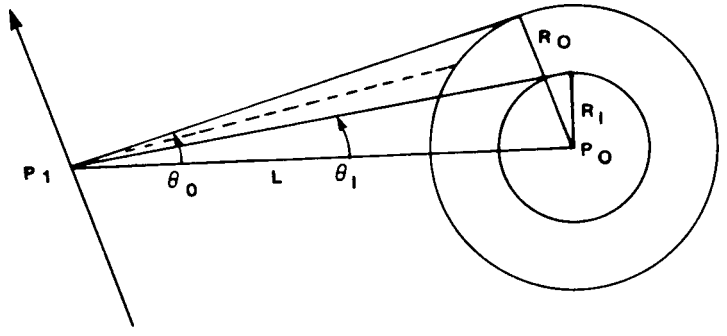


Figure 2.17

sphere - which is a common situation.

The quantities θ_I, θ_O, η_I, η_O, R_I, and R_O are defined in the figure. Thus L, the distance between the collision point and center of the spheres, is

$$L = [(x-x_o)^2 + (y-y_o)^2 + (z-z_o)^2]^{1/2}$$

On collision, a pseudo particle is placed at a point on the outer sphere of radius R_O as described below, unless altered by the DXCPN or DXCPP cards. Provision is made for biasing the contributions of these pseudo particles on the outer sphere within the cone defined by the inner sphere. The weight of the pseudo particle is adjusted to account for the probability of

scattering in the direction of the point on the outer sphere, and traversing the distance with no further collision.

The steps in sampling the pseudo particles is outlined:

$$\eta_1 = \cos\theta_1 = (L^2 - R_1^2)^{1/2}/L$$

$$\eta_0 = \cos\theta_0 = (L^2 - R_0^2)^{1/2}/L$$

Sample $\eta = \eta_1 + \xi(1 - \eta_1)$ uniformly in $(\eta_1,1)$ with probability
$Q(1 - \eta_1)/[Q(1 - \eta_1) + \eta_1 - \eta_0]$,
and with probability
$(\eta_1 - \eta_0)/[Q(1 - \eta_1) + \eta_1 - \eta_0]$
sample $\eta = \eta_0 + \xi(\eta_1 - \eta_0)$ uniformly in (η_0,η_1). The quantity Q is a factor which measures the weight or importance which is assigned to scattering in the inner cone relative to the outer cone.

Having chosen $\eta = \cos\theta$, a new direction (u',v',w') is computed by scattering through an angle θ (and a uniform azimuthal angle φ) from the initial line of flight direction to the new direction:

$$\left(\frac{x_o-x}{L}, \frac{y_o-y}{L}, \frac{z_o-z}{L}\right)$$

The particle is advanced in the direction (u',v',w') to the surface of the sphere of radius R_0. The new pseudo particle with appropriate direction and coordinates is banked. The weight of the pseudo particle is determined by multiplying the weight of the particle at collision by

$$\nu \cdot \frac{P(\mu)\ \{Q(1-\eta_1)+\eta_1-\eta_0\}\ e^{-\int_{P_1}^{P_2}\Sigma_t(s)ds}}{Q}, \quad \eta_1 \le \eta \le 1$$

and

$$\nu \cdot P(\mu)\ \{Q(1-\eta_1)+\eta_1-\eta_0\}\ e^{-\int_{P_1}^{P_2}\Sigma_t(s)ds}, \quad \eta_0 \le \eta \le \eta_1.$$

where

$$\mu = uu' + vv' + ww'$$

$P(\mu)$ = scattering probability density function for scattering through the angle $\cos^{-1}\mu$ in the lab system for the event sampled at (x,y,z)

ν = number of neutrons emitted from the event

$e^{-\int_{P_1}^{P_s}\Sigma_k(s)ds}$ = the attenuation along the line between $P_1(x,y,z)$ and P_s, the point on the sphere where the particle is placed.

In arriving at the weight factor, note that the density function for sampling η is given by:

$$Q/[Q(1-\eta_1)+\eta_1-\eta_0], \quad \eta_1 < \eta \leq 1$$
$$1/[Q(1-\eta_1)+\eta_1-\eta_0], \quad \eta_0 \leq \eta \leq \eta_1$$

Thus the weight needs to be scaled by the true probability divided by the probability density function.

The attenuation is calculated at the energy obtained by scattering through the angle μ_o. The energy is uniquely determined from μ_o in elastic scattering (and also in level scattering), while for other non-elastic events, the energy is sampled from the corresponding probability density function for energy, given μ_o, and may not depend on μ.

It should be clear from the discussion above that this routine has certain features in common with the point detector.

The main advantage of DXTRAN is that it can be used to improve statistics in tallying information in a localized region in space. There are many cases where some quantity is needed in a small region of space (e.g., flux or reactions in a small region), but where statistical accuracy is difficult to obtain. The tally of interest may be only one of many in the problem and the flux of real particles may tend to be low in the vicinity.

The disadvantage of the method is that it is time consuming, being similar in nature to that of a point detector. In fact, many of the same considerations as one applies to the successful use of a point detector should be used in DXTRAN. Success is not guaranteed in using DXTRAN unless one pays attention to the problem of sampling the histories in the vicinity of the DXTRAN sphere. The same is true in the case of a point detector embedded in the scattering medium. DXTRAN can be a help in a problem, but it is not a cure-all for all difficulties in sampling. It should be used selectively.

Analogous to the PDn card for the detectors to control where contributions are made from, the DXCPN (for neutrons) and DXCPP (for photons) cards are available for the DXTRAN routine. The DD card that can be used with detectors also works for DXTRAN.

There are currently two restrictions imposed:

(1) there is a limit of five pairs of non-overlapping spheres; i.e., the outer spheres are disjoint.

(2) the importance Q has a fixed value of 5

In many applications, the inner sphere will be taken at least as large as the region over which one wishes to tally. As a rule of thumb, $R_O - R_I$ may be taken to a distance of one mean free path for particles of average energy. However, one should consider the physics of the particular problem in setting these radii, and perhaps make a few short runs to aid in this process.

The DXTRAN spheres need not define actual geometric cells in the problem. Usually the outer sphere will not be an actual problem surface, but in many applications the inner sphere will be.

H. *Energy Splitting*

In some cases, particles become more important to a problem after their energy has dropped to a certain point. For example, it may be difficult to calculate the number of ^{235}U fissions because the thermal neutrons are also being captured and there are then not enough thermal neutrons available for a reliable sample. In this case, once a neutron falls below a certain energy level it can be split into several neutrons with an appropriate weight adjustment. A second example involves the effect of fluorescent emission following photoelectric absorption. With energy splitting the low-energy photon track population can be built up rather than rapidly depleted as would occur naturally with the high photoelectric absorption cross section.

MCNP allows for a particle to be split at up to five different energy levels. Energy splitting with Russian roulette, however, is not played because of an increase in energy from events such as upscattering or fission.

Energy splitting may increase as well as decrease tally variances. At present, the MCNP weight cutoff game does not take into account whether a particle has undergone splitting or not.

With only a minor modification to MCNP, the mechanics for energy splitting may be used for time splitting.

I. *Source Biasing*

Provision is made for biasing the MCNP standard sources in both energy and direction. This allows the production of more source particles, with suitably reduced weights, having the more important energies and directions. For example, one may start more "tracks" at high energies and in strategic directions in a shielding problem, while correcting the distribution by altering the weights assigned to these tracks. Sizable reductions in the variance may result from energy and directional biasing of the source.

If negative weight-cutoff values are used, then the weight cutoff is made to be relative to the lowest value of source particle weight generated by the biasing schemes.

1. Energy

The SBIAS input card determines source energy biasing by establishing the frequency at which source particles will be produced in the various energy groups. If this fictitious frequency does not correspond to the fraction of actual source particles in an energy bin, the corrected weight of the source particles in a particular bin is determined by the ratio of the actual frequency (defined on the SPROB card) divided by the fictitious frequency (defined on the SBIAS card). The total weight of particles started in a given energy interval is thus conserved.

2. Direction

Direction biasing (defined by parameters ν and p on the SRCn input cards) is limited to be with respect to the y-axis. The direction cosine with the y-axis, v, is sampled uniformly within the cone $\nu < v < 1$ with probability p and within $-1 < v < \nu$ with the complementary probability $(1 - p)$. The weights assigned are $W(1 - \nu)/(2p)$ and $W(1 + \nu)/[2(1 - p)]$, respectively. The sampling of the direction cosines u and w with the x and z axes is not biased.

A word of warning - one should never bias a source completely. For example, if $p = 1$ on a SRC1 card, all particles would be started within the cone $\nu < v < 1$. Such a source is no longer an isotropic point source. Secondly, note that for a very small cone defined by ν and a high probability p for being within the cone, the few source particles generated

outside the cone will have a very high weight which can severely perturb a tally.

X-6 has UPDATE patches available to varibly bias a source with a smooth function rather than into a fixed cone. A patch is also available to bias in a specified direction rather than in the y-direction.

J. *Correlated Sampling*

MCNP provides for correlated sampling to estimate the change in a quantity resulting from a small perturbation of any type in the problem. This technique enables the evaluation of small quantities that would otherwise be masked by the statistical errors of uncorrelated calculations. MCNP correlates a pair of runs by providing each new history in the unperturbed and perturbed problems with the same initial pseudo-random number and thus the same sequence of subsequent numbers until the perturbation causes the sequences to diverge. This is done by incrementing the random number generator at the beginning of each history by 4297 random numbers, from the beginning (not end) of the previous history, a quantity greater than would be needed by any reasonable history.

VIII. CRITICALITY CALCULATIONS

Calculations of k_{eff} consist of determining the expected number of fission neutrons at several separate iterative steps, called generations or cycles. The calculation uses an initial neutron source distribution which is either obtained from a deterministic calculation or a guess. The initial distribution is used as the neutron source for the first cycle. During a cycle a group of source particles (called source size) is individually followed until termination. Particle termination results from either an anticipated fission event or loss from the system. The resulting progeny from fission events is used to construct the neutron source for the next cycle. After all source particles for the given cycle have been processed, k_{eff} is calculated by dividing the total number of fission neutrons produced by the nominal source size. The distinction between source size and nominal source size is that the former is the number of source points stored for a cycle and the latter is the user input number of source points for a cycle and also the total weight started for each cycle.

Any number of cycles can be used in the iteration process and calculated k_{eff} values are averaged over previous cycles.

No external sources, such as the MCNP standard sources or a user-provided source, can be used when the KCODE option is used.

A. Initialization Stage

In the initialization stage the user supplies an initial guess for k_{eff} and for the initial spatial distribution of the source. The source is specified discretely as a set of m points (x,y,z). The initial weight of each particle is the nominal source size n (provided by the user) divided by m and adjusted by the following recipe to reduce the range of particle weights between cycles. Each source point is repeated an integral number of times which is equal to the nominal size n divided by the number of initial source points m. (Note that the spatial distribution has not been changed, only the weights.) After spatial and weight distributions have been determined, each source point is then given an energy by selection from a standard thermal fission distribution.

It is possible to use at the initialization stage a source from another problem. The details for this operation are described on page 171, but basically MCNP keeps only those source points (weight and energy) which appear in valid fissile regions. An initial guess at the eigenvalue is still required input. At this point the problem is ready for the calculation stage.

B. Calculation Stage

In the calculation stage, the prompt or total eigenvalue and the neutron removal lifetime are calculated in an iterative process called generations or cycles.

Seven features of the calculation stage are discussed in this section. The features are spatial point selection, fission energy selection, eigenvalue estimators and weight reduction, removal lifetime estimators, scattering nuclide selection, normalization, and the estimator correlation.

The initial spatial point is chosen by sequentially picking from the input source distribution. The target nucleus is determined by standard MCNP algorithms and the initial direction is isotropic. In subsequent generations the collision points determine the spatial source points. Source points for the next generation are stored at the collision point if the number of expected fission neutrons is large enough. If e is the total

expected number of fission neutrons and c is the current eigenvalue, define a ratio $e/c = j + p$ where j is an integer and p is the remainder ($p \leq 1$). Then the number of source points stored in the bank is j with probability $(1 - p)$ and $(j + 1)$ with probability p. Each source point stored has the coordinates (x,y,z) of the collision point as well as the collision cell.

Source energies for the first generation are selected from a standard thermal fission spectrum or from a previous problem as discussed in the initialization stage. For subsequent generations, the energy of a fission neutron is chosen in the same manner as the regular fission process of MCNP. That is, although the weight of fission is deterministic, selection of the fissionable nuclide and the fission is performed using standard MCNP algorithms in order to choose the energy of the fission neutron for the next generation.

In MCNP three estimators for the eigenvalue are used, namely collision, absorption, and track length, and each is employed for every source point. The advantage of each estimator will vary with problem specification – composition, geometry, weighting scheme, etc., and no one estimator will be advantageous for all problems. However, the availability of more than one estimator permits an additional opportunity for variance reduction. If one considers two estimators of approximately equal variance and negligible correlation a simple average will have about one-half the variance of either estimate alone. If the two estimators are significantly correlated or have radically differing variances, little or nothing will be gained by such an average; however, if the two estimators happen to be anticorrelated the potential variance reduction may exceed a factor of two. Estimator correlation is discussed later.

As implemented in MCNP, neutron termination or weight reduction is separated from the selection procedure for source sites. The collision estimate for k_{eff} is calculated by binning the increment

$$XF_c \equiv \left[\frac{\sum_k f_k \, \bar{\nu}_k \, \sigma_k^F}{\sum_k f_k \, \sigma_k^T} \right] WGT$$

where at the incident energy

σ_k^T = total microscopic cross section for k^{th} nuclide,
σ_k^F = fission microscopic cross section for k^{th} nuclide,
$\bar{\nu}_k$ = average number of neutrons/fission for k^{th} nuclide,
f_k = atomic fraction for k^{th} nuclide, and

WGT = neutron weight.

The sum k is over all fissionable nuclides in the material of the cell where the collision occurs. The running total of the expected number of fission neutrons from all collisions, SUMK(1), is now computed as SUMK(1) = SUMK(1) + XF_c. Source-site selection then takes place as previously described, but no weight reduction is made at this point.

Since some nuclide *k* has been selected as the target nucleus, either of two optional processes can occur. If analog termination has been selected, the neutron history is terminated with probability $(\sigma_k^A + \sigma_k^F)/\sigma_k^T$. If the history is terminated, an absorption estimate for k_{eff} is obtained by binning the increment

$$XF_a = [\bar{\nu}_k(E)\sigma_k^F/(\sigma_k^F + \sigma_k^F)]*WG\ T \text{ where}$$

σ_k^a = absorption microscopic cross section for k^{th} nuclide.

The running total of the expected number of fission neutrons from absorption, SUMK(2), is now computed as SUMK(2) = SUMK(2) + XF_a. If on the other hand weight reduction is employed the neutron weight is reduced by the increment

$$WGT' = WGT*(\sigma_k^a + \sigma_k^F)/\sigma_k^T$$

and the absorption estimate of k_{eff} is determined by binning the above equation for XF_a using WGT' instead of WGT. Similarly, the running total of expected fission neurons from fission is incremented.

In addition to the collision and absorption estimators, a track length estimate of k_{eff} is computed by binning the increment

$$XF_t \equiv WGT * \rho d \sum_k \nu_k\ \sigma_k^F\ f_k$$

where ρ = atomic density of the cell, and
d = track length in the cell.

The running total of the expected number of fission neutrons from the track estimator, SUMK(3), is now computed as SUMK(3) = SUMK(3) + XF.

Paralleling the collision and absorption eigenvalue estimators are estimators for the removal lifetime. The removal lifetime is defined as

the average lifetime of a fission neutron from birth to termination in absorption, fission, or escape and is useful in approximating the average spectrum energy and the "α" time eigenvalue. In the case of the collision estimator the removal lifetime is calculated by binning

$$RLT(1) = \left[\frac{\sum_k f_k \ (\sigma_k^A + \sigma_k^F)}{\sum_k f_k \ \sigma_k^T} \right] WGT * TME$$

where TME is the time at removal. In the case of the absorption estimator the removal lifetime is calculated by binning either RLT(2) = WGT*TME upon analog terminator or RLT(2) = WGT'*TME (for survival biasing). Both the collision and absorption estimates RLT(1) and RLT(2) are accumulated for each source point and are normalized after all source points are processed for the given cycle.

Regardless of the estimators employed, selection of a reaction nuclide for each source point is the same. The reaction nuclide is picked by deleting the fission cross sections from the total cross section. That is, the normalized probability, p_j of colliding by nuclides in a mixture,

$$p_j = \frac{f_j * \sigma_j^T}{\sum_k \sigma_k^T \ f_k}$$

is redefined as

$$p_j = \frac{f_j * (\sigma_j^T - \sigma_j^F)}{\sum_k (\sigma_k^T - \sigma_k^F) \ f_k}$$

where k is summed over all nuclides in the cell and the fission cross section for a non-fissile nuclide is defined as zero. Note that the selection of the reaction nuclide is now identically the same as in the fixed source case - only the fission is treated as an absorption, as described above, and not a scattering event.

Once a reaction nuclide has been chosen, the next step is to select a non-fission reaction for the nuclide.

The probability p_i of picking the i^{th} non-fission reaction is given as

$$p_i = \sigma_i^R / (\sigma^T - \sigma^c - \sigma^F)$$

where σ^R_i is the i^{th} microscopic reaction cross section, and σ^c is the microscopic capture cross section. The non-fissile scattering reaction is then sampled and at this point the calculational flow is the same as a fixed-source MCNP calculation.

After all of the source points have been processed, a new value for the i^{th} estimator of k_{eff} for this generation is defined as

$$k^I_{eff} = SUMK(I)/NSRCK$$

where SUMK(I) is the total number of expected number of fission neutrons from the i^{th} estimator, and
NSRCK is the nominal source size

Similarly the i^{th} estimate for the revmoval lifetime is given as

$$RLT(I) = RLT(I)/NSRCK$$

At this point there are three estimates of the eigenvalue and two estimates of the removal lifetime for this generation or cycle. Several operations are performed on the estimates and their covariances as is discussed in the next paragraph; however, the calculation could proceed to the next generation. The generation number is incremented so that the source points stored in the last generation now become the source of the current generation. These points are all given a new weight WTO

$$WTO = NSRCK/NSA$$

where NSA is the number of source points stored in the last generation. Control then returns to sampling the new source.

As previously stated, several operations are performed on the eigenvalue and removal lifetime estimates. These include cycle averages, averages of the estimates by pairs, and average of the three estimates of the eigenvalue. The utility of these operations depends on the estimator correlation. In many cases the absorption estimator will be only weakly correlated with either the collision or track length estimator and maybe significantly anticorrelated. Under these conditions, the optimum estimator will be simple average of the absorption estimator with one of the other two. Except in the most heterogeneous systems, the collision and track length estimator are likely to be so strongly correlated that little gain may be expected by averaging them. A simple average of all estimators

may be advantageous under some circumstances, but is not likely to produce an additional variance reduction. When a weight reduction rather than an analog scheme is employed, the correlation between the absorption estimator and the others will be increased and the advantage of averaging reduced. Some gain can be achieved with a well-chosen weight-reduction strategy.

In addition to the simple averaging of estimators, linear combinations of estimators are computed using the properties of the sample covariance matrix. In the case of small or negative correlation with variances that are significantly different, the method of combined estimators may produce a modest gain. On the other hand, if two estimates with variances σ_1^2 and σ_2^2 are such that $\sigma_2 > \sigma_1$ and if the sample correlation coefficient r is greater than σ_1/σ_2, the combined estimate will be outside the interval defined by the two individual estimates. Using the simple and combined averages, confidence intervals can be constructed.

C. Multiplication Factor

The MCNP multiplication factor M that is printed out on the Creation and Loss summary page differs from the k_{eff} you may be familiar with from nuclear engineering and from the above criticality code.

In reactors, the multiplication factor is the ratio of the number of neutrons in one generation to the number in the preceding generation. This is also equal to $\bar{\nu}$ times the number of fission events in the earlier generation, where $\bar{\nu}$ is the average number of neutrons emerging per reaction of type (n,2n), (n,3n), fission, and all other neutron multiplying reactions. This assumes that $\bar{\nu}$ is the average $\bar{\nu}$ for all the materials in the system.

In a regular, fixed source MCNP calculation (i.e., not a KCODE calculation), the net multiplication in the Creation and Loss summary table is the source weight plus net fission weight increase plus net (n,xn) weight increase divided by the source weight. This is summed over all generations. Therefore, the MCNP net multiplication M comes from the gain in weight (or profit) from the various reactions relative to the source weight because a fission or (n,xn) reaction is not counted as a loss in the summary table.

Consider one source particle of net increase in the weight 1. Then the net increase in the weight of neutrons in the second generation is just 1 times the number of fission events times the fission profit, or

$$\frac{k_{eff}}{\bar{\nu}} (\bar{\nu} - 1)$$

The weight of particles in the third generation is the weight in the second generation times k_{eff}, or

$$M_{MCNP} = 1 + \frac{k_{eff}}{\bar{\nu}} (\bar{\nu} - 1) + \frac{k^2_{eff}}{\bar{\nu}} (\bar{\nu} - 1) + \cdots$$

$$= 1 + \frac{\bar{\nu} - 1}{\bar{\nu}} k_{eff} (1 + k_{eff} + \cdots)$$

$$= 1 + \frac{\bar{\nu} - 1}{\bar{\nu}} k_{eff} \frac{1}{1 - k_{eff}}$$

$$= \frac{1 - \frac{k_{eff}}{\bar{\nu}}}{1 - k_{eff}}$$

The only assumption made here is that the system contains fissionable material but is subcritical (but a regular MCNP calculation will only handle subcritical assemblies anyway). Super-critical systems must be calculated with the KCODE option.

IX. VOLUMES AND AREAS[16]

The particle flux in Monte Carlo transport problems is often estimated as the track length per unit volume or the number of particles crossing a surface per unit area. Therefore, knowledge of the volumes and surface areas of various geometric regions in a Monte Carlo problem is very

important. Knowledge of volumes is also useful in calculating the masses and densities of problem cells and thus in calculating volumetric or mass heating. Furthermore, a calculation of the mass of a geometry is frequently a good check on the accuracy of the geometry setup when the mass is known by other means.

The calculation of volumes and surface areas in modern Monte Carlo transport codes is non-trivial. In particular MCNP allows for cells to be constructed from the unions and/or intersections of any regions defined by an arbitrary combination of second degree surfaces and/or toroidal fourth degree surfaces. These surfaces may have different orientations, they may be segmented for tallying purposes, or the cells they compose may even consist of several disjoint subcells. Although such generality greatly increases the flexibility of MCNP, computing cell volumes and surface areas understandably requires increasingly elaborate computational methods.

MCNP automatically calculates volumes and areas if they have been generated by surfaces of revolution about any axis, even a skew axis. For other volumes and areas, a stochastic method is outlined that uses MCNP for ray tracing.

A. Symmetric Volumes and Areas

A volume and surface area algorithm has been developed for MCNP which deals with a large class of widely used geometries and includes the following features:

- *surface areas are computed;
- *cells are not required to be rotationally symmetric about the y-axis of the MCNP coordinate system: they may have rotational symmetry about any axis, even a skewed axis. (The restriction to rotational symmetry is not a significant limitation for most MCNP problems.)
- *volume and most surface area integrals are computed by formula rather than by numerical integration;
- *cells consisting of the union of several disjoint subcells may be calculated;
- *arbitrary segments of cells and surfaces (which satisfy the symmetry requirement) may also be calculated for tallying purposes.

The procedure for the new volume and surface area calculation is as follows:

1. All surfaces bounding a given cell are identified. Second degree

surfaces in the MCNP (x,y,z) Cartesian coordinate system are put into the generalized form

$$Ax^2 + By^2 + Cz^2 + Dxy + Eyz + Fxz + Gx + Hy + Jz + K = 0$$

For toroidal surfaces this step is a more complicated special case.

2. The (x′,y′,z′) coordinate system in which the cell is rotationally symmetric must be identified if it exists. This procedure is not straight-forward when the bounding surfaces of the cell are not symmetric about a single axis parallel to a coordinate axis. In the case of a skew axis, the above equation must be rewritten in matrix form and then diagonalized. A special translation method has been developed for parabolic cases in which resulting singular matrices cause the standard procedure to fail.

3. All surfaces bounding a cell are rotated and translated into the (x′,y′,z′) coordinate system so that the equation is of the two-dimensional cylindrical form

$$ar^2 + br + cs^2 + ds + e = 0 \quad (r^2 = x'^2 + y'^2;\ s = z')$$

or

$$r = f(s)$$

4. The intersections of all bounding surfaces with each other are found, but only those intersections which are corners of the cell are kept. How an intersection is identified as a corner is illustrated in Figure 2.18 where the intersection of two surfaces is sketched in the (r,s) coordinate system. The two surfaces divide space into four zones and a cell could conceivably be within any combination of zones. The intersection of the two surfaces defines a corner only if

$$f = \sum_{i=1}^{4} \delta_i \cdot 2^{(i-1)}$$

is not divisible by 3. Here,

$$\delta_i = 0 = \text{cell not present in zone } i$$
$$= 1 = \text{cell present in zone } i$$

For example, a cell present in zones 1 and 3 but not in zones 2 and 4

($\delta_1 = \delta_3 = 1$; $\delta_2 = \delta_4 = 0$) can be identified as a corner because f = 5.

5. The surfaces are integrated (using standard integration formulas) between corners as

$$V_i = \pi \int r^2 \, ds \qquad \text{for volumes;}$$

$$A_i = 2\pi \int r \sqrt{1 + \left(\frac{\partial r}{\partial s}\right)^2} \, ds \qquad \text{for surface areas;}$$

(Only for toroidal surfaces must A_i be computed by numerical integration.)

6. The integrals are appropriately added and subtracted to determine the total volume of each cell and the total active area of each surface. The area integrals are actually computed twice; once for each side of the surface. In this way rotationally symmetric surfaces bounding some

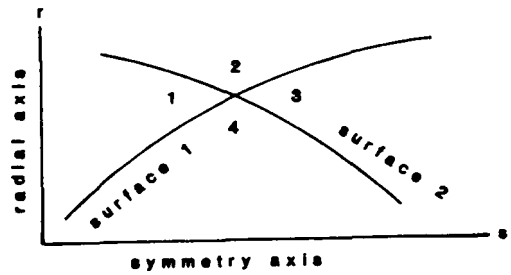


Figure 2.18

non-symmetric cells may still be considered.

B. Non-Symmetric Volumes and Areas

For cells and surfaces where there is no rotational symmetry a stochastic computation is possible. The procedure is to construct a sphere around the geometric region of interest and to start particles on the sphere inward with a cosine distribution (see the SRC3 source described on page 134). An F4 tally is made which produces the cell volume, and an F2 tally produces the surface area. After the volumes and areas of interest are determined, they are added to the INP file on the VOL and AREA cards for a subsequent run with MCNP.

The SRC3 source is started on a sphere with radius large enough to encompass the geometry of interest, and all the cells in the region are defined as voids. This produces a uniform isotropic flux of unit strength inside the sphere, and the value of the F4 tally is just the cell volume with the value of the F2 tally being the surface area for the surfaces listed on the F2 card. The energy of the starting particles is immaterial. The SRC3 source is specified with J set to the number of the spherical surface and all other parameters ignored (i.e., taken as default). The importance of all cells inside the source sphere should be unity. For cells outside the sphere, the importance must be zero.

This method will generally give good results for volumes and areas in a small amount of running time, but the source sphere should be as small as possible to increase the efficiency of the calculation. Note that if a surface area or cell volume inside the source sphere is computed deterministically by MCNP, then the F2 and F4 tallies are divided by these computed areas and volumes. Thus the stochastically computed F2 and F4 fluxes will approach unity for these surfaces and cells.

X. PLOTTER

The mechanism of the plotter makes use of the fact that all first and second order surfaces used in MCNP may be expressed in the following forms. The general quadratic GQ is normally represented as

$$Ax^2 + By^2 + Cz^2 + Dxy + Eyz + Fzx + Gx + Hy + Jz + K = 0$$

One may also use a real, symmetric matrix representation for the same surface:

$$\begin{bmatrix} 1 & X & Y & Z \end{bmatrix} \begin{bmatrix} K & G/2 & H/2 & J/2 \\ G/2 & A & D/2 & F/2 \\ H/2 & D/2 & B & E/2 \\ J/2 & F/2 & E/2 & C \end{bmatrix} \begin{bmatrix} 1 \\ X \\ Y \\ Z \end{bmatrix} = 0$$

From the representation of the general quadratic, one may readily infer the forms of other surfaces used in MCNP. Using a more concise notation, any surface may be represented as

$$[1\ R]\ A \begin{bmatrix} 1 \\ R \end{bmatrix} = 0$$

where A represents the 4X4 symmetric matrix and R represents the spatial coordinates.

Generally, the geometry in any MCNP problem is quite straight forward and unambiguous. One knows the position of the origin and the orientation of the coordinate axes with respect to the surfaces of the problem. The plot plane is defined using this coordinate system. Two non-colinear basis vectors $\vec{a}$ and $\vec{b}$ are introduced, these defining the orientation of the plot plane and lying in it. An origin vector $\vec{r}_o$ is introduced, and the plot plane is required to pass through this point. Again, these three vectors are defined within the original problem coordinate system. Introducing plot plane coordinates s and t, the equation of the plot plane in the problem coordinate system may be written as

$$\vec{r} = \vec{r}_o + s\vec{a} + t\vec{b}$$

Introducing components and rewriting in matrix notation, the relation becomes

$$\begin{bmatrix} 1 \\ x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ x_o & a_x & b_x \\ y_o & a_y & b_y \\ z_o & a_z & b_z \end{bmatrix} \begin{bmatrix} 1 \\ s \\ t \end{bmatrix}$$

Using concise notation, the matrix equation becomes

$$\begin{bmatrix} 1 \\ r \end{bmatrix} = PL \begin{bmatrix} 1 \\ R \end{bmatrix}$$

where PL represents the 4X3 matrix and R is the two-dimensional plot plane coordinates.

By inspection, it is obvious that

$$[1\ r] = [1\ R]\ PL^T$$

with PL^T representing the transpose of PL. Thus, the intersection of the plot plane with a three-dimensional surface is given by those points satisfying the relation

$$[1 \; R] \; PL^T \; A \; PL \begin{bmatrix} 1 \\ R \end{bmatrix} = 0$$

which is a general quadratic expression in s and t, the plot plane coordinates. Let

$$Q = PL^T \; A \; PL$$

where Q is a 3X3 symmetric matrix.

At this point, the equation of the curve in the plot plane has been determined. However, having only one equation in the variables s and t, there is no consistent way of generating the sets of points that lie on the curve. What is needed is a one-parameter set of curves of the form $s = s(p)$, $t = t(p)$, $-\infty \leq p \leq \infty$. It is obvious that the variety of curves occurring in the plot plane is limited to the conic sections which reduce to four distinct equations. In order to identify the conic section associated with the curve, introduce a general translation and rotation in the plot plane defining a new coordinate system, u and v, such that

$$\begin{bmatrix} 1 \\ s \\ t \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ s_o & \cos\theta & -\sin\theta \\ t_o & \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} 1 \\ u \\ v \end{bmatrix}$$

Let

$$DIA = \begin{bmatrix} 1 & 0 & 0 \\ s_o & \cos\theta & -\sin\theta \\ t_o & \sin\theta & \cos\theta \end{bmatrix}$$

The parameters of the transformation, namely s_o, t_o, and θ, are determined by the condition

$$DIA^T \; Q \; DIA = \text{a diagonal matrix.}$$

The diagonalization of the Q matrix introduces a coordinate system, namely in u and v, in which the equations of the conic sections have their simplest and most symmetric form. One can then establish a one-parameter set of relationships of the form $u = u(p)$, $v = v(p)$ where $-\infty \leq p \leq \infty$. Using the transformation DIA, s and t are then determined as a one-parameter set of curves.

In essence, the plotting routine checks all problem surfaces for the existence of an intersection with the plotting plane. For those surfaces having intersections, a closely spaced set of points, s and t, is generated by advancing the parameter p and checked to see if they are within the extent of the plotting frame and have the proper sense with respect to the cells bounded by the particular surface. It is these edited sets of points that form the output to a particular plotting device.

In the case of the torus, the method outlined above is not applicable, in general. However, for certain orientations of the plot plane the intersections reduce to a form equivalent to that associated with second order equations. Only these configurations are plotted in the case of the torus.

CHAPTER 3
DESCRIPTION OF MCNP INPUT

The input to MCNP consists of several files, but the main one as far as a user is concerned is the INP (the default name) file which contains the input information necessary to describe the problem.

Throughout this chapter maximum dimensions will be given for various MCNP input items, and they are all summarized at the end of this chapter on page 176. Parts of MCNP are variably dimensioned, but where this will interfere with speed, the dimensions are fixed. If the user wishes to increase any of these maximum values, he can do so by altering the code and re-compiling. Instructions for doing this are described in Appendix A.

MCNP does extensive input checking, but it is not foolproof. It is not logically possible for MCNP to catch geometry specifications which are consistent but topologically incorrect. This is particularly true with the geometry specification (especially if otherside cells are not specified, see page 20). A geometry should not be used without first looking at it from several different views with the PLOT overlay and then running a short job and studying the output to see if you are calculating what you hope you are calculating.

All features of MCNP should be used with caution and understanding. This is especially true of detectors and variance reduction schemes, and you are encouraged to read the appropriate sections of Chapter 2 before using them.

The units used throughout MCNP are

(1) Lengths in centimeters
(2) Energies in MeV
(3) Times in shakes (10^{-8} sec)
(4) Temperatures in MeV
(5) Atomic densities in units of 10^{24} atoms/cm^3
(6) Mass densities in gm/cm^3

I. INP FILE

The INP file can have two forms: (1) initiate-run and (2) continue-run.

A. Initiate-Run

This form is used to set up a Monte Carlo problem (describe geometry, materials, tallies, etc.) and run if desired. The initiate-run file has the following form:

Title Card
Cell Cards
.
.
.
Blank Line Delimiter
Surface Cards
.
.
.
Blank Line Delimiter
Data Cards
.
.
.
Blank Line Terminator

The first card of the file is the problem title card and is required. It is limited to one line from columns 1-80 and is used as a title in various places in the MCNP output.

B. Continue-Run

Continue-run is used to continue running histories in a problem that was terminated earlier - for example, to run the job two hours on one night and then to run it an additional hour some later night. It can also be used to re-construct the output of an initial run.

In general, two files in addition to the C. option on the MCNP execution line (see Appendix A) are needed for this procedure: (1) the continue-run input file and (2) the run file RUNTPE (this is the default name). The run file is produced by MCNP in the initiate-run sequence and contains the geometry, cross sections, problem parameters, and all other necessary information to restart the job. The continue-run input file must

have the word CONTINUE starting in column one of the first line. The file has the following form:

```
CONTINUE
Data Cards
.
.
.
Blank Line Terminator
```

The data cards allowed in the continue-run file (default name is INP) are a subset of the data cards available for an initiate-run file. The allowed continue-run data cards are CUTN, CUTP, CTME, PRDMP, NPS, LOST, DBCN, and PRINT (see pages 162 through 170).

A very convenient feature is that if none of the above items is to be changed, then the continue-run input file is not required - only the run file RUNTPE plus the C. option on the MCNP execution line. For example, if you run a job for a minute or so and it quits on time limit but you want a few more particles run, simply type in

```
MCNP C. / t p
```

and the job will pick up where it stopped and continue until another time limit or particle cutoff is reached or until you stop it manually with CTRL-E IQ at some point. This example assumes that a run file called RUNTPE from the initial run is in your local files.

The complete continue-run execution line option is C. m, where m specifies which dump to pick up and continue with. By not specifyning m, the last dump is taken by default.

If the initial run producing the RUNTPE was stopped because of particle cutoff (NPS card, page 164), then NPS must be increased for a continue-run. The NPS card refers to the total number of histories to be run, including preceding continue-runs and the initial run.

In a continue-run, entering a negative number on the NPS card will produce a print output file at the time of the previous dump. No more histories will be run. This can be useful when the printed output has been lost or you want to alter the content of the output with the PRINT card.

Caution should be used if you are using a FILES card in your initial run, see page 169.

C. Card Format

All input lines (we will refer to "cards" in this manual) of INP are limited to columns 1–72 and consist of card images. Columns 73–80 may be used for comments. The title card is limited to one line and can contain any information the user desires, even blank. It usually contains information describing the particular problem. Note that a blank card is used as a delimiter and as a terminator.

Comment cards may be used anywhere in the INP file after the problem title card and before the last blank terminator card. These cards must have the letter C in the first column of the card followed by four blanks, and then columns 6–80 are available to the user for any comments. Unlike the title card, these cards are printed only with the input file listing and are not printed anywhere else in the MCNP output file.

Cell cards, surface cards, and data cards all conform to the same format. Columns 1–5 are reserved for the name (or number) associated with the card. The name (or number) field can appear anywhere in columns 1–5. Blanks in these columns indicate a continuation of the data from the last named card. Columns 6–72 are for free–field format data entry associated with the card name. With some exceptions on cell cards, separation between data entries is by one or more blank columns. In general, data entries may be of any type (fixed point, floating point, octal, or exponential) inasmuch as MCNP makes the appropriate conversion.

Two features have been incorporated into the code to facilitate card preparation:

1. nR which means repeat (R stands for repeat) and also include the immediately preceding entry on the card n times, and

2. kI which means insert k linear interpolates (I stands for interpolate) between the entries immediately preceding and following this feature.

These features apply to both integer and floating point quantities and may be used wherever applicable. As an example, 1 3R 4I 6 on a card is the same as 1 1 1 1 2 3 4 5 6 .

A duplication of any cards in the INP file is not permitted.

II. CELL CARDS

The problem number of the cell is in columns 1–5. Columns 6–72 will contain, in the following order,:

(a) the cell material number. This material is described by the material card (see page 154) with the same material number. If the cell is a void, a zero should be entered for the material number.

(b) the cell material density. A positive entry is interpreted as the atomic density in units of 10^{24} atoms/cm^3. A negative entry is interpreted as mass density in units of gm/cm^3. Nothing is entered for a void cell; the list in (c) is started immediately.

(c) a complete specification of the geometry of the cell. This specification includes a list of the signed (see below) surfaces bounding the cell and the relation between the regions defined by the surfaces. This latter relation is indicated by the intersection and/or union of the regions. At the user's discretion, each surface listed may or may not be followed by a list of all otherside cells. Any surface or cell numbers on this card are problem numbers. If an ambiguity surface (see page 13) is required to define a cell, its signed problem number is added as an entry without a list of otherside cells.

A blank space delimits items in the above categories. Commas are used to separate the otherside cell list.

As an example of the above format, if cell C_1 of material 16 (defined on an Mm card) with density 4.2 gm/cm^3 and bounded by three surfaces S_1, S_2, and S_3 is being specified,

$$C_1 \quad 16 \quad -4.2 \quad S_1,C_2,C_3 \quad -S_2,C_5 \quad S_3,C_4$$

where C_1 and C_3 are cells on the other side of S_1 from C_1 (and points in C_1 have a positive sense with respect to S_1), C_5 is a cell on the other side

of S_2, and C_4 is on the other side of S_3. This is illustrated by a

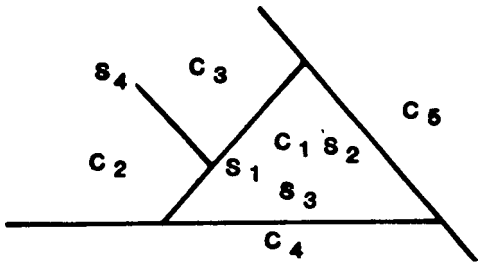


Figure 3.1

geometry such as in Figure 3.1.

The problem numbers referred to above may be chosen by the user in any order he wishes. However, MCNP assigns the cells monotonically increasing integer values (beginning with one) as the cell cards are read in. These integers are referred to as program cell numbers. The user needs to be aware of the distinction between problem and program cell numbers since subsequent data cards will refer to one or the other.

As an example of problem versus program numbers, you may have numbered your first four cells in the input as 1, 2, 16, and 4. These are the problem numbers. However, MCNP internally reassigns sequential numbers to these cells as 1, 2, 3, and 4. These are the program numbers. Perhaps the safest approach and best way to avoid confusion is to keep the problem and program numbers the same.

The problem numbers of the surfaces bounding a cell are signed quantities, the sign being determined by the sense (see page 13) that any point within the cell has with respect to the surface bounding the region containing the point. If the sense is positive, the sign may be omitted.

In the list consisting of a problem surface number followed by the problem numbers of the cells on the other side, each entry except the last must be followed immediately by a comma. The absence of the comma (i.e., at least one blank) indicates that another bounding surface follows with its attendant cells on the other side.

The maximum number of cells allowed is MAXA=175. The maximum number of surface numbers entered on all cell cards is 7*MAXA. The maximum number of cell numbers entered with all surfaces on the cell cards is 12*MAXA.

III. SURFACE CARDS

The problem number of the surface appears in columns 1-5. Columns 6-72 contain, in the following order,

(a) an alphabetic mnemonic indicating the surface type, and

(b) the required card entries for the specific surface in proper order.

MCNP permits any surface appearing in the problem to be a reflecting surface except a surface used as an ambiguity surface. To designate a reflecting surface, columns 1-5 on the surface card must contain an asterisk in addition to the surface number. A particle hitting such a surface is specularly reflected and the calculation continues. A detector should not be used with a reflecting surface, see page 63.

As with cell cards, MCNP assigns the surfaces monotonically increasing integer values as the surface cards are read. These integers are referred to as program surface numbers in contrast to the problem surface numbers designated on the cell and surface cards. The maximum number of surfaces allowed is JMAX=175. The maximum number of surface coefficients allowed on all surface cards is 5*JMAX.

A. Surfaces Defined by Equations

The surface types, equations, their mnemonics, and the order of the card entries are given in Table 3.1. The order of the surfaces listed in Table 3.1 is the same order in which the surfaces appear in MCNP in the TRACK subroutine.

To specify a surface by this method, find the needed surface in Table 3.1 and then determine the required coefficients for the particular equation (you may need to consult a book on analytical geometry). The information is entered on a surface card according to the format of the above section.

Table 3.1 MCNP Surface Cards

Mnemonic	Type	Description	Equation	Card Entries
P	Plane	General	$Ax+By+Cz-D=0$	A B C D
PX		Normal to X-axis	$x-D=0$	D
PY		Normal to Y-axis	$y-D=0$	D
PZ		Normal to Z-axis	$z-D=0$	D
SO	Sphere	Centered at Origin	$x^2+y^2+z^2-R^2=0$	R
S		General	$(x-\bar{x})^2+(y-\bar{y})^2+(z-\bar{z})^2-R^2=0$	$\bar{x}$ $\bar{y}$ $\bar{z}$ R
SX		Centered on X-axis	$(x-\bar{x})^2+y^2+z^2-R^2=0$	$\bar{x}$ R
SY		Centered on Y-axis	$x^2+(y-\bar{y})^2+z^2-R^2=0$	$\bar{y}$ R
SZ		Centered on Z-axis	$x^2+y^2+(z-\bar{z})^2-R^2=0$	$\bar{z}$ R
C/X	Cylinder	Parallel to X-axis	$(y-\bar{y})^2+(z-\bar{z})^2-R^2=0$	$\bar{y}$ $\bar{z}$ R
C/Y		Parallel to Y-axis	$(x-\bar{x})^2+(z-\bar{z})^2-R^2=0$	$\bar{x}$ $\bar{z}$ R
C/Z		Parallel to Z-axis	$(x-\bar{x})^2+(y-\bar{y})^2-R^2=0$	$\bar{x}$ $\bar{y}$ R
CX		On X-axis	$y^2+z^2-R^2=0$	R
CY		On Y-axis	$x^2+z^2-R^2=0$	R
CZ		On Z-axis	$x^2+y^2-R^2=0$	R
K/X	Cone	Parallel to X-axis	$\sqrt{(y-\bar{y})^2+(z-\bar{z})^2}-t(x-\bar{x})=0$	$\bar{x}$ $\bar{y}$ $\bar{z}$ t^2 ±1
K/Y		Parallel to Y-axis	$\sqrt{(x-\bar{x})^2+(z-\bar{z})^2}-t(y-\bar{y})=0$	$\bar{x}$ $\bar{y}$ $\bar{z}$ t^2 ±1
K/Z		Parallel to Z-axis	$\sqrt{(x-\bar{x})^2+(y-\bar{y})^2}-t(z-\bar{z})=0$	$\bar{x}$ $\bar{y}$ $\bar{z}$ t^2 ±1
KX		On X-axis	$\sqrt{y^2+z^2}-t(x-\bar{x})=0$	$\bar{x}$ t^2 ±1
KY		On Y-axis	$\sqrt{x^2+z^2}-t(y-\bar{y})=0$	$\bar{y}$ t^2 ±1
KZ		On Z-axis	$\sqrt{x^2+y^2}-t(z-\bar{z})=0$	$\bar{z}$ t^2 ±1
				±1 used only for 1 sheet cone
SQ	Ellipsoid Hyperboloid Paraboloid	Major axis parallel to X, Y, or Z-axis	$A(x-\bar{x})^2+B(y-\bar{y})^2+C(z-\bar{z})^2$ $+2D(x-\bar{x})+2E(y-\bar{y})$ $+2F(z-\bar{z})+G=0$	A B C D E F G $\bar{x}$ $\bar{y}$ $\bar{z}$
GQ	Cylinder Cone Ellipsoid Hyperboloid Paraboloid	Major axis is not parallel to X, Y, or Z-axis	$Ax^2+By^2+cz^2+Dxy+Eyz$ $+Fzx+Gx+Hy+Jz+K=0$	A B C D E F G H J K
TX	Torus	Torus relative to the X, Y, or Z-axis	See discussion on page 123	$\bar{x}$ $\bar{y}$ $\bar{z}$ a b c
TY				
TZ				

As an example, a plane normal to the y-axis at y=3 is specified by PY 3 whereas K/Y 0 0 2 .25 1 specifies a cone whose vertex is at (x,y,z)=(0,0,2) and is parallel to the y-axis. The slope t of the cone is 0.5 (note that t^2 is entered) and only the positive (right hand) sheet of the cone is used. Finally,

```
GQ    1    .25   .75   0      -.866
      0    -12   -2    3.464  39
```

is a cylinder of radius 1 cm whose axis is in a plane normal to the x-axis at x=6 but rotated 30° about the origin off the y-axis toward the z-axis. This cylinder was easily determined by first starting with a cylinder around the y-axis, CY 1, and then rotating and translating it with the MOVE feature in the PLOT overlay (see Appendix B).

The TX, TY, and TZ input cards represent elliptical tori (fourth degree surfaces) rotationally symmetric about axes parallel to the x, y, and z axes, respectively. A TY torus is illustrated in Figure 3.2a. Note that the input parameters

$$\bar{x} \quad \bar{y} \quad \bar{z} \quad a \quad b \quad c$$

specify the ellipse

$$\frac{s^2}{b^2} + \frac{(r - a)^2}{c^2} = 1$$

rotated about the s-axis in the (r,s) cylindrical coordinate system (Figure 3.2b) whose origin is at $\bar{x}$, $\bar{y}$, $\bar{z}$ in the x, y, z system. In the case of a TY torus,

$$s = (y - \bar{y})$$

and $$r = \sqrt{(x - \bar{x})^2 + (z - \bar{z})^2}$$

A torus is degenerate if $|a| < c$ where $0 < a < c$ produces the outer surface (Figure 3.2c), and $-c < a < 0$ produces the inner surface (Figure 3.2d).

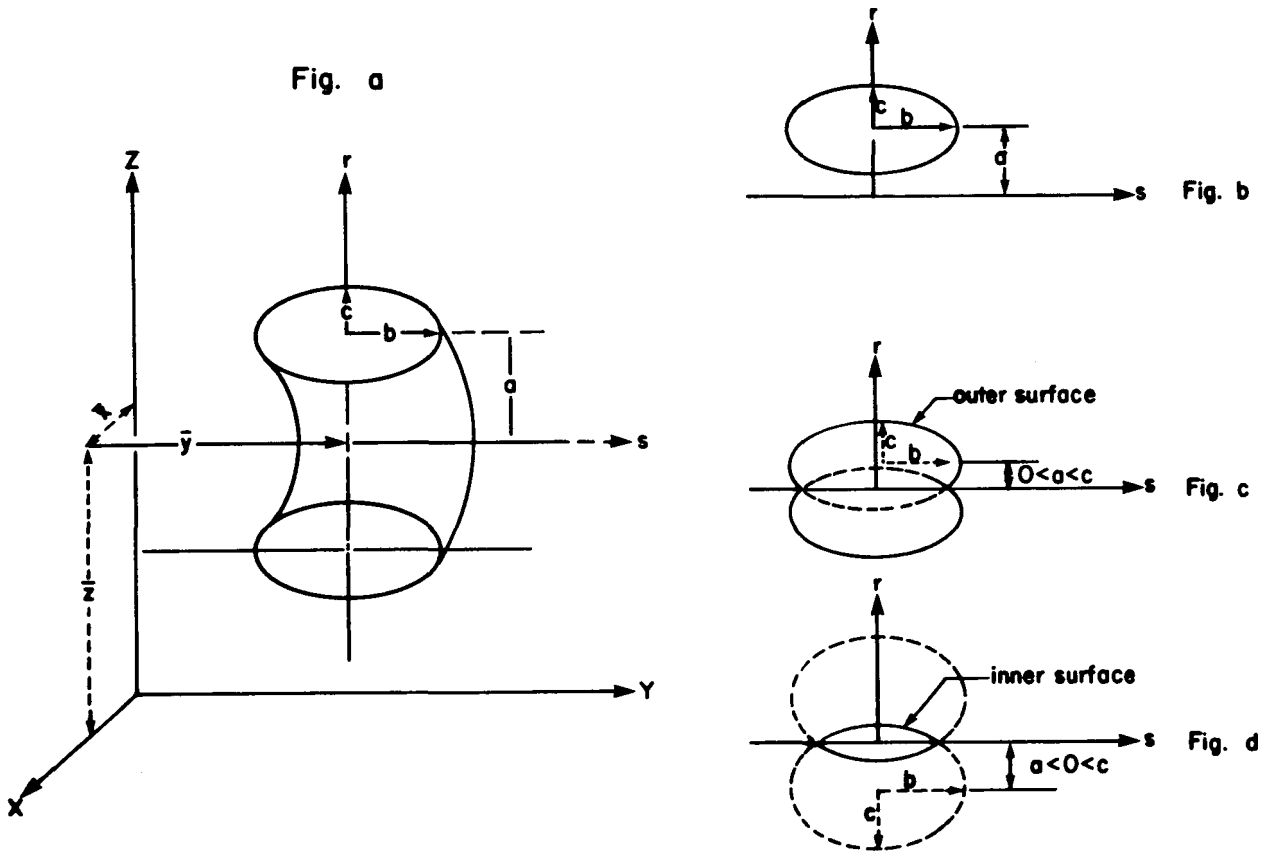


Figure 3.2 Torus

B. Surfaces Defined by Points

Surface cards X, Y, and Z may be used to describe surfaces by coordinate points rather than by equation coefficients as in the previous section. The surfaces described by these cards *must be symmetric about the x, y, or z axis, respectively*, and they must be unique, real, and continuous.

The entries on these three cards are one to three coordinate pairs, each pair defining a geometrical point on the surface. On the Y card, for example, the entries may be

Y y_1 r_1 y_2 r_2

where r_i = SQRT(x_i^2 + z_i^2) and y_i are the coordinates of point i.

If one coordinate pair is used, then a plane (PX, PY, or PZ) is defined.

If two coordinate pairs are used, then a linear surface (PX, PY, PZ, CX, CY, CZ, KX, KY, or KZ) is defined.

If three coordinate pairs are used, then a quadratic surface (PX, PY, PZ, SO, SX, SY, SZ, CX, CY, CZ, KX, KY, KZ, or SQ) is defined.

The conditions that a surface be unique, real, continuous, and axisymmetric must all be met or MCNP will reject the coordinate pairs and set a fatal error flag. Some examples may be helpful:

Example 1: X 7 5 3 2 4 3

describes a surface symmetric about the x-axis which passes through the three (x,r) points (7,5), (3,2), and (4,3). This surface is equivalent to, and in MCNP is converted to,

SQ −.083333333 1 1 0 0 0 68.52083 −26.5 0 0

Example 2: Y 1 2 1 3 3 4

describes two parallel planes at Y=1 and Y=3 and is rejected because the condition of a single, continuous surface is not met. However, the surface defined by

y 3 0 4 1 5 0

describes a sphere of radius 1 with origin at y=4 about the y-axis.

Example 3: Z 1 3 2 4 3 7

is rejected because the coordinates are on two different branches of the SQ hyperboloid

$$x^2 + y^2 - 13x^2 + 32z - 28 = 0.$$

However, the surface

Z 2 4 3 7 4 10.3923

which has the same surface equation as above is accepted because all coordinates lie on a single surface, namely the right branch of the hyperboloid.

Example 4: This final example defines a cell bounded by a cone, hyperboloid, and an ellipsoid. The three surfaces define the donut-like cell that is symmetric about the y-axis. A cross section of this cell is seen in Figure 3.3.

One surface goes through the points (-3,2) and (2,1). The second surface goes through (2,3), (3,3), and (4,2). The last surface is defined by the points (2,1), (3,1), and (4,2). These coordinate points are in the form (y,r). The surface cards are

```
1     Y  -3 2   2 1
2     Y   2 3   3 3   4 2
3     Y   2 1   3 1   4 2
```

Using these cards, MCNP indicates that surface 1 is a cone of one sheet, surface 2 is an ellipsoid, and surface 3 is a hyperboloid of one sheet. The equation coefficients for the standard surface equations are printed out for the various surfaces. For example, we see that an SQ card defining surface 3 is

```
3     SQ    1 -1.5 1 0 0 0 -.625 0 2.5 0
```

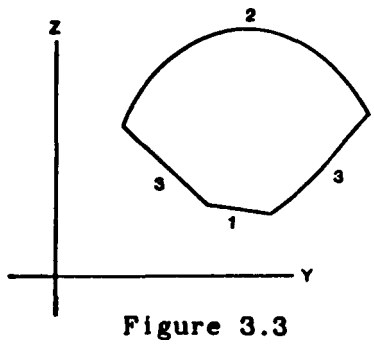


Figure 3.3

IV. DATA CARDS

All input cards to MCNP other than cell and surface cards are entered

after the second blank card delimiter. These cards fall into the following categories:

(A) Mode
(B) Cell parameters
(C) Source specification
(D) Tally specification
(E) Material specification
(F) Energy and thermal treatment specification
(G) Problem cutoffs
(H) User data arrays
(I) Peripheral cards

These card categories are described below. Only cards of category G and some from category I are allowed in a continue-run input file. No data card can be used more than once.

A. Mode (MODE) Card

The MCNP code can be run in three different modes:

Mode 0 - Neutron transport only
1 - Neutron and neutron-induced photon transport
2 - Photon transport only

The MODE card consists of the mnemonic MODE in columns 1-5 and either a 0, 1, or 2 in columns 6-72 depending on which mode is being used. If the MODE card is omitted, a mode of zero is assumed.

Gamma-production cross sections do not exist for all nuclides. If they are not available for a Mode 1 problem, XACT will print out warning messages. If an isotope has gamma-production cross sections, the letters *CPXS* are found with the isotope in the Appendix F cross-section list.

Mode 1 does not account for photo-neutrons but only neutron-induced photons. All source particles in a Mode 1 calculation must be neutrons.

B. Cell Parameter Cards

The following cards contain information specified by program cell number:

Mnemonic (Columns 1-5)	Card Type
IN	Importances, neutron
IP	Importances, photon
VOL	Cell volumes
AREA	Surface areas
PWT	Photon production weights
EXTYN	Exponential transform, neutron
EXTYP	Exponential transform, photon
FCN	Forced collision, neutron
FCP	Forced collision, photon
PDn	Detector contribution to tally n
DXCPN	DXTRAN contribution, neutrons
DXCPP	DXTRAN contribution, photons

Entries on all of the following cards correspond in order to the list of cells or, as the case may be, to the list of surfaces. To get to the particular cell(s) or surface(s) on a card you are interested in, you will need to supply the appropriate default values on the cards as spacers (the nR repeat feature may help).

Since there are no ordering restrictions of cell cards, the n^{th} entry on a cell parameter card will be the value assigned to program cell number n. The n^{th} cell entry does not necessarily correspond to problem cell number n as entered in columns 1-5 of the cell cards. The number of entries on a cell parameter card should always equal the number of cells in the problem (thus the number of entries is also limited to MAXA=175).

1. Importance (IN and IP) Cards

The importance of a cell serves two purposes:

(a) it is used to terminate the particle's history if the importance is zero, and
(b) the importances are used for geometry splitting and Russian roulette as described on page 80.

Because the user may wish to split photons differently than neutrons in a neutron-photon problem, two sets of importances can be used (IN for neutrons, IP for photons). If an IP card is not included in a Mode 1 problem, all photon importances are set to unity by default.

It is a fatal error if the number of entries on either card is not equal to the number of cells.

2. Cel. Volume (VOL) Card

The VOL card may be needed only if an F4/14, F6/16, or F7 tally is being used. The volume of each cell is entered in units of cm^3. If a volume card is not used and if a volume is not automatically calculated for a cell, a volume of 1 cm^3 is used.

Entries on the VOL card must be 0 for the cells that MCNP is to calculate the volume of.

MCNP will automatically calculate the volumes of cells whose volumes are not specified on the VOL card and which are rotationally symmetric about any axis (even a skew axis); that is, generated by surfaces of revolution. Initially all volumes are set to 0. The user can reset any (or all) or them on the VOL card. MCNP then automatically tries to calculate any volumes that are still 0. If there are some it cannot calculate, it sets them to 1. With this sequence, MCNP can be forced not to calculate any particular volumes. The volume calculator prints out the masses as well as the volumes for each cell.

If you need the volume of a segment of a cell, you may need to use the FSn and VAn cards. The VOL card refers to all cells in the geometry, whereas the VAn card refers to a segmented volume associated with a particular tally. Irregular volumes may also be calculated by using the ray-tracing technique described on page 110.

3. Surface Area (AREA) Card

This card is analogous to the VOL card and is used to supplement the automatic calculation of surface areas by MCNP. Only the F2/12 surface flux tallies require the surface area. The surface area referred to here is the total area of a surface which may bound several cells and not a surface segment bounding only a particular cell. If you need only the segment of a surface, you need to segment a full surface with the FSn (see page 150) card and then use the VAn card if MCNP cannot automatically calculate the segmented area.

MCNP will attempt to calculate all surface areas unless a non-zero area for a surface is entered on the AREA card. If MCNP cannot calculate the area and it is not entered, it is set to unity.

Like the VOL card, the AREA card is not associated with a particular tally but with the entire geometry.

4. Photon Weight (PWT) Card

The PWT card is used only for Mode 1 problems, see page 29. For each cell a minimum photon weight, $W_m(IA)$ can be specified. Photons born with weight below this value play Russian roulette to see if they survive. In order to turn off photon production in a cell a value of −1.0E6 should be entered. If a negative minimum weight is entered, the minimum weight used is made relative to the weight of the source neutrons by

$$|W_m(IA)|*W_s$$

where W_s is the weight of the source neutron that eventually led to this photon production event.

If the PWT card is omitted all $W_m(IA) = -1$. This default should be adequate for most problems.

5. Exponential Transform (EXTYN and EXTYP) Cards

See page 88 for cautions.

The entries on these cards are the individual p's for each cell used for the exponential transformation in the Y direction as described on page 89. If these cards (EXTYN for neutrons, EXTYP for photons) are omitted, the p's are set to zero which is the case for non-biased transport.

If an EXTYN card is used in a neutron-photon problem but an EXTYP card is not, only the neutrons will be biased with the transformation.

6. Forced Collision (FCN and FCP) Cards

The number of forced collisions desired for each cell may be entered on this card (FCN for neutrons, FCP for photons), one entry for each cell. A value of zero turns off forced collisions for a cell and is used as the default in case an FC(N or P) card is not used.

If an FCN card is used in a neutron-photon problem but an FCP card is not, only neutrons will have forced collisions.

The weight-cutoff game is bypassed (in subroutines COLIDN and COLIDP) in forced-collision cells.

7. Detector Contribution (PDn) Card

The entries on this card are the probabilities, P_I, of contributing to the detectors for neutron and photon transport for a detector tally n.

At each collision in cell I, the detector tallies are made with probability P_I ($0 \le P_I \le 1$). The tally is then increased by the factor $1/P_I$ to obtain unbiased results for all cells except those where $P_I=0$. This enables the user to increase the running speed by setting $P_I<1$ for cells many mean free paths from the detectors. It also enables the selective suppressing of contributions from cells by setting the P_I's to zero.

If this card is omitted, by default all $P_I=1$. This default value can be changed for all detector tallies, however, by entries on the PD0 (zero) card.

Consider also using the DD card, page 152.

8. DXTRAN Contribution (DXCPN and DXCPP) Cards

The two cards are analogous to the above PDn card but are used for contributions to the DXTRAN sphere. The cards are for neutrons and photons, respectively, and each has a default value of 1.

Consider also using the DD card, page 152.

C. Source Specification

The following cards specify information for the neutron or photon source:

Mnemonic (Columns 1-5)	Card Type
SRCn (n=1,2,3,4,5,blank)	Source Type
SBIAS	Source Energy Bias
SPROB	Source Energy Probabilities
SERG	Source Energies

CHAPTER 3
Source Cards

The specification of a source particle consists of the following information:

(1) geometrical location
(2) direction of flight
(3) energy
(4) time
(5) particle weight

To further complicate matters, probability distributions can exist for any of the above variables. The user can pick from five standard sources by using an SRCn card, or because of the infinite number of possible sources, MCNP allows the user to input data to his own subroutine SOURCE by using an SRC card. The energy distribution for the standard sources is specified by using the SBIAS, SPROB, and SERG cards as described below.

A word of warning concerning biasing. One should never bias a source completely. For example, if p=1 on a SRC1 card, all particles would be started within the cone $\nu<v<1$. Such a source is no longer an isotropic point source.

Secondly, when source biasing (either direction or energy) is used, the actual average weight, or weight multiplier W, may deviate statistically a few percent from what is specified on the SRCn card. You may then renormalize the printed results when this occurs. If the discrepancy is significant, this is an indication of a serious problem such as poor biasing.

The source in a problem with one or more reflecting surfaces should be considered carefully, see page 21.

It is a fatal error to start source particles in or into a cell of zero importance.

1. Source Type (SRCn) Card

The type of source used is determined by the n on the SRCn card as follows:

Mnemonic (Columns 1-5)	Source Type
SRC	Subroutine SOURCE is supplied by user

SRC1	Directionally biased point isotropic source
SRC2	Directionally biased outward cosine distribution on a spherical surface
SRC3	Directionally biased inward cosine distribution on a spherical surface
SRC4	Uniform distribution in volume
SRC5	Plane wave

The time distribution for all standard sources assumes that all particles are emitted at time TME=0. The energy distributions are input through the SBIAS, SPROB, and SERG cards as described below. The geometrical locations and biasing descriptions are input on the SRCn cards.

Source Type — Entries and Description

SRC1 x y z I W p ν

This card specifies an isotropic point source located at the point (x,y,z) in problem cell I with an average particle weight W. The direction cosine with the y-axis, v, is sampled uniformly within the cone $\nu<v<1$ with probability p and within $-1<v<\nu$ with the complementary probability (1-p). The weights assigned are $W(1-\nu)/(2p)$ and $W(1+\nu)/[2(1-p)]$, respectively. The sampling of the direction cosines with the x and z axis (u and w) is not biased. The default values set by MCNP are x=y=z=0, I=1, W=1, p=.5, ν=0. The value of p is not allowed to be 0 or 1.

SRC2 J W p ν

This card specifies an outward cosine distribution on the spherical problem surface J with an average particle weight W. There can be more than one cell outside of sphere J but only one cell inside the sphere. The surface J has to be a sphere

defined by one of the five sphere surface cards (i.e., not by SQ or GQ). The point on the sphere is obtained by sampling (u,v,w) exactly as described for the SRC1 card using ν and p. Directional biasing is toward the y-axis. The weight is set as for the SRC1 card and the point (x,y,z) is the intersection of the (u,v,w) vector from the center of the sphere with the surface. The actual (u,v,w) direction is then selected by sampling the cosine of the angle with respect to the outward normal from a cosine distribution [$\cos(\varphi)=\text{sqrt}(\xi)$ with ξ a random number] and determining (u,v,w) by sampling a random azimuthal angle between 0 and 2π on the cone defined by $\cos(\varphi)$. The default values are J=1, W=1, p=.5, and ν=0.

SRC3 J W p ν

This card specifies an inward cosine distribution on the spherical problem surface J with weight multiplier W. Surface J must be a sphere defined by one of the five sphere surface cards. The particles are biased inward from the spherical surface. For a void this source provides a uniform isotropic flux inside the spherical surface. There can be only one cell outside surface J, but there may be more than one inside. The default values are J=1, W=1, p=.5, and ν=0.

SRC4 x y z I W p ν R_1 R_2 D u v w

This source is in a single starting cell and is isotropic in angle and uniform in volume. The source cell is surrounded or enclosed by a spherical or a cylindrical shell which does not have to be a real surface of the problem.

Input items on the SRC4 card:

(1) x
(2) y
(3) z

} Coordinates of the center of the enclosing surface

(4) I = Starting cell problem number (default = 1)

(5) W = Starting weight multiplier of particles (default = 1)

(6) p = Cone bias probability (same as SRC1)

(7) ν = cosine of biasing cone (same as SRC1)

(8) R_1 = Inner radius of covering surface (may be zero)

(9) R_2 = Outer radius of covering surface (normally not zero). If this entry is zero, this becomes a point source identical to the SRC1 source. However, the SRC1 source would be preferrable since it would use fewer random numbers, less coding, and thus be more efficient.

(10) D = Extent (zero for spheres or non-zero for cylinders)

(11) u
(12) v
(13) w

} Direction cosines of a vector along the cylindrical axis (need not be normed)

If the 10^{th} entry D = 0, the covering surface is a sphere. The direction-cosine entries are then not needed. The source points are thrown uniform in volume in the spherical shell from R_1 to R_2, and rejected if they fall outside of cell I. If the 10^{th} entry $\neq$ 0, the covering surface is a cylinder. The three direction-cosine entries are then needed. The value of the 10^{th} entry D is the extent measured from the point (x,y,z) along the cylindrical axis, either positively or negatively. The source points are selected by throwing linearly in $\pm$D along the axis and then uniform in area in the circular cross section of the cylinder at this axial point from R_1 to R_2. The source point is rejected if it lies outside cell I.

The efficiency of the rejection is printed. This source efficiency should be the ratio of the source cell volume to the covering cell volume within statistics. The user should take care that the covering surface completely covers the source

cell. A fatal error results from the SOURCEA subroutine if the source efficiency drops below 1%, or if the first 100 source particles are rejected.

SRC5 x y z I W R u v w I′ J A

This is a plane wave source incident on a circular window; the window radius may be zero. The source is either uniform in area or linear on the window radius.

The input parameters on this card are:

(1) x
(2) y
(3) z
} Coordinates of the center of the circular window

(4) I = Problem number of starting cell (default = 1)

(5) W = Starting weight multiplier of particles (default = 1)

(6) R = Radius of the circular window. If this entry is zero, the source becomes a point source incident at (x,y,z) with starting direction (u,v,w). If R = 0, direct detector contributions are not allowed.

(7) u
(8) v
(9) w
} A vector representing the plane wave direction (need not be normed)

(10) I′ = Cookie-cutter cell (zero or non-zero) see *Options*

(11) J = Starting surface problem number

(12) A = Area of source and used only in conjunction with detectors (default is πR^2)

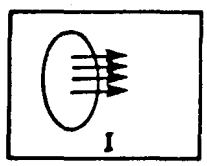
The window is defined on a plane normal to the plane wave direction, centered at (x,y,z) and having radial extent R. This circular window must lie completely within cell I. The source particles are selected uniform in area on the circle and are given direction cosines (u,v,w).

Options:

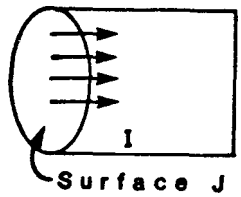
(A) If the 5th entry W is negative, the source particles are chosen linearly in R instead of uniformly in area. This is still effectively a uniform plane-wave source, but more weight-adjusted particles are started toward the center of the source area. The starting weight is multiplied by the absolute value of the 5th entry.
(B) If the 6th entry R is negative, then the source starting directions are chosen in a cosine distribution instead of a plane wave. The directions (u,v,w) are then defined to be those of the outward normal to the window. The radial extent is the absolute value of the 6th entry.
(C) If the 10th entry I′ is zero, the circular extent is R. If it is *non-zero*, this entry represents the problem name of a cell defined on the window, but *within* the circle. This is a way of defining a window (or a cookie cutter) of a certain shape. The source point is thrown uniformly in area on the circle, and rejected if it lies outside this cell. The efficiency of this rejection scheme is printed.

The geometry for this window cell must be specified by the appropriate cell and surface cards. This cell need not be a real cell of the problem, nor need it be connected to any real cell. It is simply a way of defining a shape on the window within the circular extent. This shape is the intersection of the window cell and the window. By definition, all source points selected will lie on the window.
(D) The 10th entry I′ and the 11th entry J can be used in four combinatins for various applications:

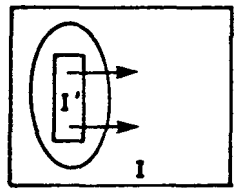
Case 1: I′ = 0 J = 0
The circular window must be in cell I (the 4th entry). No rejection of source points is made, but it is a fatal error if particles start outside I.



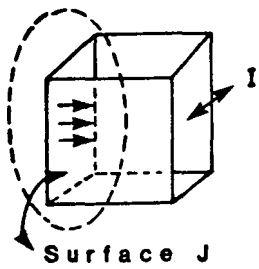
Case 2: I′ = 0 J ≠ 0 (plane)
J is a real *plane* surface of the problem. No rejection of source points is made, but a fatal flag is set if particles starting in the window do not enter cell I. The 4th entry I becomes the cell which will be entered.



Case 3: I′ ≠ 0 J = 0
I′ is a cell specified by appropriate additional cell and surface cards. Source points lie on the intersection of I′ and the circular window and must be in cell I or the fatal flag is set. However, rejection takes place if the points are not in cell I′.



Case 4: I′ ≠ 0 J ≠ 0 (plane)
Cell I′ may be an extra cell or a real cell, and J is a real plane surface of the geometry. Rejection takes place if starting particles are not in cell I′ and fatal if then not in I.



2. Source Energy (SERG) and Energy Probability (SPROB) Cards

The SERG and SPROB cards together give the distribution function of the energy spectrum for the sources. The first entry on the SERG card should be the minimum particle energy from the source, followed by the energy entries in order of increasing magnitude through the maximum allowable energy. The entries on the SPROB card are, in one case, the cumulative probabilities that a source particle has an energy less than or equal to the corresponding entry on the SERG card. The first entry on the

SPROB card must always be 0 and the last entry must be 1 for a cumulative distribution. All entries must be positive.

However, if the source probability distribution is derived from data giving the number of particles started in each energy group (i.e., a probability density function), this data can be entered directly onto the SPROB card. The first entry is again 0, followed by the input for each energy group up through the highest energy group. The code will process these entries to form the corresponding cumulative probability distribution. The code distinguishes between the two modes of entry on the SPROB card by examining the last entry. If this is 1, it assumes that a cumulative probability distribution was read in; otherwise, it processes the data to form the distribution. Therefore, the user should be certain that a 1 is not used for the last entry if the distribution is not cumulative.

The maximum number of entries on each of the SERG and SPROB cards is 51, and the number of entries on each card must be the same.

Two examples of these cards may be instructive. For a monoenergetic source energy of 14 MeV, the two cards would be

```
SERG     14    14
SPROB     0     1
```

For a linear energy distribution from 0 to 14 MeV, the cards become:

```
SERG      0    14
SPROB     0     1
```

3. Source Energy Bias (SBIAS) Card

The entries on the SBIAS card are used to bias the sampling of the energy distribution of the source. These entries are called track fractions. A track fraction is the fraction of particle histories (regardless of the weights attached to these histories) or "tracks" started in a given energy interval. For example, it is possible to start more tracks at high energies in a problem and correct the distribution by lowering the weights assigned to these tracks. By altering the weights, this technique conserves the total weight of particles started in an energy interval.

The first entry on the SBIAS card must be a 0. This is followed by an entry proportional to the number of tracks to be started in the lowest

energy group, the entries continuing in the same way, one for each energy group, through the highest group defined for the source. Like the SPROB card this card can also contain a cumulative probability distribution with the last entry being 1. All entries must be positive and no entry other than the first one may be zero.

The code normalizes the entries on the SBIAS and SPROB cards and then divides the fraction of actual source particles in an energy bin by the fraction of tracks started in that bin to obtain the weight assigned to particles in that bin. These weights are then stored in the SBIAS data block. The cumulative probability distribution for the fictitious source is then calculated and stored in the SPROB data block.

Like the SERG and SPROB cards, the maximum number of entries is 51, and the number must be the same as on the SERG and SPROB cards.

4. Subroutines SOURCE and SRCDX

If an SRC card is used, it signals the MCNP code (by setting NSR=0) that the user wishes to describe his source by supplying a FORTRAN subroutine SOURCE.

Appendix A gives details about how to supply the SOURCE subroutine to MCNP with UPDATE. Chapter 4 has an example of a fairly elaborate SOURCE that illustrates some very useful features of MCNP. The parameters which must be specified within the subroutine are:

Variable	Description
X	x-coordinate of particle's position (in cm)
Y	y-coordinate of particle's position (in cm)
Z	z-coordinate of particle's position (in cm)
U	x-axis direction cosine of particle's direction
V	y-axis direction cosine of particle's direction
W	z-axis direction cosine of particle's direction

IA The program name of the cell containing the source particle, or in the case of a surface source, the cell which will be entered.

JA For a surface source, this must be the program name of the surface upon which the particle is starting. If the particle is not starting on a surface, JA must be set to zero.

TME Particle's starting time in shakes (10^{-8} sec)

WGT Particle's weight (usually 1.0)

ERG Particle's energy in MeV

Prior to calling subroutine SOURCE, the MCNP code calculates an isotropic (U,V,W). Therefore, the user need not specify the direction cosines (U,V,W) if he desires an isotropic distribution.

The SERG, SPROB, and SBIAS cards may also be used with the SRC card to define the particle's energy ERG and weight WGT. To do this, however, a call to the energy sampling subroutine, CALL ERGSMP, must be in the subroutine SOURCE.

A random number generator RANF(1) is available for use by subroutine SOURCE for generating random numbers between 0 and 1. The number of random numbers called must be incremented manually by adding the statement NRN=NRN+1 after every call to RANF.

Up to 51 entries (such as energy, coordinates, or anything else) may be put on the SRC card. They will be stored in the SRC array which is in COMMON and can therefore be used anywhere in MCNP. Most frequently, however, the SRC array is used to input parameters to the user-provided subroutine SOURCE, and the order and meaning of entries on the SRC card depend upon the structure of SOURCE.

If you are using a detector or DXTRAN and your source has an anisotropic angular distribution, you will also need to supply an SRCDX subroutine to specify PSC for each detector or DXTRAN sphere, see Chapers 2 and 4.

D. Tally Specification

The tally cards are used to specify what type of information the user wants to gain from the Monte Carlo calculation, i.e., current across a surface, flux at a point, heating in a region, etc. This information is requested by using the following cards:

Mnemonic (Columns 1-5)	Card Type
Fna	Tally Type
FCn	Tally Comment
En	Tally Energies
Tn	Tally Times
Cn	Cosines - Tallies 1 and 11
FMn	Tally Multiplier
EMn	Energy Multiplier
TMn	Time Multiplier
CMn	Cosine Multiplier
CFn	Cell Flagging
SFn	Surface Flagging
FSn	Tally Segment
VAn	Segmented Volume/Area
FUn	TALLYX Input
DD	Detector and DXTRAN Diagnostics
DXN	DXTRAN for Neutrons
DXP	DXTRAN for Photons

The ***n*** is a tally number no longer than three digits.

Much of the information entered on these cards is used to describe tally "bins." A tally bin is the smallest unit of a tally and is for a specific increment: $(T_1<\text{Time}\leq T_2)$, $(E_1<\text{Energy}\leq E_2)$, etc.

The results of all tallies are normalized to be per source particle. If reflecting planes are used, the user may have to further normalize the tallies himself (can be done by setting the weight of the source particles or by using the FMn or EMn cards).

1. Tally (Fna) Cards

There are 6 basic neutron tallies and 5 basic photon tallies available

in MCNP as standard tallies. All are normalized to be per source particle unless changed by the user with a TALLYX subroutine.

Mnemonic (Columns 1-5)	Tally Description
F1 and F11	Current across a surface
F2 and F12	Flux across a surface
F4 and F14	Flux in a cell
F5a and F15a	Flux at a detector
F6 and F16	Energy deposition in a cell
F7	Fission energy deposited in a cell

The card mnemonic for a tally is Fna where n=1,2,4,5,6, and 7 (or increments of 20) for neutrons and n=11,12,14,15, and 16 (or increments of 20) for photons, with the restriction that n must be less than four digits. Thus you may have as many of any one basic tally as you need, each with different energy bins or flagging or anything else. For example, F4, F24, F104, and F204 are all legitimate cell flux tallies. Entries on this card (except for detectors) are the problem numbers of the cells or surfaces for which the tally is desired.

The tallies have one default bin over all energy, time, and direction. To specify particular bins for each tally or new defaults for all tallies, you will have to provide the En, Tn, and Cn cards (see description below).

The total number of detectors is restricted to 20. The total number of tallies is limited to NTALMX-6 = 40.

Tally types 1, 2, 4, 5, 11, 12, 14, and 15, are normally weight tallies; however, if the F card is flagged with an asterisk (i.e., *F1), energy times weight will be tallied. The units will then be MeV, MeV, MeV/cm^2, MeV/cm^2, MeV, MeV, MeV/cm^2, and MeV/cm^2, respectively. The asterisks flagging can also be used on tallies F6, F7, and F16 to change the units from MeV/gm to jerks/gm.

The quantity a on the tally card has meaning only for the n=5/15 (or increment of 20) detector tallies. In particular

a = blank for analog point detectors,
= W for *wunce*-more collided detectors, and
= X, Y, or Z signifying an analog ring detector about the x, y, or z axis, respectively.

For point detectors, the complete card is

Fna X Y Z $\pm R_o$

where a is blank or W,
+ or blank R_o is the sphere radius in centimeters, and
$-R_o$ is the sphere radius in mean free paths (cannot be used in a void region, see cautions on page 65).

For ring detectors, the complete card is

Fna a_o r $\pm R_o$

where a is X,Y, or Z
a_o is where the ring plane intersects an axis,
r is the radius of the ring in centimeters, and
$\pm R_o$ has the same meaning as for point detectors but is a sphere about the point selected on the ring.

Ring (rather than point) detectors should be used in all problems with axial symmetry.

You are encouraged to read about detectors starting on page 55 before using them. Remember that contributions to a detector are not made through a region of zero importance.

For more than one detector with the same *na* designation, sets of the above input parameters are simply continued on the same Fna card.

Detectors cannot be used in problems with the $S(\alpha,\beta)$ thermal treatment and should be used with caution in problems with reflecting surfaces (see page 63).

Rules of Thumb for R_o: For the analog point and ring detectors, R_o should be about 1/8 to 1/2 mean free paths for particles of average energy at the sphere and zero in a void. For the OMCFE detector, R_o should be about 1.5 to 3 mean free paths for the source energy. Supplying R_o in terms of mean free path will increase the variance and is not recommended unless you have no idea how to specify it in terms of centimeters.

2. Tally Comment (FCn) Card

Anything entered in columns 6–72 on this card will appear as the title heading of tally Fn. This card is particularly useful when tallies are modified in some way so that later readers of the output will be warned of

nonstandard tallies. The comment on this card may be continued for as many lines as you like simply by following the FCn card by cards with the continued comment in columns 6–72 (i.e., columns 1–5 are blank).

3. Tally Energy (En) Card

The entries in MeV on this card are the upper bounds of the energy bins for tally n. The entries must be entered in the order of increasing magnitude. If a particle has an energy greater than the last entry, it will not be tallied. If the last entry is greater than the upper energy limit E_{max} specified on the ERGN card, the last bin will be lowered to E_{max}.

The default is to tally over all energies, but this default can be changed for all tallies with entries on the E0 (zero) card.

4. Tally Time (Tn) Card

The entries in shakes on this card are the upper bounds of the time bins and like the En card must be entered in order of increasing magnitude. If a particle has a time greater than the last entry on the Tn card, it will not be tallied. The last time bin entry should always be less than or equal to the time cutoff (see page 162).

If the time bins are entered greater than the time cutoff, the first bin limit over the cutoff will be lowered to the cutoff. All other bins will remain the same.

The default is to tally over all time, but this default can be changed for all tallies with entries on the T0 (zero) card.

5. Cosine (Cn) Card (Tallies 1 and 11 only)

The entries on this card are the cosine limits of the angular bins used for tallies 1 and 11. The angular limits are defined with respect to the normal to the surface at the particle point of entry. The normal to the surface is always in the direction of a cell that has positive sense with respect to that surface.

The card entries are the upper bounds of the cosine bins where the order of entry starts with the first angle less than 180° to the normal and continues to the normal (cos=1). Thus, to tally currents within the angular limits 180° to 150°, 150° to 120°, 120° to 90°, 90° to 60°, 60° to 30°, and 30° to 0° with respect to the normal, the entries on the Cn card

would be -.866, -.5, 0., .5, .866, 1.0. The last entry must always be 1. A lower bound of -1 is set in the code and is not entered on the card.

As an example of the relation between a surface normal and sense for the C1 card, consider a source at the origin of a coordinate system and a plane intersecting the +y axis. An entry of 0 and 1 on the C1 card will tally all source particles transmitted through the plane in the 0 to 1 angular bin (0° to 90°) and all particles reflected back across the plane in the -1 to 0 angular bin (90° to 180°). A plane intersecting the -y axis will result in a tally of all source particles transmitted through the second plane in the -1 to 0 bin (90° to 180°) and all particles reflected back across the plane in the 0 to 1 bin (0° to 90°).

The default is to tally over all angles, but this default can be changed for all tallies with entries on the C0 (zero) card.

6. Tally Multiplier (FMn) Card

This card is used to calculate any quantity of the form

$$C \int \varphi(E)\ f(E)\ dE$$

where $\varphi(E)$ is the energy-dependent flux and $f(E) = \prod R_i$, the product of quantities in the cross-section libraries or specially designated quantities. The entries on the FMn card are

FMn C m R_1 R_2 ... R_i

where

- C is any constant,
- m is the material number defined by m on an Mm material card,
- R_1 is the first ENDF reaction reaction number,
- R_2 is the second ENDF reaction number, etc.

There is no restriction on the number of ENDF or special reactions which can be used. If none is given, then the tally is simply multiplied by the constant C unless m=0. In the latter case f(E) effectively becomes $1/\varphi(E)$ which then tallies the number of tracks.

In addition to the approximately one hundred standard ENDF reaction

numbers (see Appendix F) available (R = 1,2,16,102,etc. which is σ_{tot},σ_{el},$\sigma_{n,2n}$,$\sigma_{n,\gamma}$,etc.), the following nonstandard, special R numbers may be used:

Neutrons:	-1	total cross section with thermal effects
	-2	absorption cross section without thermal effects
	-3	absorption cross section with thermal effects
	-4	average heating number
	-5	gamma-ray production cross section
	-6	total fission cross section
	-7	fission ν
	-8	fission Q (MeV/fission)
Photons:	-1	incoherent scattering cross section
	-2	coherent scattering cross section
	-3	fluorescent cross section
	-4	pair production cross section
	-5	total cross section
	-6	photon heating number
Both:	m=0	tally score set to 1 to tally number of tracks rather than flux or number of particles (reaction number ignored).

A list (taken from the ENDF Manual) of many of the ENDF reaction numbers can be found in Appendix F. Note that for tritium production the R number differs from one nuclide to another.

As an example, the F5 tally computes the quantity

$$\int \varphi(E)\, dE$$

If the following FM5 card is used,

Fm5 C m R_1 R_2 R_3

this tally becomes

$$C \int \varphi(E)\ \sigma_{R_1}{}^{m}(E)\ \sigma_{R_2}{}^{m}(E)\ \sigma_{R_3}{}^{m}(E)\ dE$$

where C is a constant and $\sigma_{R_i}^{m}(E)$ is the R_i reaction cross section for material m specified on an Mm card. Material m does not have to be the actual material in the cell containing the detector. The actual material may be iron or a void for example, but the FM5 card may determine photon heating for carbon as though it were in the cell.

Several more examples are in Chapter 4.

7. Energy Multiplier (EMn) Card

This card can be used with any tally (specified by n) to calculate a response function rather than the usual current, flux, etc. There should be one entry for each energy entry on the corresponding En card. When a tally is being recorded within a certain energy bin, the regular contribution is multiplied by the entry on the EMn card corresponding to that bin. For example, a dose rate can be tallied with the appropriate response function entries. Tallies can also be changed to be per unit energy if the entries are $1/\Delta E$ for each bin.

8. Time Multiplier (TMn) Card

The i^{th} entry is the multiplier for the i^{th} time bin of tally n. This card is just like the EMn card except that the TMn card multiplies time bins rather than energy bins. The Tn and TMn cards must have the same number of entries. As an example, if the entries are $1/\Delta T$ where ΔT is the width of the corresponding time bin, the tally will be changed into per unit time with the units of $1/\Delta T$.

9. Cosine Multiplier (CMn) Card

The i^{th} entry is the multiplier for the i^{th} cosine bin specified on the Cn card. The number of entries on the CMn card must be the same as on the Cn card. For example, if you want the directionally-dependent F1 tally results to be per steradian, the i^{th} entry on the CM1 card is

$$\frac{1}{2\pi(\cos\ \theta_i - \cos\ \theta_{i-1})}$$

where θ_o is 180°.

10. Cell Flagging (CFn) Card

Cell flagging cannot be used for detector tallies. The same purpose can be accomplished with a TALLYX subroutine.

The entries on the CFn card are problem cell numbers whose contributions are to be "flagged" for the n^{th} tally.

Example:

F4	6	10	13
CF4	3	4	

The flag is turned on when a neutron enters cell 3 or 4. The print of Tally 4 is doubled. The first print is the total track length tally in cells 6, 10, and 13. The second print is the tally in these cells for only those neutrons that have passed through cell 3 or 4 at some time before making their contribution to the cell 6, 10, or 13 tally.

The cell flag is turned on only upon leaving a cell and not entering. A source particle born in a flagged cell does not turn the flag on.

In Mode 1 the flagged neutron tallies are those caused by neutrons passing through the flagged cell, but the flagged photon tallies can be caused by either a photon passing through a flagged cell or a neutron passing through a flagged cell and then leading to a photon which is tallied.

11. Surface Flagging (SFn) Card

Surface flagging cannot be used for detector tallies. The same purpose can be accomplished with a TALLYX subroutine.

The entries on this card are like those on the CFn card except that they are problem surface numbers. Thus a second tally print is given for only those particles that have crossed some surface specified on the SFn card.

Both a CFn and an SFn card can be used for the same tally. The tally is flagged if the track passes through one or more of the specified cells and surfaces.

12. Tally Segment (FSn) Card

This card allows you to subdivide a cell or a surface into segments for tallying purposes. The advantage of this approach is that it is then not necessary to specify the problem geometry with extra cells just for tallying.

The entries on the FS card are the signed surface problem numbers which define how to segment any surface or cell tally. The sign is the sense of the segmented region with respect to that particular surface.

The segmenting surfaces specified on the FSn card are listed with the regular problem surfaces, and they should not be part of the actual geometry and hence do not complicate the cell-surface relationships.

For every K entries on the FSn card, K+1 surface or volume segments are created. Tally n is subdivided into K+1 tallies according to the order and sense of the segmenting surfaces listed on the FSn card. If n is 1/11 or 2/12, the segmenting surfaces divide a problem surface into segments for the F1/11 or F2/12 tallies. If n is 4/14, 6/16, or 7, the segmenting surfaces divide a cell into segments for these tallies.

As an example, consider the flux across surface *A* calculated with the F2 tally (surface *A* is an entry on the F2 card). It is desired to subdivide surface *A* into three sections and get the flux across each section. Two segmenting surfaces *B* and *C* which intersect surface *A* are added to the INP surface list and then added to the FS2 card. The order and sense of the surfaces on the FS2 card is important. If they are listed as *-B -C* there are three prints for the F2 tally: (1) the first is all the flux crossing surface *A* but with negative sense with respect to surface *B*, (2) the second is all the flux crossing surface *A* with respect to the second surface listed on the FS2 card which is then with negative sense to surface *C* - but which has not been scored already (this then precludes the flux with a negative sense to surface *B*), and (3) everything else that has not already been scored. Note that listing the same surfaces on the FSn card as *-B C* changes the order of the printed flux. The single entry *-C* produces two segments: (1) the first is the sum of the flux across the two segments in the first case above, and (2) the second is everything else.

Several addition examples of the FSn card are in Chapter 4.

13. Segmented Volume/Area (VAn) Card

For segmented cell volumes or surface areas defined by the FSn card that are not automatically calculated by MCNP, the user can provide these volumes or areas to be used by tally *n*. This card is analogous to the VOL and AREA cards but is used for specific tallies where the other two are used for the entire problem geometry.

The entries correspond to the subdivided tallies as specified on the FSn card. One more entry is required on the VAn card than is on the FSn card to account for the last catch-all tally bin implicit with the FSn card.

14. TALLYX Input (FUn) Card

This card is used in conjunction with a user-supplied tally modification subroutine, TALLYX. The N entries on the FUn card serve two purposes - (1) each entry establishes a separate tally bin based on the mechanics of the standard tally bin *n*, and (2) each entry can be used as an input parameter for the corresponding tally (or bin) it establishes.

You may wish to think of the FUn card dividing the standard tally *n* in several sub-tallies or adding a set of user bins to a standard tally just like energy or time.

The entries on the FUn card are stored in the array TDS(L+I), I=1,N where L=IPTAL(19,ITAL). There are N=ITDS(L) entries on the FUn card.

15. Subroutine TALLYX

This user-supplied tally-modification subroutine is called whenever a tally with an FUn card is scored. The sequence in which it is called for the various standard tallies is outlined in Chapter 2 in Section V. TALLYX can be called by more than one FUn card; a branch must be constructed inside the subroutine based on which tally Fn is calling TALLYX where n = IPTAL(20,ITAL).

TALLYX has the following form:

```
SUBROUTINE TALLYX(T,IB)
User-supplied FORTRAN statements
RETURN
END
```

The quantity T that is scored in a standard tally may be multiplied or replaced by anything. The modified score T is then put into one of the N user tallies (or bins) established on the FUn card.

Which of the N user tally bins a score goes into is determined by the integer IB. If a particular user tally bin IB is not specified in TALLYX, the score goes into the first bin established by the FUn card. If IB < 1 or IB > N, then no score is made.

Each user tally (or bin) established by an FUn card may be further subdivided into energy, time, or cosine bins by supplying the En, Tn, or Cn cards in the input file.

Several examples of the FUn card and TALLYX are in Chapter 4. The procedure for implementing a TALLYX subroutine is the same as for the user-provided SOURCE subroutine which is outlined in Appendix A on page 282.

16. Detector Diagnostics (DD) Card

This card serves two purposes: (1) it provides more information about the origin of large contributions or the lack of a sufficient number of collisions close to the detector, and (2) it can significantly speed up detector calculations by limiting the distance from a detector for which contributions from collisions are accepted. The information provided may be useful for setting cell importances or source biasing parameters.

The format of the DD card is

DD λ_{max} m

The first entry λ_{max} is the number of mean free paths beyond which Russian roulette is played for contributions to the detector. Beyond λ_{max}, 90% of the contributions are killed but 10% survive with an increase in weight by a factor of ten. λ_{max} = 0 means to accept all contributions to the detectors; its value does not have to be an integral number of mean free paths and can even be less than 1. This parameter is generally set by making a preliminary short run with it set to 0. The output will then show you the detector contributions as a function of mean free path which should give you insight into setting λ_{max} (see page 259).

The second entry *m* determines the diagnostic printout. If it is absent or zero, there is none. A non-zero *m* means to print the first one hundred heavy scores for each detector and to print a summary of where the scores came from along with the regular tally print. A heavy score is one

whose value is at least *m* times as large as the average of the previous scores (at least 200 scores must have been made before heavy scores are printed).

The DD card applies to all detectors in the problem and can also be used for contributions to the DXTRAN spheres. An example of the DD card and a description of its output is in Chapter 4.

17. DXTRAN (DXN and DXP) Cards

DXTRAN is used to improve the particle sample in the vicinity of a tally, see page 95. However, it should not be misconstrued as a tally itself such as a detector; it is used in conjunction with tallies. As such, the DXN and DXP cards are not true tally specification cards.

The format of the input cards is

DXN	X	Y	Z	R_I	R_O	DWC1	DWC2	(neutrons)
DXP	X	Y	Z	R_I	R_O	DWC1	DWC2	(photons)

where

(X,Y,Z) is the point at the center of the spheres
R_I is the radius of the inner sphere in cm
R_O is the radius of the outer sphere in cm
DWC1 is the upper weight cutoff in the spheres
DWC2 is the lower weight cutoff in the spheres
(The weight cutoffs are defaulted to zero.)

There can be up to five sets of X Y Z R_I R_O. There is only one set of the two weight cutoffs which is the last entry after conclusion of the other data.

Consideration should be given to using the DXCPN, DXCPP, or DD cards when using DXTRAN.

DXTRAN cannot be used with the S(α,β) thermal energy treatment and should be used with the same caution as for a detector in a problem with a reflecting surface (see page 63).

Rule of Thumb for R_I and R_O: The inner radius R_I should be at least as large as the tally region, and R_O–R_I should be about one mean free path for particles of average energy at the spheres.

E. Materials Specification

The cards in this section specify the isotopic composition of the materials to be used in the cells and also which cross-section evaluations are to be used.

Cards appropriate to this section are

Mm	Material
DRXS	Discrete reaction
TOTNU	Total fission $\bar{\nu}$
AWTAB	Atomic weight
XS	Cross-section files

1. Material (Mm) Card

The following card is used to specify materials:

Mm ZAID$_1$ *fraction*$_1$ ZAID$_2$ *fraction*$_2$ ....

The m on a material card is the cell material number (see page 119) to which this material description corresponds.

The entries on the material card should consist of the identification number (ZAID) of a constituent element or nuclide followed by the atomic fraction (or weight fraction if entered as a negative number) of that element or nuclide, the number of a second constituent element or nuclide followed by its atomic fraction, etc., running through all the elements and nuclides needed to define the material. These two entries are discussed in further detail below.

Nuclide Identification Number, ZAID. This number is used to identify an element or nuclide to any degree desired by the user. The form of the number is

ZZZAAA.nn

where ZZZ is the atomic number of the element or nuclide,
AAA is the mass number of the nuclide, and
nn is the neutron cross-section set identifier.

For naturally occurring elements, AAA = 000. Thus, ZAID=74182.01 represents the isotope $^{182}_{74}W$, and ZAID=74000.01 represents the element tungsten. Natural elements not available from among those listed in Appendix F must be constructed on an Mm card by adding together the individual isotopes (or nuclides) if they are available.

If the density for cells with AAA=000 is input in gm/cm^3, then MCNP will assume the atomic weight for the natural element. If the isotopic distribution for the element differs from the natural element, then the atom density should be entered on the cell cards to insure the correct atom density for these cells.

The ZZZ and AAA quantities are determined for neutrons by looking at the list of cross-sections in Appendix F and finding the appropriate ZAID associated with an evaluation that you want. If you do not find an evaluation for a nuclide that you need, contact X-6.

There are currently seven different cross-section files available and maintained for use by MCNP:

(1) Neutron Recommended (RMCCS),
(2) Neutron Alternate (AMCCS),
(3) Neutron Extraneous (XMCCS),
(4) Neutron Ubangi (UMCCS),
(5) Neutron Discrete Reaction (DRMCCS),
(6) Neutron S(α,β) (TMCCS), and
(7) Photon (MCPLIB).

The contents of these files are changed from time to time and are listed in Appendix F. If nn is not specified the RMCCS file will be searched for any nuclide with the same ZZZAAA. If it is not found on this file the AMCCS file is searched and so forth until all appropriate files are searched. The order of the search can be changed by the XS card or XS execution line message. If the user wishes to use a specific nn he must look at the contents of these files (Appendix F) and decide which cross-section set he wishes to use. Users should be aware that when using the ZZZAAA. specification in runs which use different versions of the cross-section files, their cross-section sets may vary. It is more consistent to always use the ZZZAAA.nn specification.

The amount of space available for cross-section storage is about $300{,}000_{10}$ words. This should be considered your upper limit for planning purposes; however, depending upon how many tallies you have, whether or not you are running a KCODE problem, and a few other things, you may get a few

thousand fewer words. The storage required for each isotope is listed in Table F.3 in Appendix F.

Nuclide Fraction. The nuclide fractions may be normalized to 1 or left unnormalized. For instance if the material is H_2O the fractions can be entered as (.667 and .333) or as (2 and 1) for (H and O) respectively. If the fractions are entered with negative signs they are assumed to be weight fractions; otherwise, atomic fractions. Weight fractions and atom fractions cannot be mixed on the same Mm card.

Photons. If a photon-only problem is being run the AAA can be set to 000 and the nn omitted. Photon cross sections are specified exactly like the neutron cross sections, but for photons ZZZAAA.nn is equivalent to ZZZ000. There is no distinction between isotope and element for photons.

Photon cross sections are available in MCPLIB for Z = 1 to 94 with the exceptions of 84, 85, 87, 88, 89, and 93.

The total number of "nuclide-fraction" entries allowed is MEMAX=120. The total number of different nuclides allowed is MAXE=40.

2. Discrete Reaction Cross Section (DRXS) Card

Any nuclide listed on the optional DRXS card is given a discrete treatment (pseudo-multigroup) using the DRMCCS cross-section file instead of the regular fully continuous-energy cross-section treatment if the necessary discrete data are available (check the list in Appendix F). Nuclides are listed on the DRXS card by ZAID number. Any number of entries is allowed, but if there are no entries on the DRXS card, discrete cross sections will be used for every nuclide if available.

Unless you are transporting neutrons in the resonance region where self-shielding may be of importance, it is recommended to use this card since computer storage requirements will be reduced and timesharing will thereby be enhanced.

3. Total Fission $\bar{\nu}$ (TOTNU) Card

Unless a TOTNU card is used, prompt $\bar{\nu}$ is used for all fissionable nuclides. If a TOTNU card is present then total $\bar{\nu}$ will be used for those fissionable nuclides for which total $\bar{\nu}$ values are available. The symbol Ψ in the nuclide list of Appendix F indicates which fissionable nuclides

have only prompt $\bar{\nu}$ data available. There are no entries on the TOTNU card. The MCNP neutron cross-section summary print from XACT will indicate whether prompt or total $\bar{\nu}$ was used.

All steady-state problems should use this card.

4. Atomic Weight (AWTAB) Card

An AWTAB card on which up to twenty-five combinations of ZAID and associated atomic weight are listed is available. Entries on this card override any existing entries in the MCNP atomic weight table or add new ones to it.

This card is useful in conjunction with the following XS card when you are using cross sections for an isotope for which there is no entry in the MCNP atomic weight table. The ZAID required is precisely the one on the user-supplied cross-section set.

5. Cross-Section File (XS) Card

The names of standard cross-section libraries can be changed on the MCNP execution line: AMCCS=MYMCCS. Up to ten more user-supplied libraries in the proper format can be specified in two ways.

(1) execution line: XS = *filename1, filename2, filename3, ...*

(2) input card : XS *filename1 filename2 filename3 ...*

The file names are limited to eight characters. Both ways can be used in the same problem. A fatal error results if any user-specified library cannot be opened.

The order of search is (1) execution message, (2) INP file, and (3) standard libraries. Duplicate names are detected and later occurrences are ignored. Thus the user could, for example, force UMCCS to be searched before RMCCS by XS=UMCCS.

If you are adding isotopes for which there are no atomic weights in MCNP, you can add them with the above AWTAB card.

F. Energy And Thermal Treatment Specification

The following cards are involved with energy aspects of MCNP:

Mnemonic (Columns 1-5)	Card Type
ERGN	Energy Card, Neutron
ERGP	Energy Card, Photon
NSPLT	Neutron Energy Splitting Card
PSPLT	Photon Energy Splitting Card
TEMPn	Thermal Temperature Card
THTME	Thermal Times Card
TI	Thermal Isotopes Card
MTm	S(α,β) Material Card

All energy entries on these cards are in units of MeV.

1. Neutron Energy (ERGN) Card

The neutron energy card has three entries in units of MEV:

E_{th} E_{max} EMCNF

All neutrons having an energy less than E_{th} are given the thermal treatment by using the free-gas model (see page 31) for designated thermal isotopes (see page 161) and by making all elastic collisions with other isotopes isotropic (in the lab system) with no energy loss.

This thermal cut-in is usually chosen to be a factor of 10 greater than the energy a neutron would have if it were at the temperature entered on the TEMPn cards. For example, if the temperature of a cell is given as $kT=2.5\times10^{-8}$ MeV then a neutron at that temperature would have an energy of $3/2\ kT=3.75\times10^{-8}$ MeV, and an E_{th} of 3.75×10^{-7} should be used. If E_{th} is negative, the thermal cut-in used is given by

$$|E_{th}|*Temp$$

where Temp is the thermal temperature entered on the TEMPn card for the appropriate time and cell. It is recommended that an entry of -15 be used for most problems where the thermal treatment is desired.

If the thermal treatment is not desired, E_{th} should be set to zero which is the default value. If $E_{th}\neq0$ then the user must supply THTME and TEMPn cards.

All neutron cross sections pertaining to energies greater than E_{max} are eliminated by EXPUNG. This entry should be at least as large as the

energy of any neutron in the problem. If a neutron is transported at an energy greater than E_{max}, the cross sections at E_{max} will be used. The purpose of this entry is to trim off unnecessary cross sections to save storage. The default (and maximum) value used for E_{max} is 100 MeV. Neutron cross sections are typically available up to 20 MeV (see Appendix F). For energies greater than the upper limit of the cross-section evaluation, the cross section at that upper limit is used.

The third parameter EMCNF controls the type of capture. Any neutron with energy greater than EMCNF will have the implicit capture treatment; below EMCNF, analog capture. The default value is 0 MeV. This parameter is analogous to EMCPF on the photon card ERGP.

2. Photon Energy (ERGP) Card

This card has a single entry, EMCPF. All photons having an energy greater than EMCPF will be given the simple physics treatment of MCG. All photons with an energy less than EMCPF will be treated with the more detailed physics of MCP. If this card is omitted, the detailed treatment will be used at all energies below 100 MeV.

Photon cross-section data are available up to 100 MeV and down to 0.001 MeV. Below this lower limit, MCNP allows only analog capture regardless of EMCPF which results in a quick termination of photon histories.

3. Energy Splitting (NSPLT and PSPLT) Cards

The entries on these cards consist of pairs of energy splitting parameters, N_{spl} and E_{spl}, with a maximum of five pairs allowed. N_{spl} is the number of tracks into which a particle will be split, and E_{spl} is the energy at which particles are to be split. N_{spl} can only be an integer and cannot be less than 1 (i.e., Russian roulette is not currently allowed). As as example,

NSPLT 2 .1 2 .01

specifies a 2 for 1 split when the neutron energy falls below .1 MeV and another 2 for 1 split when the energy falls below .01 MeV.

NSPLT is for neutrons, and PSPLT is for photons.

4. Free-Gas Thermal Temperatures (TEMPn) Cards

These cards are necessary for neutron transport only if the user wishes to use the free-gas thermal treatment as described on page 31 (also see the ERGN card) and the problem includes thermal isotopes. The thermal isotopes are hydrogen, deuterium, and isotopes specified on a thermal isotope card, TI. Thermal temperatures are entered as a function of time with a maximum of 7 time entries allowed. These times (t_1,t_2,...t_n, n<8) are entered on a thermal time (THTME) card. The thermal temperatures at time t_1 are listed, cell by cell, on the TEMP1 card; the corresponding cell thermal temperatures at time t_2 are listed on the TEMP2 card, etc. A linear interpolation is used to determine the cell thermal temperatures at times between two entries. Time values occurring before t_1, or after t_n, use the thermal temperatures at the nearest time entry. Because thermal temperature entries are required only for those cells whose material composition includes one or more thermal isotopes, all other cell entries can be set to zero.

A fatal error occurs if a temperature is not specified for a cell with thermal isotopes.

If the user does not wish to use the thermal treatment for special isotopes (see TI card) in a given cell the TEMPn entry for that cell should be entered negative. The absolute value of this entry will then be used for hydrogen and deuterium only.

We use kT to denote the thermal temperature of a cell and use units of MeV. The following formulas can be used to provide the values of kT for temperatures in degrees kelvin, Celsius, Rankine, and Fahrenheit.

$$kT(MeV) = 8.617\times10^{-11}T \text{ where T is in degrees k}$$
$$= 8.617\times10^{-11}(T+273.15) \text{ where T is in degrees C}$$
$$= 4.787\times10^{-11}T \text{ where T is in degrees R}$$
$$= 4.787\times10^{-11}(T+459.67) \text{ where T is in degrees F}$$

5. Free-Gas Thermal Times (THTME) Card

The entries on this card are the times in shakes (10^{-8} sec) at which thermal temperatures are specified on the TEMPn cards. A maximum of 7 time entries is allowed in order of increasing magnitude. For each entry on this card, a TEMPn card is required.

6. Free-Gas Thermal Isotopes (TI) Card

The entries on this card are the nuclide ZAID numbers (see Mm card description) of the nuclides up to and including Z=8 for which a thermal free gas treatment is desired. Numbers to the right of the decimal point are ignored so that either ZZZAAA.nn or ZZZAAA. can be entered. Note that hydrogen and deuterium will always be treated with the free gas treatment regardless of whether a TI card is used or omitted. The user should only input important isotopes since the thermal treatment is time consuming.

7. S(α,β) MTm Card

For any material defined on an Mn card, a particular component of that material (represented by a ZAID number) may be associated through an MTm card with an S(α,β) data set if that data set exists. The S(α,β) data for that ZAID is used in every cell in which that material is specified. For a particular ZAID in a material, the free-gas treatment may be used down to the energy where S(α,β) data are available. At that point, the S(α,β) treatment automatically overrides the free-gas treatment (i.e., there is no mixing of the two treatments for the same ZAID in the same material at a given energy).

Typically the free-gas model will be used for a particular ZAID of a material from 10 eV to 4 eV and then the S(α,β) treatment will take over. In general, S(α,β) effects are most significant below 2 eV.

There is no connection between the TI and MTm cards; one may be used without the other.

S(α,β) data currently available on the TMCCS data file include light and heavy water, graphite, polyethylene, and beryllium metal. If you have other needs like uranium dioxide, beryllium oxide, or zirconium hydride, contact X-6.

The input card that invokes the S(α,β) treatment is the MTm card plus a defined mnemonic on that card. The *m* refers to the material *m* defined on a regular Mm card. The appearance of an MTm card will cause the loading of the corresponding S(α,β) data from the thermal data file TMCCS.

The list of current S(α,β) mnemonics for the MTm card is

LWTR	^{1}H in light water	($^{1}H_2O$)
HWTR	^{2}H in heavy water	($^{2}H_2O$)
POLY	^{1}H in polyethylene	(CH_2)
GRPH	^{12}C in graphite	

BE	^{9}Be in solid beryllium

Examples of use:

```
M1      1001 2   8016 1          light water
MT1     LWTR

M14     1001 2   6012 1          polyethylene
MT14    POLY

M8      6012 1                   graphite
MT8     GRPH
```

There is currently no theory for $S(\alpha,\beta)$ contributions to detectors.

C. Problem Cutoffs

Particle tracks are terminated from the Monte Carlo transport process by the energy, time, weight cutoffs, Russian roulette, and by DXTRAN in addition to entering a cell of zero importance. The following cards can be used in an initiate-run or a continue-run input file in order to specify some of these cutoffs. You may also terminate a job by the total number of histories run or by the computer time used.

The cards in this section are:

CUTN	Neutron cutoffs
CUTP	Photon cutoffs
NPS	History cutoff
CTME	Computer time cutoff

1. Neutron Cutoffs (CUTN) Card

Four entries exist on this card:

T_{con} E_{con} WC1 WC2

The time cutoff, T_{con}, is in units of shakes (10^{-8} seconds). The transport of a neutron is immediately stopped and the neutron killed if its time

becomes greater than T_{con}. The default value of T_{con} is 1.0E123 shakes (1.0E115 seconds).

The energy cutoff, E_{con}, is in units of MeV. Any neutron having an energy lower than E_{con} is killed. The default value for E_{con} is 0.

If a neutron's weight WGT falls below WC2 times the ratio R of the source cell importance to the collision cell importance, then with probability WGT/(WC1*R), the neutron survives and is assigned WGT=WC1*R. If negative values are entered for the weight cutoffs, the values

$$|WC1|*W_s \text{ and } |WC2|*W_s$$

will be used for WC1 and WC2, respectively, where W_s is the minimum weight assigned to a source neutron from an MCNP standard source with the exception of SRC3 where the minimum weight is the weight multiplier times the minimum weight resulting from direction biasing times 10^{-10}. These negative entries are recommended for most problems. The default values for WC1 and WC2 are -0.50 and -0.25, respectively. See page 84 for a discussion of weight cutoffs.

If WC1 is set to zero, capture is then treated explicitly in the analog fashion versus implicitly by reducing the neutron's weight according to the capture probability.

2. Photon Cutoffs (CUTP) Card

As with the CUTN card there are four entries on this card:

$$T_{cop} \quad E_{cop} \quad WC1 \quad WC2$$

The time and energy cutoffs, T_{cop} and E_{cop}, are identical to those for neutron transport, T_{con} and E_{con}, except for their defaults. The default for E_{cop} is .001 MeV. The default for T_{cop} is T_{con}. If T_{con} is not specified, the default is 1.0E123 shakes.

The weight cutoffs are used the same as for neutrons except that (1) they are only used for energies above the simple physics (MCG) cutin EMCPF (see ERGP card description, page 159) and (2) there is no provision for explicit analog capture in the simple physics treatment. For energies below EMCPF, the cutoffs have no effect since the detailed physics (MCP) considers only analog capture.

In a Mode 1 problem, the photon weight cutoffs are the same as the neutron weight cutoffs (either the default values or from the CUTN card)

unless overridden on the CUTP card. Again, the photon weight cutoffs have no effect at energies below EMCPF.

MCNP allows only analog capture below 0.001 MeV. Since the photoelectric cross section is virtually 100% of the total cross section below that energy, tracks will be quickly captured and terminated.

3. History Cutoff (NPS) Card

The single entry N on this card is used to terminate the Monte Carlo calculation after N histories have been transported - unless the calculation is terminated earlier for some other reason such as computer time cutoff.

In the case of a continue-run, NPS is the total number of particles including runs prior to the continue-run. However, a negative NPS entry means to print an output file at the time of the last history run.

4. Computer Time Cutoff (CTME)

The single entry on this card is the maximum amount of computer time (in minutes) to be spent in the Monte Carlo calculation. For a continue-run this time limit is the time relative to the start of the continue-run.

If both an NPS and a CTME card are used, whichever of the two cutoffs is hit first predominates.

H. User Data Arrays

Two arrays, IDUMMY and RDUMMY, are in MCNP variable COMMON and are available for the user. They are included in the dumps on the RUNTPE file and may therefore be used for any purposes including accumulating information over the entire course of a problem through several continue-runs. Each array is dimensioned 50, and they may be filled by cards in the input file INP. IDUMMY is an integer array and RDUMMY is a real array.

1. Integer Array (IDUM) Card

The entries (up to 50) on this card fill the IDUMMY array with integer

numbers. If real numbers are entered on this card, they will be truncated and converted to integers.

2. Real Array (RDUM) Card

The entries (up to 50) on this card fill the RDUMMY array with real numbers.

I. Peripheral Cards

The following cards have nothing to do with the transport of particles but offer a variety of conveniences:

Mnemonic (Columns 1-5)	Card Type
PRDMP	Print and Dump Cycle
LOST	Lost Particle Cutoff
DBCN	Debug Information
FILES	Create user files
MCT	Write RUNTPE to tape
PRINT	Printing control

1. Print and Dump Cycle (PRDMP) Card

The two entries on this card

NDP NDM

are the cycle limits for printing out tallies and dumping information on a run file (i.e., RUNTPE) for a continue-run in addition to printing and dumping at the end of the calculation. After every NDP particles the summary and tallies are printed to the output file. After every NDM particles the necessary information is dumped to the run file. A negative entry on this card changes the cycle limit from particles to minutes of computer time. The default for NDP is to print only after the calculation has successfully ended. The default for NDM is to dump every fifteen minutes plus at the end of the problem.

When used with the KCODE, positive entries on the PRDMP card are

interpreted as number of KCODE cycles rather than number of particles started. Printing and dumping are done only at the ends of cycles.

It is recommended to print fairly frequently (every few thousand histories or so, depending upon the total number of histories expected to be run) in order to watch the variance of the variance. This will give you an indication of how reliable your results are - this is especially important when using detectors. For example, a result after 10000 histories may have an *indicated* error of less than 10%, but the variance after 20000 histories may be up to 20% - indicating the earlier result with the 10% variance is not reliable.

2. Lost Particle (LOST) Card

There are two entries on this card:

LOST(1) LOST(2)

The first sets the number of histories that may be lost before the job terminates. The second entry determines the number of debug prints obtained as a result of lost particles. The default for both is 10.

This card should be used cautiously: the user should know why the particles are being lost, and the number lost should be statistically insignificant out of the total sample.

Lost here means just that; a particle gets to an ill-defined section of the geometry and does not know where to go next.

3. Debug Information (DBCN) Card

The nine entries on this card are used primarily for debugging problems and the code itself. All entries are defaulted to 0 except the fifth which is 600.

(1) KRNT is the first entry and is the random number used for starting the transport of the first particle history in a given run. For example if a user wants to run the same job with a different sequence of random numbers or has had difficulties with particle 124 and wishes to restart a calculation with particle 124 as the first particle, he should enter the random number that started particle 124 as KRNT. KRNT may be entered as a decimal number, although the sixteen digit octal number followed by the letter B is the most common form. The last digit of KRNT should be odd to

satisfy requirements of the random number generator. The beginning random number of a particular errant history is labeled SOURCE RN on a debug print. The starting random-number of the last particle run is also listed on the Creation and Loss summary page of the regular output.

(2) The second entry NMDB is used to print out information about every NMDBth particle. The information consists of: (a) the particle history number, (b) the total number of collisions, (c) the total number of random numbers generated, and (d) the current random number at the beginning of the NMDBth history. This information is printed at the first of the particle history.

(3) and (4) These two entries are the inclusive limits for NPS (the history number) for event-log printing. This information is automatically printed for lost particles and can be printed on request for starting histories between these two entries. The information includes a step-by-step account of the life of a particular history such as where and how a particle is born, which surface it crosses and which cell it enters, what happens to it in a cell, etc.

(5) This controls the maximum number of events in the event log to print per history. The default is 600.

(6) If this sixth entry is non-zero, a binary file SAMFIL is produced containing timing statistics of the various parts of MCNP from a memory access pattern analysis. Page 10 of the January, 1979, CCF Newsletter (Vol. 39) tells what to do with this file. For example

```
TALLY SAMFIL MCNP / t p
.ALL.
.END
```

produces the BCD file HTAL0 that gives you a timing summary of the job you just ran with MCNP. This will tell you how much relative time was spent in every subroutine in MCNP, how much relative time was spent between labels in each subroutine, and a histogram representing these times. You can send this file to a printer with ALLOUT.

The timing routine is statistical; every few milliseconds it scores what routine the job is in. These scores, or number of hits NHIT, are listed on the HTAL0 output file along with percentages.

(7) A non-zero entry here causes a detailed print from the volume and surface area calculations. This is of use only to X-6 coders.

(8) A non-zero entry for this item causes the PLOT overlay to set up but not execute commands. This is of use only to X-6 coders.

(9) When the number of collisions in the current history NCH(IPT) is equal to the number entered for this ninth entry, a normally unused piece of code is traversed containing an otherwise unused label (365 in HSTORY) where a breakpoint can be set. This feature can be useful in conjunction with DBCTRL (see Appendix A) when you want to skip over a certain number of collisions and get to the one of interest.

4. File Creation (FILES) Card

A FILES card is available to assist the user whose modifications to the code require up to six additional sequential BCD or binary files such as an output file to be written by the TALLYX subroutine for post-processing or an input file to be read by a user-provided SOURCE subroutine.

The card format is:

FILES $unit_1$ $filename_1$ $bufsize_1$ $famsize_1$ $unit_2$ $filename_2$ ...

unit = logical unit number
filename = name of the file
bufsize = size of the buffer
famsize = family size for output files

Examples:

```
FILES   8  POSTP
FILES   7  SSOURCE  0  0  9  SHAPE
FILES   5  UOUT  6000B  290000  36  MYINP
```

The logical unit numbers reserved for use by MCNP are 2, 4, 15, and 59. They should therefore not be used on a FILES card.

If bufsize or famsize is zero or omitted, the default is 10000B or 100000B, respectively.

If the filename is DUMN1 or DUMN2, then the user may optionally use the execution line message to designate a file whose name might be different from run to run, for instance in a continue-run.

Example:

```
FILES    17  DUMN1
MCNP INP=TEST3 DUMN1=POST3
```

If a user wants to read or write any random files, he should not use the FILES card. Instead he should use the facilities provided by subroutine RANDY in MCNP. MCNP uses RANDY to create, open, close, read, write, etc., the regular random files such as RUNTPE and the cross-section libraries.

Caution: In a continue-run MCNP makes exactly the same system calls as in the initial run. In particular, the names of any user files in a continue-run will be the same as in the initial run which means the names are not automatically sequenced if a file already exists of the same name. A second output file from a continue-run will clobber an existing file of the same name. Furthermore, if you are using the FILES card for an input file and do a continue-run, you will have to provide the coding for keeping track of the record number and then positioning the correct starting location on the file when you continue or else MCNP will start reading the file at the beginning.

The FILES card is not legal in a continue-run input file because there is no way to readjust dynamically allocated storage for a possible requested increase in buffer space.

5. RUNTPE Tape (MCT) Card

An MCT card may be used to provide the execution message for the LTSS routine MCT used to save the RUNTPE on magnetic tape. This feature is invoked only under production conditions. Without this card, the RUNTPE is written to a new blank tape every two hours (by default) of CPU time with the message form

CHECK ID. [*user and group*] FILES RUNTX TAPES * END

The *user and group* information comes from the user number on the $BATCH card of the ORDER file. The input on the MCT card is the above message altered by the user as desired. No item can be more than nine characters long.

The MCT card can override the above standard message to the MCT routine, and thereby the user can cause the RUNTPE to be sent to a specified tape.

The name RUNTX is used to avoid detaining MCNP while waiting on a tape to be mounted. RUNTPE is copied to RUNTX (an existing RUNTX is destroyed) and MCT is started on some unused suffix, which allows MCNP to continue.

MCT. n on the MCNP execution line (see Appendix A) changes the default time interval from 120 minutes to n minutes.

6. Print Control (PRINT) Card

By default you will get limited output: some of the initiation overlay output, all the XACT output, and only part of the summary print from MCRUN. By including a PRINT card in your input file or by using the PRINT. option on the execution line, you will get the full output. For MCRUN this will add the weight balance by cell and event tables plus the nuclide activity by cell table.

The PRINT card or PRINT. option are legitimate in a continue-run even though they were not used in the initial run. In particular, if you did not get the full print initially but wish you had, you can still get it if you saved the RUNTPE. Just set up a continue-run input file containing a CONTINUE card, an NPS card with a negative entry, and a PRINT card or use the PRINT. option. This will run no more particles but will give you a full print at the time of the last history.

Don't forget to use the CC. option on your ALLOUT line since MCNP is an FTN code.

J. Criticality - KCODE

Only two additional cards are required for solution of k_{eff} with MCNP, the KCODE card and the KSRC card. If the KSRC card is absent an SRCTP file must be present. If a steady-state calculation is being made, be sure to use the TOTNU card. No external sources, such as the MCNP standard sources or a user-provided source, can be used with the KCODE option is used.

1. Criticality (KCODE) Card

This card is required for all k_{eff} calculations and has the form

KCODE NSRCK RKK IKZ KCT MSRK

The NSRCK entry is the nominal source size for each cycle and is frequently taken to be in the range of 300 < NSRCK ≤ 3000. There is no default value for NSRCK. The RKK entry is the initial guess for k_{eff} and has a default value of 1.0. The IKZ entry is the number of cycles to skip before beginning tally accumulation (important if initial source guess is poor). The default value is 5. The KCT entry specifies the number of cycles to perform before the problem ends. A zero entry means never (terminate by time only) and is also the default value. The MSRK is the maximum number of source points to provide storage for. If a SRCTP file is read for the initial source, MSRK is modified accordingly. The default value for MSRK is 4500 or 1.5*NSRCK, whichever is larger.

In running the KCODE option of MCNP several points are worthy of consideration. Positive entries on the PRDMP are interpreted as number of cycles rather than as the number of particles. A print is always given at the end of a run in addition to the cycle increments specified on the PRDMP card. The NPS card has no meaning and a warning error message is issued if it is used. If in the first cycle the source being generated overruns the current source, the initial guess (RKK) is probably too low. The code then proceeds to print a comment, continues without writing a new source, calculates k'_{eff}, reads the initial source back in, and begins the problem using k'_{eff} and not RKK. After the first cycle if the generated source overruns the current source the job aborts.

2. Source File (KSRC) Card

This optional card contains up to NSRCK (x,y,z) triplets which are locations of initial source points. The points supplied on the card or any SRCTP file are processed to get the initial source spatial distribution. If this card is not used, then an SRCTP source file must be supplied.

As explained on page 101, it is not necessary to input all NSRCK coordinate points. For example, given one point in each fissile region, the code will start (NSRCK/number of fissile regions) particles at each point. In many cases this is quite satisfactory, since MCNP will quickly calculate and use the new fission source distribution. Similarly, an SRCTP file from a previous calculation in which the geometries overlay somewhat can be used, since the points are deleted if not in any cell, in a void cell, a cell with no fissile material, or in a zero-importance cell. If the SRCTP file does not have NSRCK particles then the source will also be expanded or contracted as necessary. As a practical guide, if an SRCTP file is not available it is easiest to input one point, run the calculation

only one or two cycles, take the resultant SRCTP file, and rerun MCNP with the omission of the KSRC card. The SRCTP file is written at completion of a job or after every cycle, provided IKZ cycles have been calculated and that at least one minute has elapsed since the SRCTP file was last written.

V. SUMMARY OF MCNP INPUT FILE

This final section summarizes the input cards and when they are required by the three modes. Secondly, a table is given that summarizes some of the more important size limitations of the MCNP input.

A. Input Cards

The following table lists the various input cards and when they are required. An *R* indicates that the card is required for the specified Mode. An *O* indicates that the card is optional depending on the user's needs and depending upon the other cards that are included. An *X* indicates that this card should not be used with a specific Mode.

There are two kinds of defaults involved in the following table: (1) if a particular entry on a given card has a default value, that value is listed in the appropriate location on the card, and (2) the omission of some cards entirely from the input file sometimes indicates a default meaning and if this is the case the default description is preceded by an asterisk.

Table 3.2 Summary of MCNP Input Cards

Mode			Card and Defaults
0	1	2	
R	R	R	Problem title card
R	R	R	Cell cards
R	R	R	Surface cards
O	R	R	MODE 0

Cell parameter cards

R	R	X	IN	None	
X	0	R	IP	0 for Mode 2, and 1 for Mode 1	
0	0	0	VOL	0	174R
0	0	0	AREA	0	174R
X	0	X	PWT	-1	174R
0	0	X	EXTYN	0	174R
X	0	0	EXTYP	0	174R
0	0	X	FCN	0	174R
X	0	0	FCP	0	174R
0	0	0	PDn	1	174R
0	0	X	DXCPN	1	174R
X	0	0	DXCPP	1	174R

Source specification cards

R	R	R	SRCn	(Cannot be used with KCODE)
0	0	0	n=blank	(requires SUBROUTINE SOURCE)
0	0	0	n=1	0 0 0 1 1 .5 0
0	0	0	n=2	1 1 .5 0
0	0	0	n=3	1 1 .5 0
0	0	0	n=4	0 0 0 1 1 .5 0 6R
0	0	0	n=5	0 0 0 1 1 0 6R
a	a	a	SERG	
a	a	a	SPROB	
0	0	0	SBIAS	

Tally specification cards

0	0	0	Fna	R_o = 0 for n = 5/15
0	0	0	FCn	
0	0	0	En	100
0	0	0	Tn	1.0E123
0	0	0	FMn	1
0	0	0	Cn	1
0	0	0	EMn	1
0	0	0	TMn	1
0	0	0	CMn	1
0	0	0	CFn	
0	0	0	SFn	
0	0	0	FSn	
0	0	0	VAn	0

0	0	0	FUn	(Requires SUBROUTINE TALLYX)
0	0	0	DD	0 0
0	0	0	DXN	- - - - - - 0 0
0	0	0	DXP	- - - - - - 0 0
			Material specification cards	
0	0	0	Mm	
0	0	X	DRXS	*fully continuous
0	0	X	TOTNU	*prompt fission $\bar{\nu}$
0	0	0	AWTAB	
0	0	0	XS	
			Energy and Thermal cards	
0	0	X	ERGN	0 100 0
X	0	0	ERGP	100
0	0	X	NSPLT	*no neutron energy splitting
X	0	0	PSPLT	*no photon energy splitting
b	b	X	TEMPn	0
b	b	X	THTME	0
0	0	X	TI	*hydrogen and deuterium
0	0	X	MTm	
			Problem cutoffs	
0	0	X	CUTN	1.0E123 0 -0.5 -0.25
X	0	0	CUTP	1.0E123 .001 -.05 -.25
0	0	0	NPS	*Use CTME for limit
0	0	0	CTME	*Use execution line time limit
			User arrays	
0	0	0	IDUM	0 49R
0	0	0	RDUM	0 49R
			Peripheral cards	
0	0	0	PRDMP	end -15
0	0	0	LOST	10 10
0	0	0	DBCN	0 0 0 0 600 0 0 0 0
0	0	0	FILES	- - 10000b 100000b
0	0	0	MCT	*BATCH line information
0	0	0	PRINT	*short output

			Criticality	
c	c	X	KCODE	- 1 5 0 MAX(4500,1.5*NSRCK)
0	0	X	KSRC	

(a) The SERG and SPROB cards are required if n is non-blank and are optional if n is a blank. If n is a blank and these two cards are used, CALL ERGSMP must appear in the SUBROUTINE SOURCE.

(b) The TEMPn and THTME cards are required if E_{th} on the ERGN card is non-zero.

(c) The KCODE card is required for criticality calculations.

*This describes the effect of not using this particular card.

B. Storage Limitations

Table 3.3 summarizes some of the more important limitations that have to be considered when setting up a problem. It may be necessary to modify MCNP to change one or more of these restrictions for a particular problem.

Table 3.3
Storage Limitations[10]

Cross-section words MEMORY = 396500 = total machine memory LOC(DAS(LXSN)) = location of first cross-section word	MEMORY-LOC(DAS(LXSN))≥300000
Cells	MAXA = 175
Cell bounding surfaces	MP2-1=7*MAXA-1=1224
Otherside cells on cell cards	MP3-1=12*MAXA-1=2099
Entries on any single cell card	*100
Surfaces	JMAX=175
Total coefficients on all surface cards	MP5-1=5*JMAX-6=874
Entries on SRC card	MAXS=51
Energy groups on SERG,SPROB, SBIAS cards	MAXS-1=50
Tallies	NTALMX-6=40
Parameters on all tally cards	100*NTALMX=4100
Detectors	MXDT=20
Neutron DXTRAN spheres	*5
Photon DXTRAN spheres	*5
NSPLT or PSPLT card entries	*10
Nonstandard cross-section library files	NLXS=10
Cross-section library files	MP8=NLXS+6=16
Different nuclides	MAXE=40
Nuclides (repeatable) on all material cards	MEMAX=120
Entries on all material cards	MXMT=250
Nuclides in all cells	MP7=4*MAXA=700
Thermal nuclides on TI card	MAXE=40
Thermal times	MAXT=7
S(α,β) entries	MXMTX-6=99
Entries on IDUM card	*50
Entries on RDUM card	*50

*Set as a dimension in an array

CHAPTER 4

EXAMPLES

Cookbook examples of several topics are given in this Chapter. You will find that you can use many of the examples essentially as is. However, the purpose of this chapter is not to provide you with things you can directly plug into your input files but to give you instructive, real examples that you can follow and learn from. These examples should be studied in conjunction with the theory and instructions of Chapters 2 and 3.

I. GEOMETRY SPECIFICATION

Several more examples of the union operator will be given with the intent that you can study them to help you understand this method. In all examples, the cell numbers will be circled and the surface numbers will not be circled but will appear next to the surface they represent. All cells are voids, and in all examples the outside cell is legitimate (i.e., properly defined in the context of Chapter 2). Because the outside is legitimate, all the examples could be written by not specifying otherside cells. This is done in some of the examples.

All examples are in the LIX file called UNION which itself is in a LIX file MCNPEX under /X6CODE. The input file for the first example is called EXP1, etc. You are encouraged to experiment with these files by plotting and modifying them.

The next ten examples become progressively more difficult, and usually one example takes advantage of what you learned in the preceding example.

Remember that unless altered by parentheses, the hierarchy of operations is that intersections are performed first and then unions.

Example 1:

In this geometry of four cells defined by three spheres, cell 3 is disjoint. Cell 3 is the region inside surface 3 but outside surfaces 1 and 2 plus the region enclosed between surfaces 1 and 2:

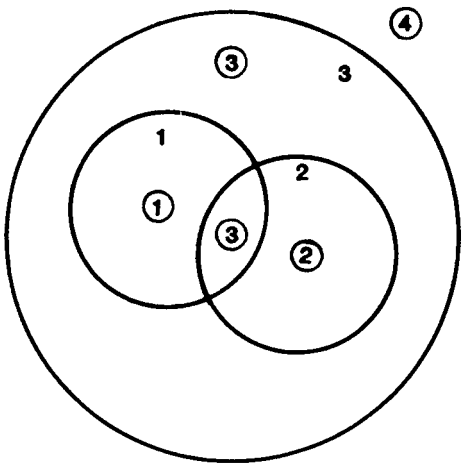


Figure 4.1

```
1     0   -1,3   2,3
2     0   -2,3   1,3
3     0   -3,4   1,1   2,2   :   -2,1   -1,2
4     0    3,3
```

Cell 3 could also be written as

```
3 0 (-3,4  1,1  2,2) :  (-2,1  -1,2)
```

Example 2:

In this example of Figure 4.2a, surfaces 2 and 4 are cylinders and the others are planes. Cells 1 and 2 are easy to specify:

```
1     0   1  -2  -3
2     0   3  -4  -5
```

Cell 3 is somewhat subtle, and you need to have in mind Figure 2.6 and its explanation on page 17. Remember that a union adds regions and an intersection gives you only what overlaps. Therefore the union of a defined region with an undefined region is the defined region, but the intersection of a defined region with an undefined region is an undefined region. Think in terms of addition for a union and multiplication for an intersection where 1+0=1 and 1·0=0.

Starting in cell 3 at surface 1 and going around clockwise, surface 1 and 2 form an illegitimate corner which means a union. The expression -1:2 defines the region that is everything in the universe with a negative sense

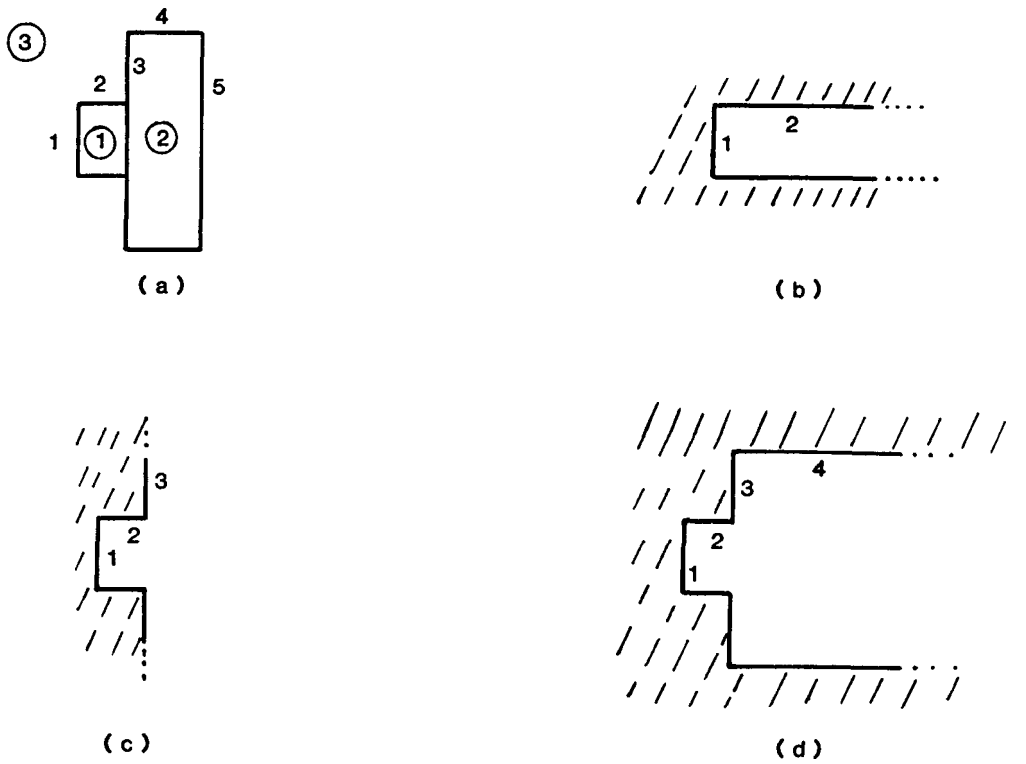


Figure 4.2

with respect to surface 1 *plus* everything in the universe with a positive sense with respect to surface 2. This region is the cross-hatched area in Figure 4.2b and includes everything in the universe except for an undefined region to the right of surface 1 and inside of surface 2 that goes off to infinity.

Continuing on around we pick up surface 3. The corner between surfaces 2 and 3 is legitimate so an intersection is in order for the region to the left of surface 3 with the region defined by the -1:2 union. Intersecting the half of the universe that has a negative sense with respect to surface 3 and -1:2, the only intersecting (or overlapping) area is shown in Figure 4.2c. The undefined tunnel to the right of surface 1 still exists since the intersection of a defined region with an undefined one is still undefined. The size of the undefined region has been increased, however.

The next surface encountered is 4 which brings in another union to account for the corner between surfaces 3 and 4. Adding all the universe with a positive sense with respect to surface 4 to the region defined by (-1:2) -3 produces the cross-hatched area of Figure 4.2d which increases the size of the undefined tunnel going off to the right but decreases the size of the total undefined region of Figure 4.c.

Between surfaces 4 and 5 is another illegitimate corner so we add all the region to the right of surface 5 with what has been determined so far, or

(-1:2) -3:4:5

There is more than one way to define cell 3. In fact the parentheses could have been omitted from the above discussion, but then the discussion (and not the cell) would have been incorrect because we wanted the intersection with -1:2 and -3 and not with 2 and -3. Another approach is to intersect the two regions -1:2 and -3:4 and then add that to the region to the right of surface 5 by

(-1:2) (-3:4) : 5

You might consider trying (-1:2)(-3:4:5), but this will not work. Intersecting -1:2 (remember the undefined tunnel to the right of surface 1) with -3:4:5 still leaves the region of the tunnel undefined.

Another approach is to forget about the reality of the geometry and for cell 3 take the inverse (or complement) of all the cells bounding cell 3 - which are cells 1 and 2. This says that cell 3 is all the region excluding that which has already been defined to be in cells 1 and 2. The advantage of this is that cells 1 and 2 are easy to specify plus you don't get bogged down in details for cell 3. Doing this, cell 3 becomes

(-1:2:3) (-3:4:5)

Note that the specifications for cells 1 and 2 are simply reversed. Intersections become unions and positive senses become negative and then each piece is intersected with the other. See Example 11 for another application of this technique.

Example 3:

Cell 1 is everything interior to the surfaces 1 and 2:

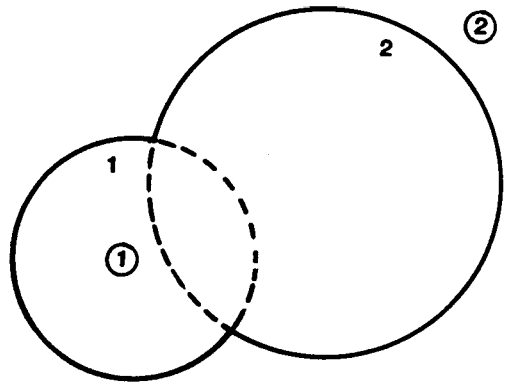


Figure 4.3

```
1     0   -1,2  :  -2,2
2     0    1,1      2,1
```

Example 4:

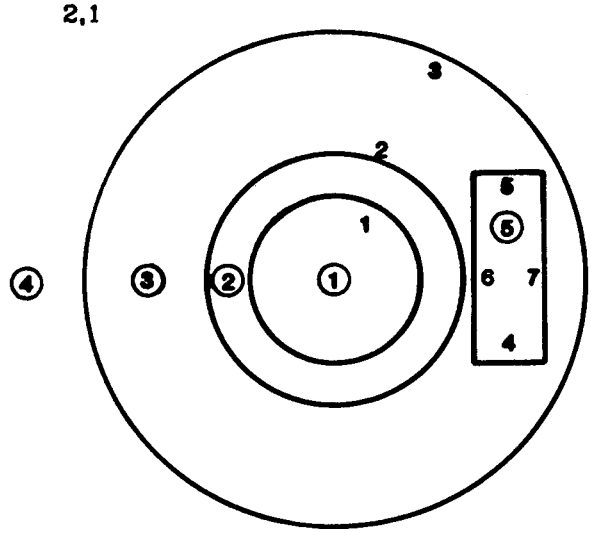


Figure 4.4

This is three concentric spheres with a box cut out of cell 3. Surface 8 is the front of the box and 9 is the back of the box:

```
1     0   -1,2
2     0   -2,3   1,1
3     0   -3,4   2,2 (-4,5:5,5:-6,5:7,5:8,5:-9,5)
4     0    3,3
5     0    4,3  -5,3  6,3  -7,3  -8,3  9,3
```

Example 5:

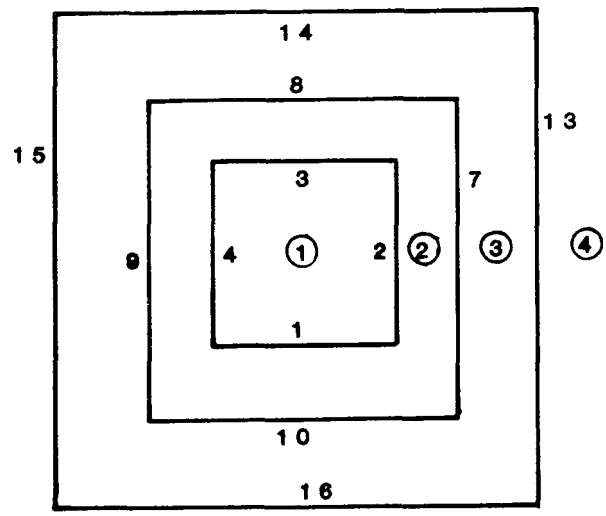


Figure 4.5

This is three concentric boxes which is very challenging to set up using only intersections; it takes a minimum of 14 cells to get around the problem of convex corners when using only intesections. Surfaces 5, 11, and 17 are the back sides of the boxes; 6, 12, and 18 are the fronts:

```
1     0  -2,2      -3,2      4,2      1,2      5,2      -6,2
2     0  -7,3      -8,3      9,3     10,3     11,3     -12,3
         (2,1  :   3,1  :  -4,1  :  -1,1  :  -5,1  :   6,1)
3     0 -13,4     -14,4     15,4     16,4     17,4     -18,4
         (7,2  :   8,2  :  -9,2  : -10,2  :  -11,2  :  12,2)
4     0  13,3  :  14,3  : -15,3  : -16,3  :  -17,3  :  18,3
```

This same example set up entirely with intersections can be seen by looking at the input file BOX which is in the LIX file UNION in MCNPEX.

Example 6:

This is two concentric spheres with a torus attached to cell 2 and cut out of cell 1:

```
1     0  -1,2   4,2
2     0  -2,3  (1,1  :  -4,1)
3     0   2,2
```

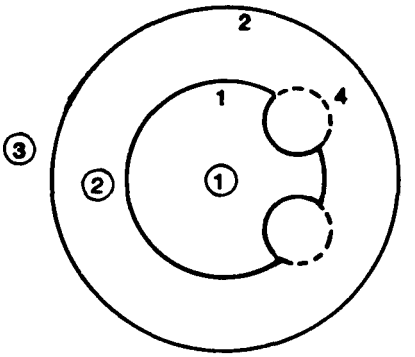


Figure 4.6

If the torus were attached to cell 1 and cut out of cell 2, this bug-eyed geometry would be:

```
1     0   -1,2  :  -4,2
2     0   -2,3      1,1   4,1
3     0    2,2
```

Example 7:

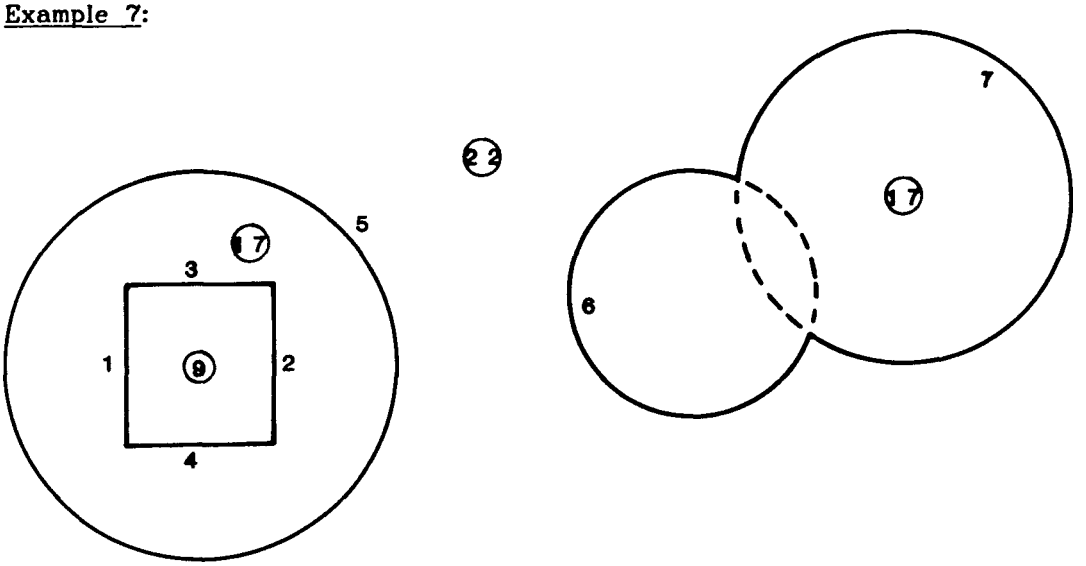


Figure 4.7

Cell 9 is a box cut out of the left-hand part of the spherical cell 17; surface 9 is the front of the box and 8 is the rear. Cell 17 is disjoint, and the right-hand part is the space interior to the spheres 6

and 7. An F4 tally in cell 17 would be the average flux in all parts of cell 17. An F2 surface tally on surface 7 would be the flux across only the solid portion of surface 7 in the figure.

```
9     0   -3,17 -2,17 4,17 1,17 8,17 -9,17
17    0   -5,22 (3,9 : -4,9 : -1,9 : 2,9 : 9,9 : -8,9) : -6,22 : -7,22
22    0    5,17 6,17 7,17
```

A variation on this problem is for the right-hand portion of cell 17 to be the intersection of surfaces 6 and 7 (the region bounded by the dashed lines in the above figure):

```
9     0   -3,17 -2,17 4,17 1,17 8,17 -9,17
17    0   -5,22 (3,9 : -4,9 : -1,9 : 2,9 :  9,9 : -8,9) : -6,22 -7,22
22    0    5,17 (6,17 : 7,17)
```

Example 8:

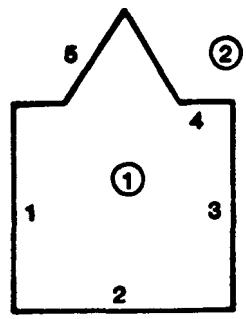


Figure 4.8

This is a box with a cone sitting on top of it. Surface 6 is the front of the box and 7 is the rear. You should understand this example before going on to the next one.

```
1     0    1,2     2,2    -3,2 (-4,2 : -5,2)    -6,2     7,2
2     0   -1,1 : -2,1 :  3,1 : 4,1     5,1    : 6,1 : -7,1
```

Example 9:

Surfaces 15 and 16 are cones, surface 17 is a sphere, and cell 2 is

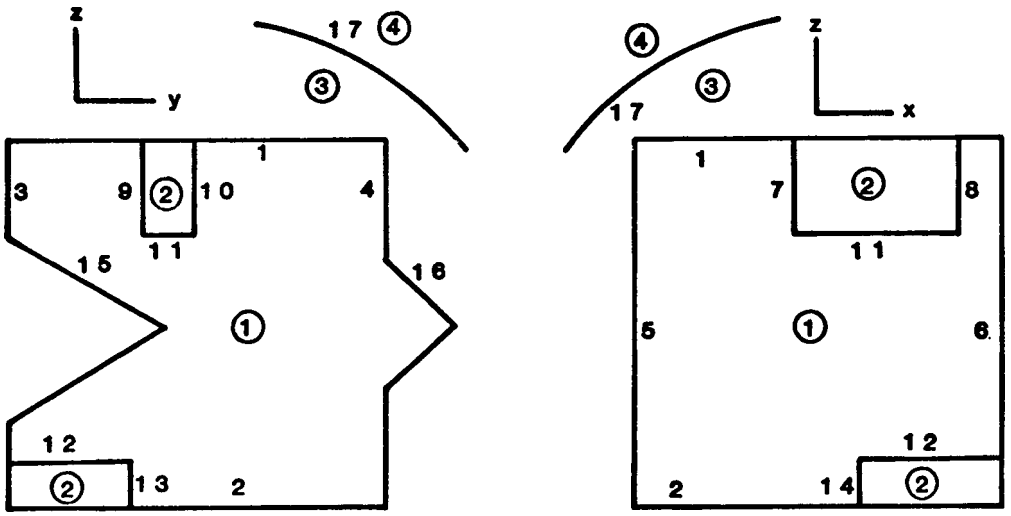


Figure 4.9

disjoint.

```
1     0   -1,3 2,3 3,3 (-4,3:-16,3) 5,3 -6,3 (12,2:13,2:-14,2)
          (10,2:-9,2:-11,2:-7,2:8,2) 15,3
2     0   -10,1 9,1 11,1 7,1 -8,1 -1,3:2,3 -12,1 14,1 -6,3 -13,1 3,3
3     0   -17,4 (1,1,2:-2,1,2:-5,1:6,1,2:-3,1,2:-15,1:16,1 4,1)
4     0    17,3
```

Example 10:

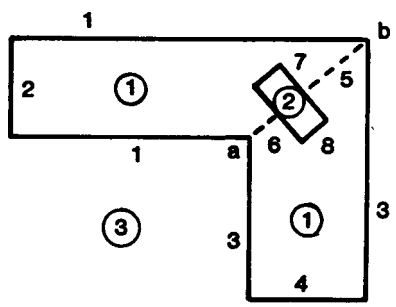


Figure 4.10

Cell 1 consists of two cylinders joined at a 45° angle. Cell 2 is a disk consisting of a cylinder (surface 8) bounded by two planes. Surface 5 is a diagonal plane representing the intersection of the two cylinders. The problem is to specify the disk (cell 2) in one cell formed by the two cylinders (cell 1). A conflict arises in specifying cell 1 since, from the outside cell 3, corner *a* between surfaces 1 and 3 is legitimate but on the other side of the cell the same two surfaces form an illegitimate corner at *b*. The dilemma is solved by composing cell 1 of two disjoint cells, each bounded by surface 5 between the corners *a* and *b*. When the two parts are joined to make cell 1, surface 5 does not appear – convince yourself by plotting it (see page 301).

```
1     0   (2 -1 -5 (7:8:-6)):(4 -3 5(-6:8:7))
2     0   -8  6 -7
3     0   (-2:1:5)(-4:3:-5)
```

Example 11:

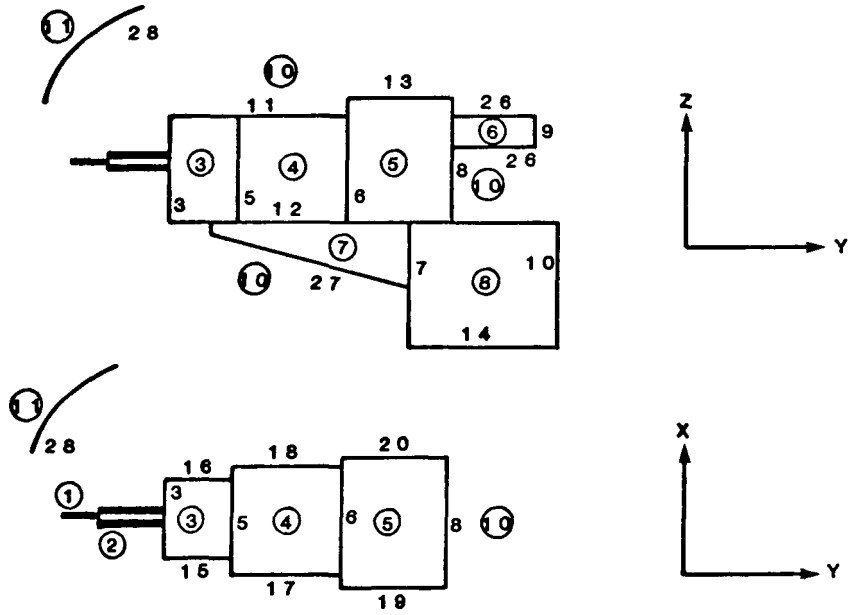


Figure 4.11

This example is the description of the target insertion mechanism for the Antares laser fusion facility at LASL. The input file is called ANTARES and is in the LIX file UNION which is in MCNPEX. It is

the most complicated geometry of any of the examples, but it can be described very simply.

Referring to Example 2, you will see that this example is virtually identical. There is just a lot more of it. It is possible to set this geometry up by any of the ways mentioned in Example 2. However, going around the outer surfaces of the cells inside cell 10 is tedious. There is a problem of visualization and also the problem of coming up with undefined tunnels going off to infinity as in Example 2. By going around each surface, when one error is made and you try to correct it, more than likely more errors will be introduced until the situation becomes hopeless.

The way to handle this geometry is by the last method in Example 2. Set up the cell/surface relations for each interior cell and then just take the complement for cell 10. For the interior cells,

```
1     0    1 -2 -23
2     0   -3 25 -24 2
3     0    3 -5 12 -15 16 -11
4     0    5 -6 12 -17 18 -11
5     0    6 -8 12 -13 -19 20
6     0    8 -9 -26
7     0   -12 4 -7 -27
8     0   -12 7 -10 14 -21 22
9     0    2 -3 -25
```

Cell 10 is inside cell 11 with the spherical surface 28 between them. Considering cell 10 to be everything outside cells 1 through 9 but inside cell 11 and not thinking about any other details of the geometry,

```
10    0    (-1:2:23)(3:-25:24:-2)
           (-3:5:-12:15:-16:11)
           (-5:6:-12:17:-18:11)
           (-6:8:-12:13:19:-20)
           (-8:9:26)(12:-4:7:27)
           (12:-7:10:-14:21:-22)
           (-2:3:25)  -28
```

II. TALLY EXAMPLES

This section contains several examples of the FMn and FSn tally cards plus the TALLYX subroutine. Refer also to page 146 for the FMn card, to page 150 for the FSn card, and to page 151 for TALLYX.

A. FMn Examples

Example 1: Consider the following input cards.

```
F4      10
FM4     1.0       999     102
M999    92238     1
```

This F4 tally is the track length estimate of the number of ^{239}U atoms produced per ^{238}U atom in cell 10. The FM4 card parameters are

C = 1.0	normalization factor
M = 999	material number for ^{238}U as defined on the material card
R_1 = 102	ENDF reaction number for capture cross section

Note that the standard F6 and F7 tallies can now be duplicated by an F4 tally with an appropriate FM4 card.

Example 2: Consider a point detector.

```
F25       0    0    0    0
FM25      0.00253   1001   -6   -8
M1001     92238    .9     92235     .1
```

This F5 tally is the fission heating of material 1001 at the origin. Material 1001 does not have to actually be in a cell at the origin. The FM25 card constants are:

C = 0.00253	atoms per gram of material 1001
M = 1001	material number for material being heated
R_1 = -6	reaction number for total fission cross section
R_2 = -8	reaction number for fission Q

Fission heating and neutron heating can now be calculated at a point or on a surface making the old F3 tally obsolete.

More examples:

F5	0	0	0	0	Neutron heating of M at a point
FM5	C	M	−1	−4	detector
F15Y	10	5	0		Photon heating of M at a ring detector
FM15	C	M	−5	−6	
F1	1	2	3		Number of neutron tracks crossing
FM1	1	0			surfaces 1, 2, and 3 per neutron started
F35	0	0	0	0	Number of collisions per source photon
FM35	1	0			that contribute to point detector
M99	3007	1			^{7}Li tritium production in cell 10
F4	10				
FM4	1	99	24		
M98	3006	1			^{6}Li tritium production in cell 11
F24	11				
FM24	1	98	107		
F104	8				Number of reactions of type R in cell 8
FM104	ρ	M	R		of material M of atom density ρ

B. FSn Examples

FSn card: This card allows you to subdivide your tally into geometry segments without having to overspecify the problem geometry with unnecessary cells for this purpose.

The entries on the FS card are simply the names and senses of surfaces which define how to segment any surface or cell tally.

Example 1: Consider a 1-MeV point isotropic source at the center of a 2 cm cube of carbon. It is desired to calculate the flux through a 1 cm^2 window in the center of one face on the cube. The input file calculating the flux across one entire face is

```
EXAMPLE 1, SIMPLE CUBE
1          1 -2.22  1,2 2,2 -3,2 -4,2 -5,2 6,2
2          0        -1,1 -2,1 3,1 4,1 5,1 -6,1

1          PY  0
2          PZ -1
3          PY  2
4          PZ  1
5          PX  1
6          PX -1

SRC1       0 1
SERG       1 1
SPROB      0 1
M1         6012 -1
IN         1 0
F2         3
```

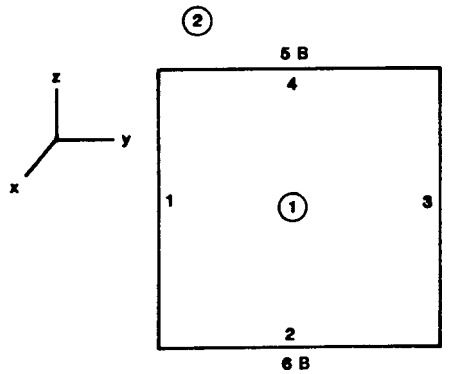
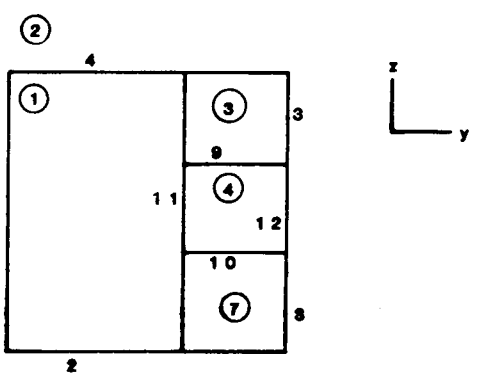


Figure 4.12

To be able to get the flux in the desired window without using the FS card the simple cube of one cell has to be divided into six cells. The input file becomes much more tedious to set up:

```
EXAMPLE 1, OVERSPECIFIED CUBE
1          1 -2.22  1,2 2,2 -4,2 -5,2 6,2 -11,3,4,5,6,7
2          0        -1,1 -2,1,7 3,3,4,6,7 4,1,3 5,1,3,4,7
                    -6,1,3,6,7 12,5
3          1 -2.22  11,1 -3,2 -4,2 9,4,5,6 -5,2 6,2
4          1 -2,22  11,1 -3,2 -9,3 10,7 -5,2 7,5
5          1 -2.22  11,1 -12,2 -9,3 10,7 -7,4 8,6
6          1 -2.22  11,1 -3,2 -9,3 10,7 -8,5 6,2
7          1 -2.22  11,1 -3,2 -10,4,5,6 2,2 -5,2 6,2

1          PY  0
2          PZ -1
3          PY  2
4          PZ  1
5          PX  1
6          PX -1
7          PX  .5
8          PX -.5
```



```
9           PZ   .5
10          PZ  -.5
11          PY   1.5
12          PY   2

SRC1        0 1
SERG        1 1
SPROB       0 1
M1          6012 -1
IN          1 0 1 4R
F2          12
```

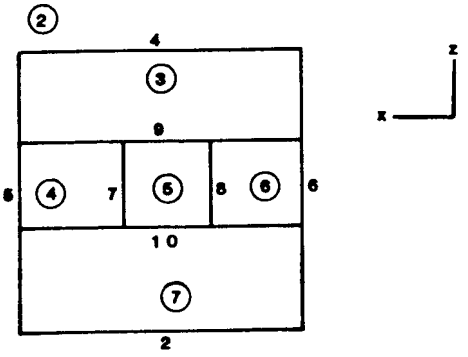


Figure 4.13

Obviously there is a lot more user time required to set this problem up by overspecification. The possibility for error increases significantly also. Using the FS card, the simple cube is still maintained. Five more input cards are required,

```
FS2  7  -10  -8  9
```

plus the four segmenting surfaces listed on the FS card:

```
 7    PX     .5
 8    PX    -.5
 9    PZ     .5
10    PZ    -.5
```

These segmenting surface cards are listed with the other surface cards, but they are not part of the actual geometry and hence do not complicate the cell-surface relationships.

In the above example, the F2 tally is subdivided into five separate tallies: (1) the first tally is the flux of particles crossing surface 3 but with a positive sense to surface 7; (2) the second is the remaining flux with negative sense to surface 7 crossing surface 3 but with a negative sense to surface 10; (3) the third is the remaining flux (negative sense to 7 and positive sense to 10) crossing 3 but with a negative sense to 8; (4) the remaining flux with positive sense to 9; and (5) everything else. In this example, the desired flux in the window is in the fifth subtally - the "everything else" portion.

The FS segmenting card could have been set up other ways. For example:

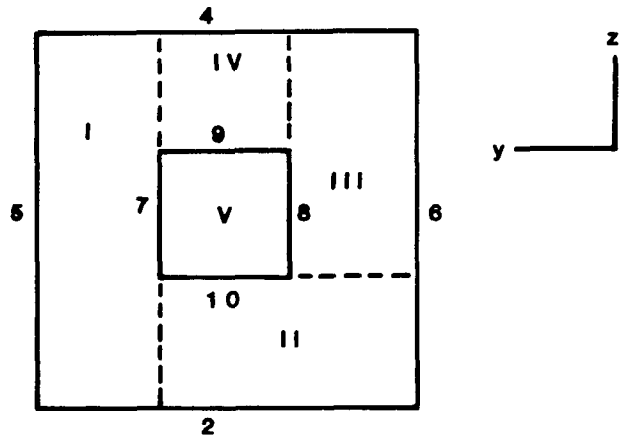


Figure 4.14

```
FS2    -10    7     9    -8
FS2      8    9   -10     7
```

each work, but the order of the subtallies is changed. A way to avoid the five subtallies and to get only the window is to use TALLYX that is described later.

Comparing the cost of running this job both ways, which is proportional to $(\sigma^2 t)^{-1}$, the overspecified problem cost twice as much to run as the segmented problem did.

In this particular example, an even simpler way to specify the cell cards is

```
1     1  -2.22   1      2     -3     -4     -5      6
2     0          -1  :  -2  :   3  :   4  :   5  :  -6
```

Example 2: Consider a source at the center of a 10 cm radius sphere called cell 1, and we want to determine the fission heating in a segment of the sphere defined by the intersection of the 10 cm sphere, an 8 cm inner sphere, and a 20° cone whose vertex is at the source and is about the Y-axis. Instead of breaking up the simple 10 cm sphere into four cells, use

```
F7     1
FS7   -2   -3
```

where surface 2 is the 8 cm surface and surface 3 is the cone. This breaks the F7 tally up into three portions: (1) the heating inside the 8 cm sphere, (2) the heating outside the 8 cm sphere but within the cone - this

is the desired portion; and (3) everything else, which is a 2 cm shell just inside the 10 cm sphere but outside the cone.

C. TALLYX Examples

Example 1: In the example of the FSn card to get the flux through a window on the face of a cube, instead of using the FS2 card which established five subtallies, TALLYX could have been used to get only the desired window tally. Two input cards are used:

```
FU2      1
RDUM  -.5   .5  -.5   .5
```

The following subroutine (which is implemented just like a user-provided SOURCE subroutine in Appendix A by deleting lines TX.2 through TX.7) does the job. Note that IB=1 upon entry into TALLYX.

```
      SUBROUTINE TALLYX(T,IB)
*CALL CM
      IF(X.LT.RDUMMY(1).OR.X.GT.RDUMMY(2))IB=0
      IF(Z.LT.RDUMMY(3).OR.Z.GT.RDUMMY(4))IB=0
      RETURN
      END
```

The subroutine was generalized a bit by using the RDUM input card, although the card could have been avoided by hard-wiring the dimensions of the window into TALLYX.

Example 2: Dump the first 22 words of the PBL array to a BCD file each time a neutron crosses surface 20. The cards in the input deck are

```
F2    20
FU2   1
```

The user-provided subroutine is

```
      SUBROUTINE TALLYX(T,IB)
*CALL CM
      WRITE(7,20) (PBL(I),I=1,11),IA,JA,NPA,IEX,NFC,IPT,NODE,
     1 IDX,SPARE1,SPARE2,SPARE3
```

```
   20 FORMAT(5E14.6/6E14.6/8I10/3E14.6)
      RETURN
      END
```

For this subroutine, a FILES card with entries 7 UOUT is required in the INP file.

Every time surface 20 is crossed and the F2 tally is scored, TALLYX is called and part of the PBL array is written to the file UOUT. If more discrimination is desired, such as dump the PBL array only for neutrons with energy between 2.5 and 4.5 MeV and crossing surface 20 at less than 30° with respect to the normal (assume surface 20 has been defined by a PY card), add the following two lines before the WRITE statement:

```
      IF(V.LT.0.866)RETURN
      IF(ERG.LT.2.5.OR.ERG.GT.4.5)RETURN
```

To write a binary file, the same FILES input card is used, but the WRITE statement in TALLYX is unformatted:

```
       SUBROUTINE TALLYX(T,IB)
 *CALL CM
       WRITE(7) (PBL(I),I=1,11),IA,JA,NPA,IEX,NFC,IPT,NODE,
      1 IDX,SPARE1,SPARE2,SPARE3
       RETURN
       END
```

The advantage of a BCD file is that it is easy to look at and manipulate, but it requires more I/O time and a larger file. A binary file is more compact than a BCD file and requires less I/O time to write; however, it may be more difficult to use.

Example 3: Calculate the number of neutron tracks exiting cell 20 per source neutron. This is also done in Chapter 5 with the TEST1 example using the FMn card. The input cards are

```
F4    20
FU4   1
```

and TALLYX becomes

```
      SUBROUTINE TALLYX(T,IB)
*CALL CM
      T=VOL(IA)/AMIN1(PMF,DLS)
      IF(AMIN1(PMF,DLS).EQ.PMF)IB=0
      RETURN
      END
```

The quantity T=1.0 is scored everytime a track exits cell 20. All the variables used in this subroutine, such as PMF, are available to TALLYX from the MCNP COMMON.

Example 4: Divide the point detector scores into separate tallies (or bins) depending upon which of the 20 cells in a problem geometry caused the contributions. The input cards are

```
F5    0  0  0  0
FU5   0  19R
```

and TALLYX is

```
      SUBROUTINE TALLYX(T,IB)
*CALL CM
      XM=PBLSAV(12)
      IB=MOVE(XM)
      RETURN
      END
```

The FU5 card establishes 20 separate F5 tallies (or bins), one for each cell in the problem.

Example 5: Determine the quantity

$$\int \varphi(E)\ f(E)\ dE$$

in cell 20 where $f(E) = e^{at}$. The input cards are

```
F4   20
FU4  α
```

and TALLYX is

```
      SUBROUTINE TALLYX(T,IB)
*CALL CM
      T=T*EXP(TDS(IPTAL(19,ITAL)+1)*TME)
      RETURN
      END
```

The FU4 card establishes a single user tally, and the value of α is stored in TDS(IPTAL(19,ITAL)+1).

Example 6: Tally the number of neutrons passing through cell 16 which have had 0,1,2,3, or 4 collisions. The input cards are

```
F4   16
FU4  0   4R
```

and TALLYX is

```
      SUBROUTINE TALLYX(T,IB)
*CALL CM
      IB=NCH(1)+1
      T=T*VOL(IA)/AMIN1(PMF,DLS)
      RETURN
      END
```

In the five user tallies, T is the number of neutrons per source neutron passing through cell 16 that has undergone 0,1,2,3, or 4 collisions, respectively. Note that the FU4 card has five dummy entries to establish the five user tallies. Note also that in this example, the number of neutrons is calculated so that T=T*renormalization factor, where in Example 3 the number of neutron tracks is calculated so that T=renormalization factor. Again the value of NCH(1) (number of collisions so far), VOL, and IA are all available from COMMON. Finally, note that if NCH(1) $\geq$ 5 (five or more collisions) no tally is made because IB is outside the range of five tallies as specified on the FU4 card. If an E4 card were added, the number of neutrons would be tallied as a function of energy.

III. SOURCE SUBROUTINE

If you need to provide your own SOURCE subroutine, page 140 summarizes the variables that must be set, and Appendix A shows how to get the subroutine into MCNP with UPDATE. The following example illustrates a somewhat contrived SOURCE, but it shows how to take advantage of some very useful features in MCNP. In particular, the MCNP subroutines CBIAS, ERGSAMP, CHGNAM, and CHKCELL are demonstrated.

The 1-MeV monoenergetic source is isotropic but biased in a fixed cone along the y-axis. CBIAS determines the direction cosine of the source particles, of which a fraction p start inside the fixed cone of opening $\nu = \cos\theta$, and (1-p) start outside the cone. The two starting weights, for particles starting inside and outside the cone respectively, are calculated once at the start of the source. MCNP calculates these weights only for the standard sources.

The source is uniform in X and Z but distributed in the Y-direction and also biased in the Y-direction. The energy bias cards SBIAS, SPROB, and SERG are used for this purpose and not to determine energy. The subroutine ERGSAMP is called, and upon return Y is set equal to the energy from it. The actual source energy is then set to 1 MeV.

Source particles start in more than one cell. A list of possible source cells is given on the SRC input card. These are problem names. The subroutine CHGNAM(M,J,K) is called to get the program names. In the calling sequence, M=1 means the call is for cells, J is the problem name, and K is the program name.

Finally, for each of the program cells the subroutine CHKCELL(I1,M,J) is called to determine in which cell the point (X,Y,Z) lies. In the calling sequence, I1 is the program number. M=0 gives a return of J=0 if the point is in cell IAP and J≠0 if it is not in the cell. The surface parameter JA is set to zero before the call to CHKCELL since this is a cell source and not a surface source. By this procedure, the sense of all surfaces bounding the cell is checked.

```
*IDENT SRCEX
*D,SO.11
      DATA FLAG/1./
C  ENTRIES ON SRC CARD:
C  SRC(1) - CONE BIAS PROBABILITY
C  SRC(2) - OPENING OF CONE
```

```
C  SRC(3) - RADIUS OF SOURCE CYLINDER
C  SRC(4) TO SRC(13) - LIST OF POSSIBLE SOURCE CELLS
C
C  THE SOURCE ILLUSTRATES USE OF SUBROUTINES
C       CHKCELL,CHGNAM,ERGSMP,CBIAS
C
      IF(FLAG.EQ.0) GO TO 10
C  TRANSFER DATA FROM SRC TO SRP BLOCK FOR CBIAS
C  CALCULATE STARTING WEIGHTS IN AND OUT OF BIAS CONE
      SRP(5)=SRC(1)
      SRP(6)=SRC(2)
      SRP(7)=(1.-SRP(6))/(2.*SRP(5))
      SRP(8)=(1.+SRP(6))/(2.*(1.-SRP(5)))
      FLAG=0.
   10 WGT=1.
      CALL ERGSMP
C  SBIAS,SPROB,SERG CARDS USED IN ERGSMP
C  TO DETERMINE Y POSITION AND NOT ENERGY
C  SOURCE POINTS UNIFORM IN AREA IN X AND Z
C  SOURCE DISTRIBUTION IN Y GIVEN BY SPROB CARD
C  SOURCE BIAS IN Y GIVEN BY SBIAS CARD
      Y=ERG
      RN=RANF(1)
      R=SRC(3)*AMAX1(RN,RANF(1))
      TH=6.283185*RANF(1)
      NRN=NRN+3
      X=R*COS(TH)
      Z=R*SIN(TH)
C  SOURCE STARTS IN MORE THAN ONE CELL
      JA=0
      DO 20 I=1,10
      CEL=SRC(I+3)
      CALL CHGNAM(1,IFIX(CEL),IAP)
      CALL CHKCEL(IAP,0,J)
      IF(J.EQ.0) GO TO 30
   20 CONTINUE
      CALL EXPIRE(3,6HSOURCE,6,
     1 42HSOURCE IS NOT IN ANY CELLS ON THE SRC CARD)
   30 IA=IAP
      ERG=1.
```

```
      TME=0.
C  SOURCE IS ISOTROPIC BUT BIASED IN CONE ALONG Y-AXIS
      CALL CBIAS
```

The above SOURCE is available from the LIX file MCNPEX under /X6CODE and is called SRCEX. Another file in MCNPEX is INPSRC which is an MCNP input file that uses SRCEX with an appropriate SRC card.

IV. SRCDX SUBROUTINE

If a user has supplied a subroutine SOURCE that does not emit particles isotropically (uniform emission in all directions) and is using either a detector tally or DXTRAN in the calculations then subroutine SRCDX must also be supplied to MCNP. The structure of this subroutine is the same as for subroutine SOURCE, except that usually only a single parameter, PSC, needs to be specified for each detector or pair of DXTRAN spheres. PSC as defined in SRCDX is used to calculate the direct contribution from the source to a point detector, to the point selected for the ring detector, or DXTRAN sphere. Other parameters may also be specified in SRCDX. For example, if a quantity such as particle energy and/or weight is directionally dependent, its value must be specified in both subroutine SOURCE and SRCDX. When using detectors and a subroutine SOURCE with an anisotropic distribution, the direct source contribution to the detectors should be checked carefully to see if it is close to the expected result.

In general, it is best to have as many parameters as possible not directionally dependent in subroutine SOURCE. Those parameters which are directionally dependent must be dealt with in subroutine SRCDX when point detectors are in the problem. For example, if subroutine SOURCE specifies a source on a surface that can emit particles into more than one cell and detectors or DXTRAN are being used, SRCDX must also determine the program cell number IA for the source particle to each detector. An alternate method that eliminates the cell determination in SRCDX is not to use a surface source, but to specify the source an epsilon distance inside one of the cells.

The most general function for emitting a particle from the source in the laboratory system can be expressed as $p(\mu,\varphi)$ where μ is the cosine of the polar angle and φ is the azimuthal angle in the coordinate system of

the problem. Most anisotropic sources are azimuthally symmetric and $p(\mu,\varphi) = p(\mu)/2\pi$. The quantity $p(\mu)$ is the probability density function for the μ variable only (i.e., $\int p(\mu)\, d\mu = 1$, $p(\mu) \geq 0$). PSC is $p(\mu_o)$, where μ_o is the cosine of the angle between the direction defining the polar angle for the source and the direction to a detector or DXTRAN sphere point in the laboratory system. (MCNP includes the 2π in the calculation automatically.) Note that $p(\mu_o)$ and hence PSC may have a value greater than unity and must be non-negative. It is valuable to point out that every source must have a cumulative distribution function based on $p(\mu,\varphi)$ from which to sample angular dependence. The probability density function $p(\mu,\varphi)$ needs only to be considered explicitly for those problems with detectors or DXTRAN.

Table 4.1 gives the equations for PSC for six continuous source probability density functions. More discussion of probability density functions is given in the detector theory section of Chapter 2 (see page 57). The isotropic case is assumed in MCNP; therefore SRCDX is required only for the anisotropic case.

As an example of calculating μ_o, consider a spherical surface cosine source (type 2 in Table 4.1) with several point detectors in the problem. Assume that a point on the spherical surface has been selected at which to start a particle. The value of μ_o for a detector is given by the scalar (or dot) product of the two directions; i.e.,

$$\mu_o = uu' + vv' + ww' \qquad (4.1)$$

where u, v, and w are the direction cosines of the line from the source point to the point detector location and u', v', and w' are the direction cosines for either the outward normal if the surface source is outward or the inward normal if the source is inward.

If $u = u'$, $v = v'$, and $w = w'$, then $\mu_o = 1$ indicating that the point detector lies on the normal line. The value of PSC for the detector point is

$$\begin{aligned} PSC &= 2|\mu_o|, && \mu_o > 0 \\ & && (\mu_o < 0) \\ &= 0, && \mu_o \leq 0 \\ & && (\mu_o \geq 0) \end{aligned}$$

where the parenthetical values of μ_o are for the inward directed cosine distribution.

For $|\mu_o|$ less than 0.25, PSC is less than 0.5 which is the value for an isotropic source. This means that source emissions for these values of $|\mu_o|$ are less probable than the isotropic case for this source distribution. The converse is also true. Note that if $|\mu_0|$ is greater than 0.5, then PSC is greater than one which is perfectly valid.

Table 4.1

Continuous Source Distributions
and their Associated PSC's

	Source Description	Source Distribution	PSC	Range of μ_o
1.	Isotropic	Uniform	0.5	$-1 \le \mu_o \le 1$
2.	Surface Cosine	μ	$2\|\mu_o\|$	$0 \le \mu_o \le 1$ (or $-1 \le \mu_o \le 0$)
			0	$-1 \le \mu_o < 0$ (or $0 < \mu \le 1$)
3.	Point Cosine	$\|\mu\|$	$\|\mu_o\|$	$-1 \le \mu_o \le 1$
4.	Point Cosine*	$a+b\mu$	$\frac{2(a+b\mu_o)}{2a+b}$	$0 \le \mu_o \le 1$
			$\left(\frac{2(a+b\mu_o)}{2a-b}\right)$	$(-1 \le \mu_o \le 0)$
			0	$-1 \le \mu_o < 0$ (or $0 < \mu_o \le 1$)
5.	Point Cosine*	$a+b\mu, a\neq 0$	$\frac{a+b\mu_o}{2a}$	$-1 \le \mu_o \le 1$
6.	Point Cosine*	$a+b\|\mu\|$	$\frac{a+b\|\mu_o\|}{2a+b}$	$-1 \le \mu_o \le 1$

*The quantities a and b must have values such that PSC is always non-negative and finite over the range of μ_o.

Subroutine SRCDX must be written to calculate PSC for each detector in the problem. An example of a subroutine SRCDX with appropriate Update lines for a surface outward cosine distribution is shown in Figure 4.15. This is basically the technique that is used in MCNP to calculate PSC for source type SRC2; the only difference is that MCNP uses the cosines of the direction from the center of the sphere used to select the source point because this is the normal to the spherical surface. The primed direction cosines were calculated in Figure 4.15 to aid in illustrating this example. The direction cosines u, v, and w as defined in Equation (4.1) have already been calculated in subroutine DDDET when SRCDX is called and are available through COMMON.

```
*DELETE SX.2,SX.8
      SUBROUTINE SRCDX
*CALL CM
C     CALCULATE PSC FOR AN OUTWARD SURFACE (SPHERE) COSINE
C     DISTRIBUTION.
C     FIND THE DIRECTION COSINES FOR THIS EXAMPLE BASED
C     ON THE SOURCE POINT ON THE SPHERE (X,Y,Z).
      UPRIME = (X - SRC(1))/SRC(4)
      VPRIME = (Y - SRC(2))/SRC(4)
      WPRIME = (Z - SRC(3))/SRC(4)
C     (SRC(1),SRC(2),SRC(3)) ARE THE COORDINATES OF THE CENTER
C     OF THE SPHERE FROM THE SRC CARD.  SRC(4) IS THE
C     SPHERE RADIUS.
C     U, V, AND W HAVE BEEN CALCULATED FOR THE CURRENT
C     POINT DETECTOR IN SUBROUTINE DDDET
      PSC=2.*AMAX1(0,U*UPRIME+V*VPRIME+W*WPRIME)
      RETURN
      END
```

Figure 4.15

The Update cards in Figure 4.16, as well as in the following two examples, are the recommended procedure for replacing the existing dummy SRCDX subroutine. Note that line SX.6 must always be deleted if a user provides an SRCDX.

For many sources, a discrete probability density function will be

used. In this situation, a cumulative distribution function $P(\mu)$ is available and is defined as

$$P(\mu) = \int_{-1}^{\mu} p(\mu')d\mu' \text{ and } P_{i+1} = \sum_{j=1,i} p_j \Delta\mu_j$$

where p_j is an average value of the probability density function in the interval $\Delta\mu_j$. Thus, the probability density function is a constant p_j in the interval $\Delta\mu_j$. For this case, there are N values of P_i with $P_1 = 0$, $P_{N+1} = 1.0$, and $P_{i-1} < P_i$. Each value of P_i has an associated value of μ_i. Since PSC is the derivative of $P(\mu_o)$, then

$$PSC = \frac{P_i - P_{i-1}}{\mu_i - \mu_{i-1}}, \quad \mu_{i-1} \leq \mu_o < \mu_i \qquad (4.2)$$

This is an average PSC between μ_{i-1} and μ_i and is also an average value of $p(\mu)$ in the specified range of μ.

Frequently, the cumulative distribution function is divided into N equally probable intervals. For this case,

$$PSC = \frac{1}{N} \frac{1}{\mu_i - \mu_{i-1}}$$

This is precisely the form that is used in MCNP for calculating contributions to the point detector for elastic scattering with N = 32.

An example of a subroutine SRCDX for a discrete probability density function is shown in Figure 4.16. The quantity μ_o is calculated as in Equation (4.1) where u', v', and w' are the direction cosines of the direction defining the polar angle of the source. The first three positions in the SRC array for the SRCDX in Figure 4.16 are these direction cosines for this source reference direction and have been entered on the SRC card. The RDUMMY array is filled from the RDUM card and contains SRC(4) pairs of entries (P_i,μ_i) where $P_1 = 0$, $P_{i-1} < P_i$, $P_{SRC(4)} = 1$, SRC(4) $\leq$ 25 (the RUMMY array is limited to 50 entries), μ_1 is the smallest value of μ, $\mu_{i-1} < \mu_i$, and $\mu_{SRC(4)}$ is the largest value of μ for the source. If μ_o is the range of μ of the source, the proper μ bin is found and PSC calculated using Equation (4.2).

```
*DELETE SX.2,SX.8
      SUBROUTINE SRCDX
*CALL CM
C     CALCULATION OF PSC FOR A GENERAL DISCRETE
C     PROBABILITY DISTRIBUTION FUNCTION.
C     U, V, AND W HAVE BEEN CALCULATED FOR THE
C     CURRENT DETECTOR IN SUBROUTINE DDDET.
      I=2*SRC(4)
      UZERO=U*SRC(1) + V*SRC(2) + W*SRC(3)
      IF(UZERO.GT.RDUMMY(2).AND.UZERO.LT.RDUMMY(I))GO TO 10
      PSC=0.
      RETURN
10    DO 20 I=2,SRC(4)
      I1=2*I
      IF(UZERO.LT.RDUMMY(I1)) GO TO 30
20    CONTINUE
      I1=2*SRC(4)
30    PSC=(RDUMMY(I1-1)-RDUMMY(I1-3))/((RDUMMY(I1)-RDUMMY(I1-2))
      RETURN
      END
```

Figure 4.16

The third SRCDX shown in Figure 4.17 illustrates a procedure for inputting hand-calculated PSC's into MCNP. The values of PSC for this example are entered in sequence on the SRC card for each detector. This approach is not applicable to ring detectors because the detector location can be anywhere on the ring. The same is true for DXTRAN because the point can be selected anywhere on the outer sphere and therefore PSC will not, in general, have a constant value for each set of entries on the DXN or DXP card.

```
*DELETE SX.2,SX.8
      SUBROUTINE SRCDX
*CALL CM
C     THIS SRCDX USES THE SRC CARD TO SET PSC FOR EACH
C     DETECTOR.  IDET IS THE SEQUENCE NUMBER FOR THE
C     POINT DETECTOR CURRENTLY BEING CALCULATED.
C     THIS APPROACH SHOULD NOT BE USED WITH RING DETECTORS
```

```
C     OR WITH DXTRAN
      PSC=SRC(IDET)
      RETURN
      END
```

Figure 4.17

It is extremely important to note that the above cases apply only when the source is anisotropic with azimuthal symmetry. For the general case,

$$PSC = 2\pi p(\mu_o,\varphi_o).$$

The 2π factor must be applied by the user since MCNP assumes azimuthal symmetry and in effect divides the user-defined PSC by 2π. For a continuous $p(\mu,\varphi)$ function, PSC is calculated as above. In the case of a discrete probability density function,

$$PSC = 2\pi \cdot \overline{p(\mu_o,\varphi_o)} = \frac{2\pi(P_i - P_{i-1})}{(\mu_i - \mu_{i-1})(\varphi_i - \varphi_{i-1})}$$

$$= \frac{2\pi(P_i - P_{i-1})}{\Delta\mu_i \Delta\varphi_i}$$

where $\mu_{i-1} \le \mu_o < \mu_i$, $\varphi_{i-1} \le \varphi_o < \varphi_i$ and $\overline{p(\mu_o,\varphi_o)}$ is an average probability density function in the specified values of μ_o and φ_o and $P_i - P_{i-1}$ is the probability of selecting μ_o and φ_o in these intervals. For the case of N equally probable bins and n equally spaced $\Delta\varphi$'s each $2\pi/n$ wide,

$$PSC = \frac{n}{N}\frac{1}{\Delta\mu_i}$$

Another way to view this general case is by considering solid angles on the unit sphere. For an isotropic source, the probability $(P_i - P_{i-1})$ of being emitted into a specified solid angle is the ratio of the total solid angle (4π) to the specified solid angle $(\Delta\varphi\Delta\mu)$. For this

case, PSC≡0.5. Thus, for the general case (normed to PSC≡0.5 for an isotropic source)

$$PSC = \frac{(0.5)(P_i - P_{i-1})4\pi}{\Delta\mu_i \Delta\varphi_i} = \frac{2\pi(P_i - P_{i-1})}{\Delta\mu_i \Delta\varphi_i}$$

Note that PSC is greater than 0.5 if the specified solid angle $\Delta\mu\ \Delta\varphi_i$ is less than $(P_i - P_{i-1})4\pi$. This is the same as the previous general expression.

CAUTIONS:

The user is cautioned to be extremely careful when using his own subroutine SOURCE with either detectors or DXTRAN. This caution applies to the calculation of the direct contribution from the source to a point detector, point on a ring, or point on a DXTRAN sphere. Not only is there the calculation of the correct value of PSC for an anisotropic source, but there may also be problems with a biased source.

For example, if an isotropic source is biased to start only in a cone of a specified angle (e.g., ψ), then the starting weight of each particle should be WGT*(1 − cos ψ)/2 where WGT is the weight of the unbiased source (i.e., WGT is the expected weight from a total source). Note that the weight in SRCDX must be changed to the expected weight WGT to calculate the direct contribution to a point detector correctly if PSC is defined to be 0.5.

This example can be viewed in a different way. The probability density function for the above biased source is

$$p(\mu) = \frac{1}{1 - \cos\psi}, \quad \text{for } \cos\psi \le \mu \le 1$$

$$= 0 \qquad \text{for } -1 \le \mu < \cos\psi$$

Thus, PSC is this constant everywhere in the cone and zero elsewhere. Multiplying this PSC and biased starting weight gives

$$WGT*(1 - \cos\psi)*0.5/(1 - \cos\psi)$$

or WGT*0.5 which is the expected result for an isotropic source.

Another source type that requires caution is for a user supplied source that is energy-angle correlated. For example, assume a source has a Gaussian distribution in energy where the mean of the Gaussian is correlated in some manner with μ. In subroutine SRCDX, the μ_o to a point detector must be calculated and then the energy of the starting particle must be sampled from the Gaussian based on this μ_o. This must be done for each point detector in the problem, which guarantees that the direct source contribution to each detector will be from the proper energy spectrum. The original energy of the starting particle, as well as all of the other starting parameters, selected in subroutine SOURCE are automatically restored after the direct source contribution to detectors are made. Thus, the subroutine SOURCE is still sampled correctly.

CHAPTER 5

OUTPUT

This Chapter begins with a simple problem that illustrates geometry splitting and track population. The complete output of the problem is included, and parts of it are discussed. A second problem illustrates detectors and the detector diagnostics print.

An important section of the Chapter illustrates the event-log print and the debug print that can help you find errors if you modify the code or set up a geometry improperly.

The DBCN input card can be used for many purposes, but only the event-log feature is discussed in this Chapter in detail. If you make any modifications to MCNP, you are encouraged to read about the DBCN card; you may find entries 1, 2, 6, and 9 useful.

The Chapter is concluded with a short discussion of the KCODE output.

I. TEST1 PROBLEM

The input file for this problem is called TEST1 and can be obtained from the LIX file MCNPEX under /X6CODE on the Common File System.

TEST1 defines a disk of concrete 100 cm. thick and 75 cm. in radius. A 14.19 MeV neutron source is incident at a point in the center of a face of the disk and normal to it. The neutron and photon flux transmitted through the other face is determined. There is no energy cutoff in the problem, and the simple photon treatment is used. Thus capture for both neutrons and photons is treated implicitly by weight reduction (survival biasing).

The disk was first divided into sixteen slabs, each 6.25 cm. thick, as seen in Figure 5.1. The importance in each slab, or cell, was set to unity so there would be no geometry splitting. Thus, except for survival biasing, the first run was analog.

After five minutes of CDC-7600 computer time, the total transmitted neutron flux was 4.75×10^{-7} ± 10.1% neutrons/cm^{2}sec, and the photon flux was 9.10×10^{-7} ± 13.3% photons/cm^{2}sec. For 11386 starting neutrons, 11386*0.0112419=128 neutron tracks (determined with the FM1 card but see also Example 3 on page 194) and 114 photon tracks were transmitted and contributed to the flux tallies.

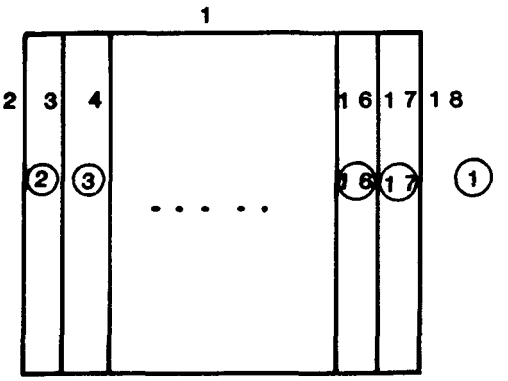


Figure 5.1

A second run was made with geometry splitting and Russian roulette. From the first run, wherever the neutron population dropped by a factor of two, the neutron importance was doubled. The photon importances were left at unity.

For the second run with 4314 starting neutrons, after five minutes of computer time, 2911 transmitted tracks produced a neutron flux of 4.59×10^{-7} ± 5.4%, and 802 photon tracks produced a flux of 8.42×10^{-7} ± 7.1%. Note that the neutron population is more or less constant (within about 25%) and so is the photon population. Comparing $1/\sigma^2 t$ for neutrons for the two runs, the first run was 3.5 times more expensive. (The importances for the second run could have been determined from the first run with only a minute of computer time, but five minutes were used here for illustration.) For a 200 cm. thick disk, the effect of splitting is dramatic. Try the thicker disk yourself by first using the good importances for the first 100 cm. and unity for the importances beyond 100 cm. On the second iteration change the beyond-100 importances in accordance with the population from 100 to 200 cm.

This second run could be optimized more in that many of the extraneous surfaces could be removed and the weight-cutoff game could be played a little better, but a second iteration would not yield a significant benefit over the first one - especially when it comes to setting importances.

II. TEST1 OUTPUT

Following is the complete output from TEST1 using the PRINT. option on the MCNP execution line. If a page has the symbol X on it, that page will not appear unless the PRINT. option or PRINT input card is used. If $\mathcal{N}$n (where n is an integer) appears before an item on a page, that item is explained in Note $\mathcal{N}$n in the text following the output pages.

```
mcnp   version 2    10/01/79                        u 10/15/79  18:55:27
**********************************************************************
        inp        =     test1    print.

        test1: 100 cm thick concrete disk with 15 splitting surfaces
        c
         1    0          1 : -2 : 18
         2    1 -2.2505  -1  -3   2
         3    1 -2.2505  -1  -4   3
         4    1 -2.2505  -1  -5   4
         5    1 -2.2505  -1  -6   5
         6    1 -2.2505  -1  -7   6
         7    1 -2.2505  -1  -8   7
         8    1 -2.2505  -1  -9   8
         9    1 -2.2505  -1  -10  9
        10    1 -2.2505  -1  -11  10
        11    1 -2.2505  -1  -12  11
        12    1 -2.2505  -1  -13  12
        13    1 -2.2505  -1  -14  13
        14    1 -2.2505  -1  -15  14
        15    1 -2.2505  -1  -16  15
        16    1 -2.2505  -1  -17  16
        17    1 -2.2505  -1  -18  17

         1    cy  75
         2    py  0
         3    py  6.25
         4    py  12.50
         5    py  18.75
         6    py  25.00
         7    py  31.25
         8    py  37.50
         9    py  43.75
        10    py  50.00
        11    py  56.25
        12    py  62.50
        13    py  68.75
        14    py  75.00
        15    py  81.25
        16    py  87.50
        17    py  93.75
        18    py  100.00

        mode  1
        c      the following is los alamos concrete
        m1    1001. -.004532
              8016  -.512597
              11023 -.011553
              12000 -.003866
              13027 -.035548
              14000 -.360364
              19000 -.014219
              20000 -.043546
              26000 -.013775
        src5  0 0 0 2 1 0 0 1 0 0 2
        serg  14.19  14.19
        sprob 0  1
        in    0 1 1 1 1 2 2 2 4 4 8 8 16 16 32 32 64
        ip    0 1 15r
        f1    18
        fm1   1 0
        fc1    this tally is the normalized tracks crossing the surface
        f2    18
```

```
60-        f11    18
61-        fm11    1 0
62-        fc11    this tally is the normalized tracks crossing the surface
63-        f12    18
64-        e0     .0001 .001 .01 .05 .1 .5 1 2 3 4 5
65-               6 7 8 9 10 11 12 13 14 15
66-        ergn   0  14.19
67-        ergp   0
68-        nps    20000
69-        ctme   5
70-
```

source = 5

M3 source coefficients

1	0.
2	0.
3	0.
4	2.0000e+00
5	1.0000e+00
6	0.
7	0.
8	1.0000e+00
9	0.
10	0.
11	2.0000e+00
12	0.

M4 source energy distribution

n	energy	cumulative probability	weight multiplier
1	1.4190e+01	0.	0.
2	1.4190e+01	1.0000e+00	1.0000e+00

M5 average energy using energy midpoints = 1.4190e+01 mev

M6 minimum source weight = 1.0000e+00

```
x       tally   1           this tally is the normalized tracks crossing the surface
        tally type 1    number of neutrons crossing a surface.

      problem surface numbers:   18
m7  warning.  energy bin limits adjusted for tally   1

     21 energy bins:
         0.0          to 1.00000e-04 mev
         1.00000e-04  to 1.00000e-03 mev
         1.00000e-03  to 1.00000e-02 mev
         1.00000e-02  to 5.00000e-02 mev
         5.00000e-02  to 1.00000e-01 mev
         1.00000e-01  to 5.00000e-01 mev
         5.00000e-01  to 1.00000e+00 mev
         1.00000e+00  to 2.00000e+00 mev
         2.00000e+00  to 3.00000e+00 mev
         3.00000e+00  to 4.00000e+00 mev
         4.00000e+00  to 5.00000e+00 mev
         5.00000e+00  to 6.00000e+00 mev
         6.00000e+00  to 7.00000e+00 mev
         7.00000e+00  to 8.00000e+00 mev
         8.00000e+00  to 9.00000e+00 mev
         9.00000e+00  to 1.00000e+01 mev
         1.00000e+01  to 1.10000e+01 mev
         1.10000e+01  to 1.20000e+01 mev
         1.20000e+01  to 1.30000e+01 mev
         1.30000e+01  to 1.40000e+01 mev
         1.40000e+01  to 1.41900e+01 mev

     1 time bin with upper limit = 1.00000+123 shakes

     1 cosine bin:  -1.0 to 1.00000e+00
```

```
x       tally   2
        tally type 2          neutron flux averaged over a surface.     units:     1/cm**2

     problem surface numbers:    18
   warning.  energy bin limits adjusted for tally   2

    21 energy bins:
         0.0           to 1.00000e-04 mev
         1.00000e-04   to 1.00000e-03 mev
         1.00000e-03   to 1.00000e-02 mev
         1.00000e-02   to 5.00000e-02 mev
         5.00000e-02   to 1.00000e-01 mev
         1.00000e-01   to 5.00000e-01 mev
         5.00000e-01   to 1.00000e+00 mev
         1.00000e+00   to 2.00000e+00 mev
         2.00000e+00   to 3.00000e+00 mev
         3.00000e+00   to 4.00000e+00 mev
         4.00000e+00   to 5.00000e+00 mev
         5.00000e+00   to 6.00000e+00 mev
         6.00000e+00   to 7.00000e+00 mev
         7.00000e+00   to 8.00000e+00 mev
         8.00000e+00   to 9.00000e+00 mev
         9.00000e+00   to 1.00000e+01 mev
         1.00000e+01   to 1.10000e+01 mev
         1.10000e+01   to 1.20000e+01 mev
         1.20000e+01   to 1.30000e+01 mev
         1.30000e+01   to 1.40000e+01 mev
         1.40000e+01   to 1.41900e+01 mev

     1 time bin with upper limit = 1.00000+123 shakes
```

```
x       tally  11          this tally is the normalized tracks crossing the surface
        tally type 1    number of  photons crossing a surface.

      problem surface numbers:   18
    warning.  energy bin limits adjusted for tally  11

     21 energy bins:
         0.0          to 1.00000e-04 mev
         1.00000e-04  to 1.00000e-03 mev
         1.00000e-03  to 1.00000e-02 mev
         1.00000e-02  to 5.00000e-02 mev
         5.00000e-02  to 1.00000e-01 mev
         1.00000e-01  to 5.00000e-01 mev
         5.00000e-01  to 1.00000e+00 mev
         1.00000e+00  to 2.00000e+00 mev
         2.00000e+00  to 3.00000e+00 mev
         3.00000e+00  to 4.00000e+00 mev
         4.00000e+00  to 5.00000e+00 mev
         5.00000e+00  to 6.00000e+00 mev
         6.00000e+00  to 7.00000e+00 mev
         7.00000e+00  to 8.00000e+00 mev
         8.00000e+00  to 9.00000e+00 mev
         9.00000e+00  to 1.00000e+01 mev
         1.00000e+01  to 1.10000e+01 mev
         1.10000e+01  to 1.20000e+01 mev
         1.20000e+01  to 1.30000e+01 mev
         1.30000e+01  to 1.40000e+01 mev
         1.40000e+01  to 1.41900e+01 mev

      1 time bin with upper limit = 1.00000+123 shakes

      1 cosine bin:  -1.0 to 1.00000e+00
```

```
x       tally  12
        tally type 2          photon flux averaged over a surface.    units:    1/cm**2

  problem surface numbers:   18
warning.  energy bin limits adjusted for tally  12

 21 energy bins:
      0.0          to 1.00000e-04 mev
      1.00000e-04  to 1.00000e-03 mev
      1.00000e-03  to 1.00000e-02 mev
      1.00000e-02  to 5.00000e-02 mev
      5.00000e-02  to 1.00000e-01 mev
      1.00000e-01  to 5.00000e-01 mev
      5.00000e-01  to 1.00000e+00 mev
      1.00000e+00  to 2.00000e+00 mev
      2.00000e+00  to 3.00000e+00 mev
      3.00000e+00  to 4.00000e+00 mev
      4.00000e+00  to 5.00000e+00 mev
      5.00000e+00  to 6.00000e+00 mev
      6.00000e+00  to 7.00000e+00 mev
      7.00000e+00  to 8.00000e+00 mev
      8.00000e+00  to 9.00000e+00 mev
      9.00000e+00  to 1.00000e+01 mev
      1.00000e+01  to 1.10000e+01 mev
      1.10000e+01  to 1.20000e+01 mev
      1.20000e+01  to 1.30000e+01 mev
      1.30000e+01  to 1.40000e+01 mev
      1.40000e+01  to 1.41900e+01 mev

  1 time bin with upper limit = 1.00000+123 shakes
```

M8 material composition

material number	component nuclide, atom fraction							
1	1001.00, .08477	8016.00, .60408	11023.00, .00947	12000.00, .00300				
	13027.00, .02483	14000.00, .24187	19000.00, .00685	20000.00, .02048				
	26000.00, .00465							

x cell volumes and masses

progr name	probl name	atom density	gram density	input volume	calculated volume	mass	pieces	reason volume not calculated
1	1	0.	0.	0.	0.	0.	0	infinite volume
2	2	7.18931e-02	2.25050e+00	0.	1.10447e+05	2.48560e+05	1	
3	3	7.18931e-02	2.25050e+00	0.	1.10447e+05	2.48560e+05	1	
4	4	7.18931e-02	2.25050e+00	0.	1.10447e+05	2.48560e+05	1	
5	5	7.18931e-02	2.25050e+00	0.	1.10447e+05	2.48560e+05	1	
6	6	7.18931e-02	2.25050e+00	0.	1.10447e+05	2.48560e+05	1	
7	7	7.18931e-02	2.25050e+00	0.	1.10447e+05	2.48560e+05	1	
8	8	7.18931e-02	2.25050e+00	0.	1.10447e+05	2.48560e+05	1	
9	9	7.18931e-02	2.25050e+00	0.	1.10447e+05	2.48560e+05	1	
10	10	7.18931e-02	2.25050e+00	0.	1.10447e+05	2.48560e+05	1	
11	11	7.18931e-02	2.25050e+00	0.	1.10447e+05	2.48560e+05	1	
12	12	7.18931e-02	2.25050e+00	0.	1.10447e+05	2.48560e+05	1	
13	13	7.18931e-02	2.25050e+00	0.	1.10447e+05	2.48560e+05	1	
14	14	7.18931e-02	2.25050e+00	0.	1.10447e+05	2.48560e+05	1	
15	15	7.18931e-02	2.25050e+00	0.	1.10447e+05	2.48560e+05	1	
16	16	7.18931e-02	2.25050e+00	0.	1.10447e+05	2.48560e+05	1	
17	17	7.18931e-02	2.25050e+00	0.	1.10447e+05	2.48560e+05	1	

N9

X surface areas

M10 progr name	probl name	input area	calculated area	reason area not calculated
1	1	0.	4.71239e+04	
2	2	0.	1.76715e+04	
3	3	0.	1.76715e+04	
4	4	0.	1.76715e+04	
5	5	0.	1.76715e+04	
6	6	0.	1.76715e+04	
7	7	0.	1.76715e+04	
8	8	0.	1.76715e+04	
9	9	0.	1.76715e+04	
10	10	0.	1.76715e+04	
11	11	0.	1.76715e+04	
12	12	0.	1.76715e+04	
13	13	0.	1.76715e+04	
14	14	0.	1.76715e+04	
15	15	0.	1.76715e+04	
16	16	0.	1.76715e+04	
17	17	0.	1.76715e+04	
18	18	0.	1.76715e+04	

M11 17 cells

progr name	probl name	mat	atom den	gram den	importance	photon wt	volume	mass
1	1	0	0.	0.	0.	-1.000e+00	1.000e+00	0.
2	2	1	7.189e-02	2.251e+00	1.000e+00	-1.000e+00	1.104e+05	2.486e+05
3	3	1	7.189e-02	2.251e+00	1.000e+00	-1.000e+00	1.104e+05	2.486e+05
4	4	1	7.189e-02	2.251e+00	1.000e+00	-1.000e+00	1.104e+05	2.486e+05
5	5	1	7.189e-02	2.251e+00	1.000e+00	-1.000e+00	1.104e+05	2.486e+05
6	6	1	7.189e-02	2.251e+00	2.000e+00	-1.000e+00	1.104e+05	2.486e+05
7	7	1	7.189e-02	2.251e+00	2.000e+00	-1.000e+00	1.104e+05	2.486e+05
8	8	1	7.189e-02	2.251e+00	2.000e+00	-1.000e+00	1.104e+05	2.486e+05
9	9	1	7.189e-02	2.251e+00	4.000e+00	-1.000e+00	1.104e+05	2.486e+05
10	10	1	7.189e-02	2.251e+00	4.000e+00	-1.000e+00	1.104e+05	2.486e+05
11	11	1	7.189e-02	2.251e+00	8.000e+00	-1.000e+00	1.104e+05	2.486e+05
12	12	1	7.189e-02	2.251e+00	8.000e+00	-1.000e+00	1.104e+05	2.486e+05
13	13	1	7.189e-02	2.251e+00	1.600e+01	-1.000e+00	1.104e+05	2.486e+05
14	14	1	7.189e-02	2.251e+00	1.600e+01	-1.000e+00	1.104e+05	2.486e+05
15	15	1	7.189e-02	2.251e+00	3.200e+01	-1.000e+00	1.104e+05	2.486e+05
16	16	1	7.189e-02	2.251e+00	3.200e+01	-1.000e+00	1.104e+05	2.486e+05
17	17	1	7.189e-02	2.251e+00	6.400e+01	-1.000e+00	1.104e+05	2.486e+05

x

cells

progr name	probl name	neutron importance	photon importance
1	1	0.	0.
2	2	1.000e+00	1.000e+00
3	3	1.000e+00	1.000e+00
4	4	1.000e+00	1.000e+00
5	5	1.000e+00	1.000e+00
6	6	2.000e+00	1.000e+00
7	7	2.000e+00	1.000e+00
8	8	2.000e+00	1.000e+00
9	9	4.000e+00	1.000e+00
10	10	4.000e+00	1.000e+00
11	11	8.000e+00	1.000e+00
12	12	8.000e+00	1.000e+00
13	13	1.600e+01	1.000e+00
14	14	1.600e+01	1.000e+00
15	15	3.200e+01	1.000e+00
16	16	3.200e+01	1.000e+00
17	17	6.400e+01	1.000e+00

x 18 surfaces

progr name	probl name	type	coefficients	M12
1	1	cy	5.6250000e+03	
2	2	py	0.	
3	3	py	6.2500000e+00	
4	4	py	1.2500000e+01	
5	5	py	1.8750000e+01	
6	6	py	2.5000000e+01	
7	7	py	3.1250000e+01	
8	8	py	3.7500000e+01	
9	9	py	4.3750000e+01	
10	10	py	5.0000000e+01	
11	11	py	5.6250000e+01	
12	12	py	6.2500000e+01	
13	13	py	6.8750000e+01	
14	14	py	7.5000000e+01	
15	15	py	8.1250000e+01	
16	16	py	8.7500000e+01	
17	17	py	9.3750000e+01	
18	18	py	1.0000000e+02	

x use of available storage space

quantity	symbol	amount used	limit
cells	mxa	17	175
surfaces	mxj	18	175
cell surfaces	nlja	53	1224
other-side cells	nlaj	0	2099
surface coefficients	nscf	18	874
source energies	mxs	2	51
tallies	ntal	4	40
tally descriptions	lpl	159	5125
detectors	ndetx	0	20
material data	mix	19	250
nuclides	mxe	9	40
nuclide pointers	mmat	9	120
thermal nuclides	isott	0	40
thermal times	mxt	0	7

********** initiation completed **********

4 warning messages so far

photon cross sections from file mcplib *<*<*<*<*<*<*<*<*<

element	table length
1	387
8	387
11	399
12	407
13	407
14	407
19	407
20	415
26	415
total	3631

```
      neutron cross sections

      nuclide     storage

     nuclides from file rmccs          rmccs   04/12/78
        1001.04       2115     h-1   endf/b iv (t404 rev.1) 10nov75 new gp format                    (  1269) 11 dec 75
                               endf/b gamma production on equal probable mesh
        8016.04      16372      o-16 endf/b-iv   new gamma production format                        (  1276) 14 oct 75
                               endf/b gamma production on equal probable mesh
       11023.01       5917     na-23    lll-howerton   1/73                                       (   516)  7 may 75
                                 na-23   gamma prod. xsec   (mat 1156)  28june72     endf/b - iii
       12000.02       3085     mg       endf/b-iii   (t302)  20feb74                               (  1014)  6 may 75
M13  warning.  12000.02 lacks gamma-ray production cross sections.
       13027.04      26401      al-27 endf/b-iv  new gamma production format                       (  1193) 14 oct 75
                               endf/b gamma production on equal probable mesh
       14000.02      18139     si       endf/b-iii   (t302)  20feb74                              (  1151)  6 may 75
                               gamma production data from xsec 14dec72.
       19000.01       6030     k        lll-howerton   1/73                                       (   524)  7 may 75
                                 k   gamma prod. xsec   (mat 1150)   15june72       endf/b - iii
       20000.10      17855     ca   endf/b-iv 1165 0k 14 sept 76   cut-off=1.0-5 ev 20 mev upper   (  1195) 28 sep 76
                               endf/b gamma production on equal probable mesh
       26000.11      45679     fe endf/b-iv t=300.0 24 february 1976  pointwise                    (  1192) 16 mar 76
                               endf/b gamma production on equal probable mesh

        total       141593

M14  cross sections outside the range from 0.          to 1.4190e+01 mev are expunged.
M15  warning.  energy cutoff below energy of some cross-section tables.


     apportionment of lcm2
      tallies                    687
      kcode source                 0
      bank                      2200
      variable common backup     413
      user file buffers            0
      photon cross sections     3631
      neutron cross sections  141593


     **********************************************************************
     dump no.  1 on file runtpe            nps =       0          ctm =   0.00

           6 warning messages so far
```

starting mcrun. lcm field length = 218137 = 0652031b cp0 = .06

test1: 100 cm thick concrete disk with 15 splitting surfaces

M16 nps	x	y	z	ia	ja	u	v	w	tme	wgt	erg
1	0.	0.	0.	2	2	0.	1.000e+00	0.	0.	1.000e+00	1.419e+01
2	0.	0.	0.	2	2	0.	1.000e+00	0.	0.	1.000e+00	1.419e+01
3	0.	0.	0.	2	2	0.	1.000e+00	0.	0.	1.000e+00	1.419e+01
4	0.	0.	0.	2	2	0.	1.000e+00	0.	0.	1.000e+00	1.419e+01
5	0.	0.	0.	2	2	0.	1.000e+00	0.	0.	1.000e+00	1.419e+01
6	0.	0.	0.	2	2	0.	1.000e+00	0.	0.	1.000e+00	1.419e+01
7	0.	0.	0.	2	2	0.	1.000e+00	0.	0.	1.000e+00	1.419e+01
8	0.	0.	0.	2	2	0.	1.000e+00	0.	0.	1.000e+00	1.419e+01
9	0.	0.	0.	2	2	0.	1.000e+00	0.	0.	1.000e+00	1.419e+01
10	0.	0.	0.	2	2	0.	1.000e+00	0.	0.	1.000e+00	1.419e+01
11	0.	0.	0.	2	2	0.	1.000e+00	0.	0.	1.000e+00	1.419e+01
12	0.	0.	0.	2	2	0.	1.000e+00	0.	0.	1.000e+00	1.419e+01
13	0.	0.	0.	2	2	0.	1.000e+00	0.	0.	1.000e+00	1.419e+01
14	0.	0.	0.	2	2	0.	1.000e+00	0.	0.	1.000e+00	1.419e+01
15	0.	0.	0.	2	2	0.	1.000e+00	0.	0.	1.000e+00	1.419e+01
16	0.	0.	0.	2	2	0.	1.000e+00	0.	0.	1.000e+00	1.419e+01
17	0.	0.	0.	2	2	0.	1.000e+00	0.	0.	1.000e+00	1.419e+01
18	0.	0.	0.	2	2	0.	1.000e+00	0.	0.	1.000e+00	1.419e+01
19	0.	0.	0.	2	2	0.	1.000e+00	0.	0.	1.000e+00	1.419e+01
20	0.	0.	0.	2	2	0.	1.000e+00	0.	0.	1.000e+00	1.419e+01
21	0.	0.	0.	2	2	0.	1.000e+00	0.	0.	1.000e+00	1.419e+01
22	0.	0.	0.	2	2	0.	1.000e+00	0.	0.	1.000e+00	1.419e+01
23	0.	0.	0.	2	2	0.	1.000e+00	0.	0.	1.000e+00	1.419e+01
24	0.	0.	0.	2	2	0.	1.000e+00	0.	0.	1.000e+00	1.419e+01
25	0.	0.	0.	2	2	0.	1.000e+00	0.	0.	1.000e+00	1.419e+01
26	0.	0.	0.	2	2	0.	1.000e+00	0.	0.	1.000e+00	1.419e+01
27	0.	0.	0.	2	2	0.	1.000e+00	0.	0.	1.000e+00	1.419e+01
28	0.	0.	0.	2	2	0.	1.000e+00	0.	0.	1.000e+00	1.419e+01
29	0.	0.	0.	2	2	0.	1.000e+00	0.	0.	1.000e+00	1.419e+01
30	0.	0.	0.	2	2	0.	1.000e+00	0.	0.	1.000e+00	1.419e+01
31	0.	0.	0.	2	2	0.	1.000e+00	0.	0.	1.000e+00	1.419e+01
32	0.	0.	0.	2	2	0.	1.000e+00	0.	0.	1.000e+00	1.419e+01
33	0.	0.	0.	2	2	0.	1.000e+00	0.	0.	1.000e+00	1.419e+01
34	0.	0.	0.	2	2	0.	1.000e+00	0.	0.	1.000e+00	1.419e+01
35	0.	0.	0.	2	2	0.	1.000e+00	0.	0.	1.000e+00	1.419e+01
36	0.	0.	0.	2	2	0.	1.000e+00	0.	0.	1.000e+00	1.419e+01
37	0.	0.	0.	2	2	0.	1.000e+00	0.	0.	1.000e+00	1.419e+01
38	0.	0.	0.	2	2	0.	1.000e+00	0.	0.	1.000e+00	1.419e+01
39	0.	0.	0.	2	2	0.	1.000e+00	0.	0.	1.000e+00	1.419e+01
40	0.	0.	0.	2	2	0.	1.000e+00	0.	0.	1.000e+00	1.419e+01
41	0.	0.	0.	2	2	0.	1.000e+00	0.	0.	1.000e+00	1.419e+01
42	0.	0.	0.	2	2	0.	1.000e+00	0.	0.	1.000e+00	1.419e+01
43	0.	0.	0.	2	2	0.	1.000e+00	0.	0.	1.000e+00	1.419e+01
44	0.	0.	0.	2	2	0.	1.000e+00	0.	0.	1.000e+00	1.419e+01
45	0.	0.	0.	2	2	0.	1.000e+00	0.	0.	1.000e+00	1.419e+01
46	0.	0.	0.	2	2	0.	1.000e+00	0.	0.	1.000e+00	1.419e+01
47	0.	0.	0.	2	2	0.	1.000e+00	0.	0.	1.000e+00	1.419e+01
48	0.	0.	0.	2	2	0.	1.000e+00	0.	0.	1.000e+00	1.419e+01
49	0.	0.	0.	2	2	0.	1.000e+00	0.	0.	1.000e+00	1.419e+01
50	0.	0.	0.	2	2	0.	1.000e+00	0.	0.	1.000e+00	1.419e+01

run terminated when it had used 5. minutes of computer time.

test1: 100 cm thick concrete disk with 15 splitting surfaces u 10/15/79 19:02:40

summary of neutron creation and loss

		tracks	weight (per source particle)	energy (per source particle)			tracks	weight (per source particle)	energy (per source particle)
	source	4314	1.0000e+00	1.4190e+01		escape	5735	3.4742e-01	9.6938e-01
						time cutoff	0	0.	0.
	fission	0	0.	0.		energy cutoff	0	0.	0.
	(n,xn)	47	5.5721e-03	5.4464e-03		all capture	0	6.6200e-01	5.0267e+00
	weight cutoff	0	7.2917e-02	1.0900e-0		weight cutoff	7906	7.6122e-02	9.7550e-02
	import. sampling	22960	3.2323e-01	2.2754e-01		import. sampling	13680	3.1617e-01	2.3018e-01
	forced collision	0	0.	0.		exp. transform	0	0.	0.
	energy splitting	0	0.	0.		scattering	0	0.	8.2082e+00
	dxtran	0	0.	0.		dxtran	0	0.	0.
	total	27321	1.4017e+00	1.4532e+01		total	27321	1.4017e+00	1.4532e+01

	predicted average source energy	1.4190e+01	average lifetime, shakes		cutoffs	
	neutron tracks per neutron started	6.3331e+00	escape	1.4868e+03	tco	1.0000+123
	neutron collisions per neutron started	1.0624e+02	capture	2.2861e+04	eco	0.
	total neutron collisions	458330	capture or escape	1.5504e+04	wc1	-5.0000e-01
	net multiplication	1.0056e+00 .0009	any termination	1.5335e+04	wc2	-2.5000e-01

summary of photon creation and loss

		tracks	weight (per source particle)	energy (per source particle)			tracks	weight (per source particle)	energy (per source particle)
	source	0	0.	0.		escape	2901	3.8545e-01	7.6946e-01
	from neutrons	15363	1.6816e+00	5.1356e+00		time cutoff	0	0.	0.
	fluorescence	0	0.	0.		energy cutoff	0	0.	0.
	pair production	0	1.0955e-01	-4.6701e-01		all capture	0	1.3830e+00	1.0847e-01
	weight cutoff	0	1.7607e-01	3.6894e-02		weight cutoff	12462	1.7877e-01	3.6247e-02
	import. sampling	0	0.	0.		import. sampling	0	0.	0.
	forced collision	0	0.	0.		exp. transform	0	0.	0.
	energy splitting	0	0.	0.		scattering	0	0.	3.7913e+00
	dxtran	0	0.	0.		dxtran	0	0.	0.
	total	15363	1.9473e+00	4.7055e+00		total	15363	1.9473e+00	4.7055e+00

	photon tracks per neutron started	3.5612e+00	average lifetime, shakes		cutoffs	
	photon collisions per neutron started	3.2550e+01	escape	1.0070e+04	tco	1.0000+123
	total photon collisions	140421	capture	1.4004e+04	eco	1.0000e-03
			capture or escape	1.3146e+04	wc1	-5.0000e-01
			any termination	1.3404e+04	wc2	-2.5000e-01

computer time so far in this run	5.07 minutes	total neutrons banked	23007
computer time in mcrun (4c0)	5.01 minutes	per source particle	5.3331e+00
source particles per minute	8.6149e+02	total photons banked	15362
lcm field length	218137 = 0652031b	per source particle	3.5610e+00
random numbers generated	5045788	maximum number ever in bank	16
last starting random number	4520076306426105b	bank overflows to disk	0

problem activity in each cell, neutrons only

cell progr	cell probl	tracks entering	population	collisions	collisions * weight (per history)	number weighted energy	flux weighted energy	average track weight (relative)	average track mfp (cm)
2	2	7190	4597	14586	2.4076e+00	2.2502e-03	7.3576e+00	7.6121e-01	6.6709e+00
3	3	8014	4128	20262	3.1101e+00	1.1698e-03	5.0395e+00	6.9428e-01	5.8199e+00
4	4	7857	3908	22274	3.2339e+00	8.0618e-04	3.8668e+00	6.4948e-01	5.3654e+00
5	5	7284	4104	22408	3.1708e+00	4.8575e-04	2.9045e+00	6.2193e-01	4.9228e+00
6	6	12496	6841	40400	2.7341e+00	3.1301e-04	2.2682e+00	5.9397e-01	4.5876e+00
7	7	9930	4933	32587	2.1609e+00	2.3892e-04	1.9338e+00	5.7712e-01	4.4267e+00
8	8	7569	4459	25032	1.6282e+00	1.8909e-04	1.6635e+00	5.6556e-01	4.2625e+00
9	9	11106	6824	37572	1.1947e+00	1.4455e-04	1.5074e+00	5.4967e-01	4.1559e+00
10	10	8101	5025	27559	8.4363e-01	1.2419e-04	1.3605e+00	5.2796e-01	4.0661e+00
11	11	11816	7250	40160	5.9502e-01	8.7972e-05	1.2541e+00	5.1250e-01	4.0118e+00
12	12	8370	5161	28512	4.1000e-01	1.0013e-04	1.1577e+00	4.9793e-01	3.9596e+00
13	13	11717	7222	40457	2.8442e-01	9.7455e-05	1.0851e+00	4.8590e-01	3.9109e+00
14	14	8021	4931	28387	1.9623e-01	9.0605e-05	9.7181e-01	4.7797e-01	3.8388e+00
15	15	10419	6585	35953	1.2145e-01	7.6163e-05	9.1786e-01	4.6725e-01	3.8294e+00
16	16	6315	4164	22133	7.4269e-02	8.2140e-05	8.7898e-01	4.6355e-01	3.8181e+00
17	17	6040	5231	20048	3.3397e-02	1.0206e-04	1.0343e+00	4.6142e-01	3.9946e+00
total		142245	85363	458330	2.2199e+01				
		$N29	$N30	$N31	$N32	$N33	$N34	$N35	$N36

problem activity in each cell, photons only

cell progr	cell probl	tracks entering	population	collisions	collisions * weight (per history)	number weighted energy	flux weighted energy	average track weight (relative)	average track mfp (cm)
2	2	1720	2781	7137	1.4652e+00	1.7524e+00	1.7524e+00	9.4709e-01	7.7099e+00
3	3	2893	3053	10063	1.9808e+00	1.5067e+00	1.5067e+00	9.2327e-01	6.9914e+00
4	4	3299	3269	10479	1.9884e+00	1.5040e+00	1.5040e+00	8.9579e-01	6.9636e+00
5	5	3642	3355	10692	1.8089e+00	1.4445e+00	1.4445e+00	7.9020e-01	6.8576e+00
6	6	3406	3510	11388	1.5339e+00	1.4491e+00	1.4491e+00	6.2584e-01	6.8599e+00
7	7	3369	3314	10415	1.2418e+00	1.4764e+00	1.4764e+00	5.4782e-01	6.9822e+00
8	8	3321	3078	8846	9.3347e-01	1.5188e+00	1.5188e+00	4.7790e-01	7.0513e+00
9	9	3071	3163	9771	7.7459e-01	1.3903e+00	1.3903e+00	3.6138e-01	6.7201e+00
10	10	3296	3072	9223	5.6921e-01	1.3940e+00	1.3940e+00	2.7954e-01	6.7376e+00
11	11	2954	3176	9462	4.2871e-01	1.4214e+00	1.4214e+00	2.0162e-01	6.7959e+00
12	12	3258	3091	9090	3.2620e-01	1.4402e+00	1.4402e+00	1.5738e-01	6.8450e+00
13	13	2894	3065	9118	2.3436e-01	1.4799e+00	1.4799e+00	1.1708e-01	6.9248e+00
14	14	2793	2663	7900	1.7572e-01	1.3355e+00	1.3355e+00	9.8807e-02	6.5816e+00
15	15	2242	2485	7475	1.1128e-01	1.3819e+00	1.3819e+00	6.7930e-02	6.8020e+00
16	16	1924	1909	5695	7.7641e-02	1.2889e+00	1.2889e+00	5.7959e-02	6.5166e+00
17	17	1130	1439	3667	4.3560e-02	1.2952e+00	1.2952e+00	4.8070e-02	6.5810e+00
total		45212	46421	140421	1.3694e+01				

N37

x
x38 neutron weight balance in each cell -- external events

cell progr	cell probl	source	entering	time cutoff	energy cutoff	exiting	total
2	2	1.0000e+00	4.2586e-01	0.	0.	-1.2929e+00	1.3292e-01
3	3	0.	1.3979e+00	0.	0.	-1.2890e+00	1.0885e-01
4	4	0.	1.2594e+00	0.	0.	-1.1618e+00	9.7657e-02
5	5	0.	1.0863e+00	0.	0.	-1.0151e+00	7.1141e-02
6	6	0.	8.9288e-01	0.	0.	-8.2847e-01	6.4406e-02
7	7	0.	6.8330e-01	0.	0.	-6.3325e-01	5.0048e-02
8	8	0.	5.0700e-01	0.	0.	-4.6911e-01	3.7894e-02
9	9	0.	3.6044e-01	0.	0.	-3.3363e-01	2.6803e-02
10	10	0.	2.5333e-01	0.	0.	-2.3280e-01	2.0528e-02
11	11	0.	1.7804e-01	0.	0.	-1.6366e-01	1.4381e-02
12	12	0.	1.2335e-01	0.	0.	-1.1261e-01	1.0740e-02
13	13	0.	8.3665e-02	0.	0.	-7.6709e-02	6.9551e-03
14	14	0.	5.5888e-02	0.	0.	-5.1367e-02	4.5209e-03
15	15	0.	3.5693e-02	0.	0.	-3.2632e-02	3.0604e-03
16	16	0.	2.1396e-02	0.	0.	-1.9553e-02	1.8426e-03
17	17	0.	1.0134e-02	0.	0.	-9.3063e-03	8.2779e-04
total		1.0000e+00	7.3745e+00	0.	0.	-7.7220e+00	6.5258e-01

x neutron weight balance in each cell -- physical events

cell progr	cell probl	fission	(n,xn)		capture	total
2	2	0.	2.1767e-03	0.	-1.3558e-01	-1.3340e-01
3	3	0.	1.7861e-03	0.	-1.1084e-01	-1.0905e-01
4	4	0.	7.6785e-04	0.	-9.6808e-02	-9.6040e-02
5	5	0.	2.0173e-04	0.	-7.8211e-02	-7.8009e-02
6	6	0.	2.7391e-04	0.	-6.3813e-02	-6.3539e-02
7	7	0.	9.5511e-05	0.	-4.9859e-02	-4.9764e-02
8	8	0.	1.6803e-04	0.	-3.7961e-02	-3.7793e-02
9	9	0.	0.	0.	-2.7152e-02	-2.7152e-02
10	10	0.	0.	0.	-1.9717e-02	-1.9717e-02
11	11	0.	4.5791e-05	0.	-1.4629e-02	-1.4583e-02
12	12	0.	3.4440e-05	0.	-9.9104e-03	-9.8759e-03
13	13	0.	1.3459e-05	0.	-7.0668e-03	-7.0533e-03
14	14	0.	3.9870e-06	0.	-4.7341e-03	-4.7301e-03
15	15	0.	3.6219e-06	0.	-3.0985e-03	-3.0949e-03
16	16	0.	0.	0.	-1.7934e-03	-1.7934e-03
17	17	0.	9.9264e-07	0.	-8.2145e-04	-8.2045e-04
total		0.	5.5721e-03	0.	-6.6200e-01	-6.5642e-01

x neutron weight balance in each cell — variance reduction events

cell progr	cell probl	weight cutoff	importance sampling	exponential transform	dxtran	total
2	2	4.8442e-04	0.	0.	0.	4.8442e-04
3	3	2.0273e-04	0.	0.	0.	2.0273e-04
4	4	-1.6166e-03	0.	0.	0.	-1.6166e-03
5	5	-1.7895e-03	8.6578e-03	0.	0.	6.8683e-03
6	6	-8.6723e-04	0.	0.	0.	-8.6723e-04
7	7	-2.8416e-04	0.	0.	0.	-2.8416e-04
8	8	3.9820e-04	-4.9875e-04	0.	0.	-1.0054e-04
9	9	3.4869e-04	0.	0.	0.	3.4869e-04
10	10	-1.6506e-04	-6.4612e-04	0.	0.	-8.1118e-04
11	11	2.0193e-04	0.	0.	0.	2.0193e-04
12	12	-1.3685e-04	-7.2717e-04	0.	0.	-8.6402e-04
13	13	9.8263e-05	0.	0.	0.	9.8263e-05
14	14	-4.3894e-05	2.5305e-04	0.	0.	2.0915e-04
15	15	3.4436e-05	0.	0.	0.	3.4436e-05
16	16	-6.3323e-05	1.4152e-05	0.	0.	-4.9170e-05
17	17	-7.3306e-06	0.	0.	0.	-7.3306e-06
total		-3.2052e-03	7.0530e-03	0.	0.	3.8477e-03

x photon weight balance in each cell — external events

cell progr	probl	source	entering	time cutoff	energy cutoff	exiting	total
2	2	0.	3.6239e-01	0.	0.	-5.9834e-01	-2.3595e-01
3	3	0.	6.0269e-01	0.	0.	-7.0571e-01	-1.0301e-01
4	4	0.	6.5032e-01	0.	0.	-7.1117e-01	-6.0854e-02
5	5	0.	6.1757e-01	0.	0.	-6.4839e-01	-3.0818e-02
6	6	0.	5.3475e-01	0.	0.	-5.4440e-01	-9.6531e-03
7	7	0.	4.3937e-01	0.	0.	-4.4764e-01	-8.2769e-03
8	8	0.	3.5420e-01	0.	0.	-3.5406e-01	1.4451e-04
9	9	0.	2.8472e-01	0.	0.	-2.8718e-01	1.7541e-02
10	10	0.	2.1305e-01	0.	0.	-2.0196e-01	1.1083e-02
11	11	0.	1.6061e-01	0.	0.	-1.5427e-01	6.3424e-03
12	12	0.	1.2516e-01	0.	0.	-1.1779e-01	7.3703e-03
13	13	0.	9.5111e-02	0.	0.	-8.9339e-02	5.7718e-03
14	14	0.	6.8937e-02	0.	0.	-6.2575e-02	6.3625e-03
15	15	0.	4.5818e-02	0.	0.	-4.2077e-02	3.7411e-03
16	16	0.	3.0056e-02	0.	0.	-2.6928e-02	3.1273e-03
17	17	0.	1.6919e-02	0.	0.	-1.5290e-02	1.6286e-03
total		0.	4.6017e+00	0.	0.	-4.9871e+00	-3.8545e-01

x photon weight balance in each cell — physical events

cell progr	cell probl	from neutrons	pair production	fluorescence	capture	total
2	2	3.4494e-01	1.5395e-02	0.	-1.2329e-01	2.3704e-01
3	3	2.8353e-01	1.4295e-02	0.	-1.9757e-01	1.0025e-01
4	4	2.4766e-01	1.6846e-02	0.	-2.0107e-01	6.3433e-02
5	5	2.0637e-01	1.4572e-02	0.	-1.8684e-01	3.4108e-02
6	6	1.5809e-01	1.1736e-02	0.	-1.6193e-01	7.9005e-03
7	7	1.2689e-01	8.9093e-03	0.	-1.2865e-01	7.1433e-03
8	8	8.6299e-02	8.0551e-03	0.	-9.3427e-02	9.2710e-04
9	9	6.4025e-02	4.5167e-03	0.	-8.5138e-02	-1.6567e-02
10	10	4.3353e-02	4.0740e-03	0.	-5.9270e-02	-1.1843e-02
11	11	3.5235e-02	2.9982e-03	0.	-4.5383e-02	-7.1491e-03
12	12	2.4365e-02	2.3676e-03	0.	-3.3529e-02	-6.7967e-03
13	13	1.7001e-02	2.4463e-03	0.	-2.4638e-02	-5.1910e-03
14	14	1.0584e-02	1.2891e-03	0.	-1.8592e-02	-6.7190e-03
15	15	7.1285e-03	1.0212e-03	0.	-1.1716e-02	-3.5662e-03
16	16	4.2881e-03	5.0671e-04	0.	-7.9830e-03	-3.1882e-03
17	17	1.8865e-03	4.8879e-04	0.	-4.0016e-03	-1.6263e-03
total		1.6616e+00	1.0955e-01	0.	-1.3830e+00	3.8815e-01

photon weight balance in each cell -- variance reduction events

cell progr	probl	weight cutoff	importance sampling	exponential transform	dxtran	total
2	2	-4.5599e-02	0.	0.	0.	-4.5599e-02
3	3	-8.2539e-02	0.	0.	0.	-8.2539e-02
4	4	-8.6492e-02	0.	0.	0.	-8.6492e-02
5	5	-1.0065e-01	0.	0.	0.	-1.0065e-01
6	6	-3.9508e-02	0.	0.	0.	-3.9508e-02
7	7	-3.8041e-02	0.	0.	0.	-3.8041e-02
8	8	-3.8624e-02	0.	0.	0.	-3.8624e-02
9	9	-1.4651e-02	0.	0.	0.	-1.4651e-02
10	10	-2.1493e-02	0.	0.	0.	-2.1493e-02
11	11	-7.7701e-03	0.	0.	0.	-7.7701e-03
12	12	-8.4549e-03	0.	0.	0.	-8.4549e-03
13	13	-2.6671e-03	0.	0.	0.	-2.6671e-03
14	14	-3.8159e-03	0.	0.	0.	-3.8159e-03
15	15	-1.1021e-03	0.	0.	0.	-1.1021e-03
16	16	-1.0981e-03	0.	0.	0.	-1.0981e-03
17	17	-2.3403e-04	0.	0.	0.	-2.3403e-04
total		-4.9274e-01	0.	0.	0.	-4.9274e-01

X neutron activity of each nuclide in each cell

N39 cell progr	probl	nuclides	atom fraction	collisions	collisions * weight (per source	weight lost to capture particle)
2	2	1001.04	8.4765e-02	2898	4.3043e-01	2.9949e-03
		8016.04	6.0408e-01	7780	1.3039e+00	5.2363e-02
		11023.01	9.4727e-03	170	2.8282e-02	1.3595e-03
		12000.02	2.9977e-03	55	8.5516e-03	5.7039e-04
		13027.04	2.4835e-02	345	5.9118e-02	4.1404e-03
		14000.02	2.4187e-01	2827	4.9130e-01	6.2927e-02
		19000.01	6.8546e-03	94	1.5958e-02	3.1266e-03
		20000.10	2.0480e-02	282	4.8209e-02	6.1561e-03
		26000.11	4.6497e-03	137	2.1878e-02	1.9430e-03
3	3	1001.04	8.4765e-02	4823	7.0306e-01	5.7691e-03
		8016.04	6.0408e-01	10410	1.6195e+00	3.8027e-02
		11023.01	9.4727e-03	279	4.2146e-02	1.3449e-03
		12000.02	2.9977e-03	72	1.1214e-02	7.6563e-04
		13027.04	2.4835e-02	412	6.5663e-02	2.9150e-03
		14000.02	2.4187e-01	3620	5.7130e-01	4.8744e-02
		19000.01	6.8546e-03	115	1.6572e-02	3.8489e-03
		20000.10	2.0480e-02	351	5.5387e-02	6.8042e-03
		26000.11	4.6497e-03	180	2.5249e-02	2.6211e-03
4	4	1001.04	8.4765e-02	5662	7.9831e-01	7.2149e-03
		8016.04	6.0408e-01	11323	1.6586e+00	2.7132e-02
		11023.01	9.4727e-03	324	4.5928e-02	1.7955e-03
		12000.02	2.9977e-03	63	9.6476e-03	4.1505e-04
		13027.04	2.4835e-02	404	6.0592e-02	2.6585e-03
		14000.02	2.4187e-01	3793	5.6020e-01	4.2514e-02
		19000.01	6.8546e-03	143	1.9561e-02	4.8482e-03
		20000.10	2.0480e-02	362	5.2691e-02	6.0853e-03
		26000.11	4.6497e-03	200	2.8388e-02	4.1451e-03
5	5	1001.04	8.4765e-02	6022	8.3792e-01	8.6680e-03
		8016.04	6.0408e-01	11357	1.6185e+00	1.8947e-02
		11023.01	9.4727e-03	267	3.8848e-02	1.1187e-03
		12000.02	2.9977e-03	51	7.3696e-03	2.7784e-04
		13027.04	2.4835e-02	392	5.6402e-02	2.9366e-03
		14000.02	2.4187e-01	3564	5.0777e-01	3.2177e-02
		19000.01	6.8546e-03	130	1.7321e-02	4.7267e-03
		20000.10	2.0480e-02	394	5.4297e-02	5.9953e-03
		26000.11	4.6497e-03	231	3.2373e-02	3.3631e-03
6	6	1001.04	8.4765e-02	11230	7.4912e-01	8.6889e-03
		8016.04	6.0408e-01	20219	1.3781e+00	1.1819e-02
		11023.01	9.4727e-03	580	3.9544e-02	1.6143e-03
		12000.02	2.9977e-03	117	7.6685e-03	3.6625e-04
		13027.04	2.4835e-02	670	4.5643e-02	2.4403e-03
		14000.02	2.4187e-01	6294	4.2982e-01	2.5650e-02
		19000.01	6.8546e-03	214	1.3853e-02	3.9189e-03
		20000.10	2.0480e-02	639	4.2355e-02	4.6825e-03
		26000.11	4.6497e-03	437	2.8049e-02	4.6329e-03
7	7	1001.04	8.4765e-02	9219	6.0781e-01	7.7788e-03
		8016.04	6.0408e-01	16077	1.0756e+00	7.4688e-03
		11023.01	9.4727e-03	429	2.8386e-02	1.3128e-03
		12000.02	2.9977e-03	97	6.1893e-03	1.2029e-04
		13027.04	2.4835e-02	499	3.2428e-02	1.7717e-03
		14000.02	2.4187e-01	5179	3.4089e-01	1.9059e-02
		19000.01	6.8546e-03	253	1.5609e-02	4.9692e-03

		20000.10	2.0480e-02	481	3.1641e-02	3.9369e-03
		26000.11	4.6497e-03	353	2.2329e-02	3.4417e-03
8	8	1001.04	8.4765e-02	7287	4.7095e-01	6.8092e-03
		8016.04	6.0408e-01	12300	8.0484e-01	4.7878e-03
		11023.01	9.4727e-03	370	2.4448e-02	1.2483e-03
		12000.02	2.9977e-03	69	4.4290e-03	1.0048e-04
		13027.04	2.4835e-02	364	2.3682e-02	1.8083e-03
		14000.02	2.4187e-01	3852	2.5101e-01	1.3993e-02
		19000.01	6.8546e-03	154	9.3695e-03	3.4750e-03
		20000.10	2.0480e-02	376	2.3243e-02	2.9606e-03
		26000.11	4.6497e-03	260	1.6224e-02	2.7785e-03
9	9	1001.04	8.4765e-02	11165	3.5464e-01	5.3213e-03
		8016.04	6.0408e-01	18482	5.9137e-01	2.7909e-03
		11023.01	9.4727e-03	573	1.8383e-02	8.4403e-04
		12000.02	2.9977e-03	105	3.4309e-03	4.4924e-05
		13027.04	2.4835e-02	486	1.4897e-02	1.1260e-03
		14000.02	2.4187e-01	5517	1.7408e-01	1.0071e-02
		19000.01	6.8546e-03	236	6.8556e-03	2.3862e-03
		20000.10	2.0480e-02	590	1.8484e-02	2.2669e-03
		26000.11	4.6497e-03	418	1.2589e-02	2.3009e-03
10	10	1001.04	8.4765e-02	8307	2.5529e-01	3.8409e-03
		8016.04	6.0408e-01	13403	4.1212e-01	1.7902e-03
		11023.01	9.4727e-03	378	1.1689e-02	7.1771e-04
		12000.02	2.9977e-03	102	3.2534e-03	1.0220e-04
		13027.04	2.4835e-02	387	1.1320e-02	8.5658e-04
		14000.02	2.4187e-01	4097	1.2384e-01	7.4803e-03
		19000.01	6.8546e-03	188	5.3383e-03	1.6827e-03
		20000.10	2.0480e-02	390	1.1803e-02	1.6396e-03
		26000.11	4.6497e-03	307	8.9714e-03	1.6069e-03
11	11	1001.04	8.4765e-02	12169	1.8032e-01	2.9954e-03
		8016.04	6.0408e-01	19467	2.9061e-01	1.1038e-03
		11023.01	9.4727e-03	584	8.8154e-03	4.6321e-04
		12000.02	2.9977e-03	124	1.8665e-03	3.9844e-05
		13027.04	2.4835e-02	567	8.1015e-03	6.7953e-04
		14000.02	2.4187e-01	5936	8.6847e-02	5.2849e-03
		19000.01	6.8546e-03	302	4.1032e-03	1.5981e-03
		20000.10	2.0480e-02	543	7.6455e-03	1.1047e-03
		26000.11	4.6497e-03	468	6.7159e-03	1.3596e-03
12	12	1001.04	8.4765e-02	8636	1.2459e-01	2.0715e-03
		8016.04	6.0408e-01	13991	2.0177e-01	6.8090e-04
		11023.01	9.4727e-03	394	5.6723e-03	3.0288e-04
		12000.02	2.9977e-03	69	9.1811e-04	1.9353e-05
		13027.04	2.4835e-02	422	5.9779e-03	4.4903e-04
		14000.02	2.4187e-01	4091	5.8320e-02	3.7126e-03
		19000.01	6.8546e-03	175	2.4949e-03	9.4558e-04
		20000.10	2.0480e-02	395	5.5496e-03	7.4083e-04
		26000.11	4.6497e-03	339	4.7102e-03	9.8794e-04
13	13	1001.04	8.4765e-02	12316	8.6687e-02	1.4819e-03
		8016.04	6.0408e-01	19710	1.3900e-01	4.6916e-04
		11023.01	9.4727e-03	525	3.6864e-03	1.9591e-04
		12000.02	2.9977e-03	103	7.0990e-04	2.5970e-05
		13027.04	2.4835e-02	611	4.2874e-03	3.7885e-04
		14000.02	2.4187e-01	5859	4.1029e-02	2.5843e-03
		19000.01	6.8546e-03	263	1.7188e-03	6.5072e-04
		20000.10	2.0480e-02	600	4.1028e-03	5.7998e-04
		26000.11	4.6497e-03	470	3.1928e-03	6.9993e-04

14	14	1001.04	8.4765e-02	8715	6.0250e-02	1.0411e-03
		8016.04	6.0408e-01	13868	9.5846e-02	2.9816e-04
		11023.01	9.4727e-03	353	2.5302e-03	1.4354e-04
		12000.02	2.9977e-03	79	5.2420e-04	1.0629e-05
		13027.04	2.4835e-02	395	2.7297e-03	1.7561e-04
		14000.02	2.4187e-01	4053	2.8155e-02	1.7433e-03
		19000.01	6.8546e-03	171	1.0809e-03	4.4646e-04
		20000.10	2.0480e-02	425	2.8415e-03	4.0061e-04
		26000.11	4.6497e-03	330	2.2689e-03	4.7467e-04
15	15	1001.04	8.4765e-02	11164	3.7653e-02	6.9898e-04
		8016.04	6.0408e-01	17376	5.8924e-02	1.5352e-04
		11023.01	9.4727e-03	519	1.7623e-03	1.2686e-04
		12000.02	2.9977e-03	96	3.4799e-04	5.4812e-06
		13027.04	2.4835e-02	519	1.7393e-03	1.7362e-04
		14000.02	2.4187e-01	5128	1.7207e-02	1.1096e-03
		19000.01	6.8546e-03	212	6.8972e-04	2.8494e-04
		20000.10	2.0480e-02	517	1.7442e-03	2.4979e-04
		26000.11	4.6497e-03	422	1.3778e-03	2.9571e-04
16	16	1001.04	8.4765e-02	6674	2.2455e-02	3.8877e-04
		8016.04	6.0408e-01	10806	3.6363e-02	8.8038e-05
		11023.01	9.4727e-03	297	9.8027e-04	6.5488e-05
		12000.02	2.9977e-03	69	2.3746e-04	5.0731e-06
		13027.04	2.4835e-02	338	1.1113e-03	1.0072e-04
		14000.02	2.4187e-01	3206	1.0718e-02	6.7565e-04
		19000.01	6.8546e-03	137	4.3737e-04	1.7345e-04
		20000.10	2.0480e-02	339	1.1013e-03	1.2416e-04
		26000.11	4.6497e-03	267	8.6531e-04	1.7210e-04
17	17	1001.04	8.4765e-02	6038	1.0093e-02	1.6742e-04
		8016.04	6.0408e-01	9754	1.6276e-02	4.8502e-05
		11023.01	9.4727e-03	287	4.9560e-04	2.5442e-05
		12000.02	2.9977e-03	53	8.2541e-05	1.9357e-06
		13027.04	2.4835e-02	313	5.2802e-04	3.7745e-05
		14000.02	2.4187e-01	2942	4.8353e-03	3.0694e-04
		19000.01	6.8546e-03	142	2.2840e-04	8.7006e-05
		20000.10	2.0480e-02	303	4.9858e-04	7.2339e-05
		26000.11	4.6497e-03	216	3.5936e-04	7.4122e-05
total				458330	2.2199e+01	6.6200e-01

x photon activity of each nuclide in each cell

cell progr	cell probl	nuclides	atom fraction	collisions	collisions * weight (per source	weight lost to capture particle)
2	2	1001.04	8.4765e-02	62	1.3612e-02	0.
		8016.04	6.0408e-01	3332	7.0293e-01	1.0578e-02
		11023.01	9.4727e-03	64	1.3517e-02	5.3594e-04
		12000.02	2.9977e-03	32	6.1977e-03	5.1488e-04
		13027.04	2.4835e-02	246	5.1569e-02	4.3445e-03
		14000.02	2.4187e-01	2631	5.3329e-01	5.9320e-02
		19000.01	6.8546e-03	138	2.6266e-02	7.4047e-03
		20000.10	2.0480e-02	454	8.7449e-02	2.4792e-02
		26000.11	4.6497e-03	178	3.0344e-02	1.5800e-02
3	3	1001.04	8.4765e-02	71	1.4240e-02	0.
		8016.04	6.0408e-01	4638	9.4142e-01	1.7485e-02
		11023.01	9.4727e-03	99	1.9381e-02	1.1612e-03
		12000.02	2.9977e-03	32	6.4860e-03	4.7841e-04
		13027.04	2.4835e-02	348	6.8185e-02	7.3915e-03
		14000.02	2.4187e-01	3691	7.2295e-01	9.1519e-02
		19000.01	6.8546e-03	194	3.5014e-02	9.8969e-03
		20000.10	2.0480e-02	654	1.1844e-01	4.0127e-02
		26000.11	4.6497e-03	336	5.4687e-02	2.9514e-02
4	4	1001.04	8.4765e-02	70	1.2629e-02	0.
		8016.04	6.0408e-01	4692	9.2025e-01	1.6406e-02
		11023.01	9.4727e-03	108	1.9930e-02	1.5203e-03
		12000.02	2.9977e-03	38	7.0514e-03	4.0783e-04
		13027.04	2.4835e-02	367	7.1268e-02	6.5503e-03
		14000.02	2.4187e-01	4027	7.5707e-01	9.8705e-02
		19000.01	6.8546e-03	218	3.6891e-02	1.1678e-02
		20000.10	2.0480e-02	656	1.1261e-01	3.8600e-02
		26000.11	4.6497e-03	303	5.0699e-02	2.7206e-02
5	5	1001.04	8.4765e-02	76	1.3043e-02	0.
		8016.04	6.0408e-01	4958	8.6372e-01	1.5639e-02
		11023.01	9.4727e-03	107	1.8400e-02	1.4832e-03
		12000.02	2.9977e-03	38	6.3882e-03	5.4267e-04
		13027.04	2.4835e-02	328	5.4304e-02	6.5486e-03
		14000.02	2.4187e-01	3924	6.6033e-01	8.5354e-02
		19000.01	6.8546e-03	228	3.5387e-02	1.0866e-02
		20000.10	2.0480e-02	708	1.0954e-01	3.9422e-02
		26000.11	4.6497e-03	325	4.7815e-02	2.6984e-02
6	6	1001.04	8.4765e-02	76	1.0323e-02	0.
		8016.04	6.0408e-01	5233	7.3601e-01	1.3956e-02
		11023.01	9.4727e-03	123	1.6994e-02	8.8943e-04
		12000.02	2.9977e-03	34	4.2657e-03	3.5760e-04
		13027.04	2.4835e-02	390	5.1767e-02	5.6763e-03
		14000.02	2.4187e-01	4250	5.6294e-01	8.0008e-02
		19000.01	6.8546e-03	210	2.6391e-02	8.3224e-03
		20000.10	2.0480e-02	724	8.6264e-02	3.1372e-02
		26000.11	4.6497e-03	348	3.8959e-02	2.1348e-02
7	7	1001.04	8.4765e-02	83	1.0978e-02	0.
		8016.04	6.0408e-01	4734	5.8882e-01	1.1107e-02
		11023.01	9.4727e-03	110	1.3827e-02	7.4284e-04
		12000.02	2.9977e-03	32	3.7525e-03	2.4772e-04
		13027.04	2.4835e-02	351	3.9335e-02	4.6589e-03
		14000.02	2.4187e-01	3898	4.5427e-01	6.1051e-02
		19000.01	6.8546e-03	181	1.9675e-02	6.1005e-03

		20000.10	2.0480e-02	709	7.8317e-02	2.5897e-02
		26000.11	4.6497e-03	317	3.2789e-02	1.8848e-02
8	8	1001.04	8.4765e-02	62	6.0480e-03	0.
		8016.04	6.0408e-01	4063	4.4087e-01	7.9467e-03
		11023.01	9.4727e-03	92	1.0053e-02	4.8407e-04
		12000.02	2.9977e-03	41	4.7194e-03	2.9129e-04
		13027.04	2.4835e-02	318	3.2398e-02	3.5775e-03
		14000.02	2.4187e-01	3308	3.4692e-01	4.4358e-02
		19000.01	6.8546e-03	151	1.4148e-02	4.3739e-03
		20000.10	2.0480e-02	536	5.0688e-02	1.7854e-02
		26000.11	4.6497e-03	275	2.7621e-02	1.4541e-02
9	9	1001.04	8.4765e-02	79	6.8494e-03	0.
		8016.04	6.0408e-01	4392	3.5835e-01	7.2955e-03
		11023.01	9.4727e-03	133	9.8579e-03	5.5576e-04
		12000.02	2.9977e-03	41	3.4644e-03	1.8441e-04
		13027.04	2.4835e-02	344	2.5733e-02	3.2249e-03
		14000.02	2.4187e-01	3673	2.9079e-01	4.1654e-02
		19000.01	6.8546e-03	181	1.3944e-02	4.1904e-03
		20000.10	2.0480e-02	617	4.4767e-02	1.6069e-02
		26000.11	4.6497e-03	311	2.0834e-02	1.1964e-02
10	10	1001.04	8.4765e-02	89	5.6699e-03	0.
		8016.04	6.0408e-01	4229	2.7187e-01	5.1732e-03
		11023.01	9.4727e-03	98	5.7107e-03	3.2285e-04
		12000.02	2.9977e-03	24	1.3107e-03	8.7077e-05
		13027.04	2.4835e-02	304	1.8449e-02	2.0496e-03
		14000.02	2.4187e-01	3427	2.0669e-01	2.7889e-02
		19000.01	6.8546e-03	188	1.0743e-02	2.8701e-03
		20000.10	2.0480e-02	585	3.3220e-02	1.1861e-02
		26000.11	4.6497e-03	279	1.5546e-02	9.0188e-03
11	11	1001.04	8.4765e-02	68	2.7433e-03	0.
		8016.04	6.0408e-01	4381	2.0624e-01	3.9385e-03
		11023.01	9.4727e-03	108	4.6821e-03	4.4250e-04
		12000.02	2.9977e-03	38	1.8286e-03	1.2596e-04
		13027.04	2.4835e-02	322	1.5713e-02	1.6340e-03
		14000.02	2.4187e-01	3475	1.5347e-01	2.1923e-02
		19000.01	6.8546e-03	183	7.7215e-03	2.3506e-03
		20000.10	2.0480e-02	599	2.4873e-02	8.7149e-03
		26000.11	4.6497e-03	290	1.1438e-02	6.2533e-03
12	12	1001.04	8.4765e-02	74	2.9768e-03	0.
		8016.04	6.0408e-01	4181	1.5242e-01	2.8114e-03
		11023.01	9.4727e-03	97	3.2202e-03	1.6949e-04
		12000.02	2.9977e-03	41	1.2757e-03	7.5103e-05
		13027.04	2.4835e-02	330	1.1824e-02	1.1812e-03
		14000.02	2.4187e-01	3349	1.2172e-01	1.6284e-02
		19000.01	6.8546e-03	168	5.2971e-03	1.6848e-03
		20000.10	2.0480e-02	563	1.8244e-02	5.9277e-03
		26000.11	4.6497e-03	289	9.2253e-03	5.3950e-03
13	13	1001.04	8.4765e-02	49	1.0527e-03	0.
		8016.04	6.0408e-01	4142	1.1402e-01	2.3462e-03
		11023.01	9.4727e-03	100	2.7701e-03	1.4666e-04
		12000.02	2.9977e-03	28	8.6434e-04	1.5479e-05
		13027.04	2.4835e-02	329	8.0800e-03	8.5673e-04
		14000.02	2.4187e-01	3399	8.3594e-02	1.1278e-02
		19000.01	6.8546e-03	170	4.3922e-03	1.4562e-03
		20000.10	2.0480e-02	620	1.3246e-02	4.8777e-03
		26000.11	4.6497e-03	281	6.3390e-03	3.6607e-03

14	14	1001.04	8.4765e-02	49	7.4621e-04	0.
		8016.04	6.0408e-01	3608	8.2080e-02	1.5798e-03
		11023.01	9.4727e-03	73	1.2531e-03	8.1972e-05
		12000.02	2.9977e-03	37	9.9976e-04	4.2253e-05
		13027.04	2.4835e-02	253	6.6691e-03	6.6385e-04
		14000.02	2.4187e-01	2995	6.5522e-02	6.8156e-03
		19000.01	6.8546e-03	137	2.4493e-03	7.7945e-04
		20000.10	2.0480e-02	504	1.0533e-02	3.4686e-03
		26000.11	4.6497e-03	244	5.4659e-03	3.1608e-03
15	15	1001.04	8.4765e-02	51	7.1355e-04	0.
		8016.04	6.0408e-01	3372	5.1288e-02	9.6617e-04
		11023.01	9.4727e-03	94	1.1145e-03	6.1076e-05
		12000.02	2.9977e-03	32	4.5438e-04	2.7668e-05
		13027.04	2.4835e-02	237	3.4495e-03	3.2865e-04
		14000.02	2.4187e-01	2837	4.2498e-02	5.6496e-03
		19000.01	6.8546e-03	121	1.4223e-03	4.2684e-04
		20000.10	2.0480e-02	514	7.9780e-03	2.8585e-03
		26000.11	4.6497e-03	217	2.3619e-03	1.3974e-03
16	16	1001.04	8.4765e-02	46	6.5946e-04	0.
		8016.04	6.0408e-01	2558	3.6834e-02	8.2919e-04
		11023.01	9.4727e-03	65	6.5085e-04	4.0063e-05
		12000.02	2.9977e-03	28	3.3393e-04	4.2069e-05
		13027.04	2.4835e-02	189	1.9560e-03	2.5201e-04
		14000.02	2.4187e-01	2133	2.8326e-02	3.5230e-03
		19000.01	6.8546e-03	112	2.1535e-03	6.6208e-04
		20000.10	2.0480e-02	406	4.5165e-03	1.4927e-03
		26000.11	4.6497e-03	160	2.2109e-03	1.1419e-03
17	17	1001.04	8.4765e-02	25	1.9758e-04	0.
		8016.04	6.0408e-01	1644	2.0536e-02	3.1623e-04
		11023.01	9.4727e-03	51	5.5451e-04	2.3897e-05
		12000.02	2.9977e-03	14	3.4307e-04	9.0426e-06
		13027.04	2.4835e-02	130	1.3488e-03	8.5311e-05
		14000.02	2.4187e-01	1399	1.6381e-02	2.1408e-03
		19000.01	6.8546e-03	71	9.6178e-04	2.1573e-04
		20000.10	2.0480e-02	219	1.9742e-03	5.8231e-04
		26000.11	4.6497e-03	114	1.2625e-03	6.2828e-04
total				140421	1.3694e+01	1.3830e+00

1140 summary of photons produced in neutron collisions

cell number	wt per neutron	mev/gram-neutron	number of photons
1	0.	0.	0
2	3.44937e-01	4.34791e-06	1478
3	2.83525e-01	3.49422e-06	1207
4	2.47661e-01	3.15441e-06	1051
5	2.06374e-01	2.47425e-06	870
6	1.58093e-01	1.92277e-06	1335
7	1.26888e-01	1.55759e-06	1049
8	8.62986e-02	1.12263e-06	722
9	6.40246e-02	7.85565e-07	1069
10	4.33529e-02	5.21555e-07	722
11	3.52354e-02	4.49683e-07	1167
12	2.43648e-02	3.11790e-07	805
13	1.70006e-02	2.14912e-07	1140
14	1.05843e-02	1.30864e-07	717
15	7.12847e-03	9.19390e-08	959
16	4.28810e-03	5.67026e-08	572
17	1.88648e-03	2.46365e-08	500

total number of photons produced = 15363

energy interval	normed wt frequency
9.0000e+00	0.
8.0000e+00	5.00968e-03
7.0000e+00	5.47167e-02
6.0000e+00	1.20002e-01
5.0000e+00	3.33657e-02
4.0000e+00	8.69070e-02
3.0000e+00	1.40899e-01
2.0000e+00	1.46052e-01
1.0000e+00	2.38702e-01
5.0000e-01	9.71970e-02
1.0000e-01	5.62746e-02
1.0000e-02	1.86210e-02
0.	2.25294e-03

total photon energy (mev) started	mev started per neutron	started wt per neutron
2.21551e+04	5.13562e+00	1.66164e+00

M41 tally 1 nps = 4314 this tally is the normalized tracks crossing the surface
tally type 1 number of neutrons crossing a surface.
this tally has been modified by an fm card.

surface 18 area = 1.76715e+04

cosine bin from -1.00000e+00 to 1.00000e+00

time: 1.0000+123

energy		
1.0000e-04	2.56838e-01	.0579
1.0000e-03	4.58971e-02	.0924
1.0000e-02	3.82476e-02	.0971
5.0000e-02	2.73528e-02	.1164
1.0000e-01	1.64580e-02	.1379
5.0000e-01	3.66249e-02	.1005
1.0000e+00	3.29161e-02	.1103
2.0000e+00	4.52017e-02	.1128
3.0000e+00	6.25869e-02	.0959
4.0000e+00	1.50672e-02	.1628
5.0000e+00	1.18220e-02	.1780
6.0000e+00	1.34446e-02	.1883
7.0000e+00	1.76171e-02	.1826
8.0000e+00	5.09968e-03	.2723
9.0000e+00	3.47705e-03	.3709
1.0000e+01	3.70885e-03	.3058
1.1000e+01	3.47705e-03	.3194
1.2000e+01	6.95410e-03	.2256
1.3000e+01	6.49050e-03	.2668
1.4000e+01	1.92397e-02	.1632
1.4190e+01	6.25869e-03	.2692
total	6.74780e-01	.0493

tally 2 nps = 4314
M42 tally type 2 neutron flux averaged over a surface. units: 1/cm**2

surface 18 area = 1.76715e+04

time: 1.0000+123

energy		
1.0000e-04	1.78409e-07	.0656
1.0000e-03	3.11400e-08	.1107
1.0000e-02	2.79014e-08	.1469
5.0000e-02	1.82818e-08	.1248
1.0000e-01	1.12864e-08	.1505
5.0000e-01	2.84262e-08	.1345
1.0000e+00	2.61130e-08	.1606
2.0000e+00	3.53750e-08	.1448
3.0000e+00	4.20821e-08	.1077
4.0000e+00	8.82495e-09	.1721
5.0000e+00	6.46442e-09	.1870
6.0000e+00	7.62481e-09	.1935
7.0000e+00	9.46353e-09	.1928
8.0000e+00	3.06597e-09	.2855
9.0000e+00	1.68369e-09	.4008
1.0000e+01	1.83508e-09	.3013
1.1000e+01	1.53221e-09	.3257
1.2000e+01	3.57696e-09	.2329
1.3000e+01	2.91948e-09	.2639
1.4000e+01	9.45384e-09	.1726
1.4190e+01	3.30445e-09	.2838
total	4.58764e-07	.0536

tally 11 nps = 4314 this tally is the normalized tracks crossing the surface
tally type 1 number of photons crossing a surface.
this tally has been modified by an fm card.

surface 18 area = 1.76715e+04

cosine bin from -1.00000e+00 to 1.00000e+00

time: 1.0000+123

energy		
1.0000e-04	0.	0.0000
1.0000e-03	0.	0.0000
1.0000e-02	0.	0.0000
5.0000e-02	6.95410e-04	.5771
1.0000e-01	1.46036e-02	.1366
5.0000e-01	5.21558e-02	.0771
1.0000e+00	2.43394e-02	.1028
2.0000e+00	2.87436e-02	.1028
3.0000e+00	1.69217e-02	.1239
4.0000e+00	1.76171e-02	.1211
5.0000e+00	1.22856e-02	.1441
6.0000e+00	4.17246e-03	.2352
7.0000e+00	7.64951e-03	.1734
8.0000e+00	6.25869e-03	.1918
9.0000e+00	4.63607e-04	.7069
1.0000e+01	0.	0.0000
1.1000e+01	0.	0.0000
1.2000e+01	0.	0.0000
1.3000e+01	0.	0.0000
1.4000e+01	0.	0.0000
1.4190e+01	0.	0.0000
total	1.85906e-01	.0532

```
 tally  12        nps =     4314
 tally type 2           photon flux averaged over a surface.    units:    1/cm**2

surface    18                  area = 1.76715e+04

  time:           1.0000+123
   energy
 1.0000e-04   0.           0.0000
 1.0000e-03   0.           0.0000
 1.0000e-02   0.           0.0000
 5.0000e-02   9.81753e-10  .5958
 1.0000e-01   5.27782e-08  .3739
 5.0000e-01   2.47792e-07  .1082
 1.0000e+00   1.55726e-07  .2019
 2.0000e+00   1.52664e-07  .1542
 3.0000e+00   5.50725e-08  .1481
 4.0000e+00   6.02470e-08  .1860
 5.0000e+00   4.63645e-08  .2206
 6.0000e+00   2.34863e-08  .3300
 7.0000e+00   2.45720e-08  .2116
 8.0000e+00   2.17471e-08  .2364
 9.0000e+00   7.95720e-10  .7809
 1.0000e+01   0.           0.0000
 1.1000e+01   0.           0.0000
 1.2000e+01   0.           0.0000
 1.3000e+01   0.           0.0000
 1.4000e+01   0.           0.0000
 1.4190e+01   0.           0.0000
      total   8.42228e-07  .0712

 ************************************************************************
 dump no.  2 on file runtpe          nps =    4314          ctm =   5.01

        6 warning messages so far
```

Notes:

*N*1: This is the MCNP execution line that you entered.

*N*2: The numbers in this first column are sequential line numbers for the input file. They will be useful if you make any changes to the file with TRIX AC.

*N*3: The source coefficients listed here are read from the SRC card into the SRC array.

*N*4: The source energy distribution is defined by the SERG and SPROB input cards. You can and should verify your intended distribution with this table.

*N*5: Predicted average energy is calculated to provide a possible check on how well the source was sampled. One of the most important considerations in a Monte Carlo calculation is adequate sampling of the source. You can have a problem with an extreme amount of geometry splitting such that, for example, for fifty source particles taken from some distribution you get several hundred contributions to a tally. The variance may be small, but the results should be questioned since the fifty initial particles probably did not sample the source distribution well.

One check (others may be needed) on source sampling may be done by comparing the predicted average energy with the actual energy started that appears on the Creation and Loss summary page. If your source uses linear interpolation between energy bounds (as the MCNP standard sources do with the SERG and SPROB cards), these numbers should be in close agreement (in exact agreement in the limit of a large number of particles). However, this check is only a first step; it tells you nothing about directional sampling and is worthless for a monoenergetic source.

*N*6: The minimum source weight is the value the weight cutoffs are made relative to if they are entered as negative numbers on the CUTN or CUTP cards.

*N*7: The warning is because the upper energy limit on the E0 card of 15 MeV is higher than the maximum energy specified on the ERGN card.

*N*8: The fraction of each nuclide in a material is printed here as atom fraction even if you entered mass fraction.

*N*9: A disjointed Godfrey cell is composed of a number of individual *pieces* put together with the union operator. In addition, improperly defined cells not even using the union operator can result and be composed of more than one piece (for example, a surface is unknowingly extented and forms a cell). If a cell is composed of more than one piece, a warning message is given, and you should verify that the number of pieces is correct or not.

*N*10: This is one of two tables where the relation between surface problem and program numbers is given. The other comes in three pages. Note that neither is printed by default.

*N*11: This is a convenient table to check cell input items against. If you know the mass or volume of a geometry or parts of it, you can compare that with what MCNP calculated as a way to verify the correctness of your geometry.

*N*12: These are the surface coefficients used by the code and are not necessarily the ones on the surface input cards.

*N*13: Any cross-section evaluations called for in a Mode 1 problem not having gamma-ray production cross sections result in a warning. You may check other evaluations of the same nuclide to see what is available.

*N*14: Any cross sections outside the energy range specified on the ERGN and CUTN cards are deleted by EXPUNG.

*N*15: The standard lower energy limit is 10^{-11} MeV, but this problem had a zero energy cutoff. Exceptions to this 10^{-11} standard are listed in Appendix F. Neutrons with energy below the lower limit of a cross-section set are transported with the cross section at the lower limit.

*N*16: The parameters describing the first fifty source particles are printed out. It is wise to look at this page to make sure your source is correct.

$\mathcal{N}$17: This is the main summary page of the problem. It is a balance sheet with the left side showing how weight and energy were created and the right side showing all the ways weight and energy were lost.

$\mathcal{N}$18: The creation from weight cutoff represents winning the weight cutoff Russian roulette game with the track weight being raised to WC1.

$\mathcal{N}$19: Importance sampling on the creation side of the table represents winning the Russian roulette game when crossing the surface into a cell of lower importance.

$\mathcal{N}$20: Since analog capture was not allowed in this problem, no tracks were lost to capture but weight was through survival biasing, or implicit capture. The energy lost to capture is from the energy of the particle times the lost weight to capture.

$\mathcal{N}$21: Loss to weight cutoff comes from losing the weight cutoff Russian roulette game. With perfect sampling, the weight and energy lost here should equal the weight gained in item 15. There is a slight imbalance here.

$\mathcal{N}$22: Loss to importance sampling results from losing the Russian roulette game when crossing a surface into a cell of lower importance. The loss here should equal the gain in item 16, and the agreement is pretty good in this problem.

$\mathcal{N}$23: Scattering represents internal energy loss from collisions that is converted into heating.

$\mathcal{N}$24: The average lifetime of a neutron is extremely problem dependent and does not necessarily represent anything physical in the context here. The categories are the average time it took a neutron to escape from the geometry, to be captured, the weighted average of capture or escape lifetimes, or average lifetime to any terminal event such as weight cutoff, importance sampling, or energy cutoff in addition to capture and escape. These lifetimes are relative to zero time which is not necessarily the time a neutron is started from the source.

For a Mode 1 problem, the photon lifetime is relative also to zero time and not the time when the photon was produced. Thus the photon lifetime may be for some problems effectively the mean time to

creation since a photon's lifetime is so much shorter than a neutron's.

These numbers may be of use in setting a time cutoff or in getting a better feel for what is happening in the problem.

*N*25: The net multiplication of the system is discussed in detail on page 106. The second number is the relative error of the multiplication corresponding to one standard deviation. In this problem, the multiplication [which comes from (n,xn) reactionsis 1.0056 ± 0.09%.

*N*26: The pair production entry may seem to be incorrect since there is a negative energy on the creation side of the table. It is correct as is. Pair production represents a weight creation (a net gain of one) but an associated energy loss. A 5-MeV photon undergoing pair production results in one photon of energy 0.511 MeV but of twice the weight of the photon entering the event with a resultant loss of 3.98 MeV of energy that is deposited by the electron-positron pair slowing down.

*N*27: Ideally one photon should be produced per starting neutron. Although 3.57 is a little high here, it is not excessive. The number can be controlled with the PWT card.

*N*28: The random number starting the last history before this printout is listed here. This can be useful in conjunction with debugging a problem using the PRDMP card and/or the features of the DBCN card. Once a troublesome history is isolated, it can be re-run using this number for KRNT on the DBCN card.

*N*29: Tracks entering a cell refers to tracks crossing a surface to enter a cell plus any source particles. However, if a track leaves a cell and later re-enters that same cell, it is counted again.

*N*30: Population in a cell is the number of tracks entering a cell plus source particles, but it does not include re-entrant tracks. A comparison of this and item 25 will give an indication of the amount of backscattering in the problem.

A good rule of thumb for setting geometry splitting surfaces is to keep the population constant. Be careful in using these numbers

for that purpose, however, because they are not normalized according to the volume of the cell.

$\mathcal{N}$31: The number of collisions in a cell is important for a detector tally or anything involving collision rate. A lack of collisions may indicate a need to force them. This quantity is not normalized by cell volume.

$\mathcal{N}$32: The collisions times the weight of the particles having them is an indication of how important the collisions were.

$\mathcal{N}$33: The next four items are determined in the HSTORY subroutine after the distance D to the next collision or surface is determined. The time DT to traverse this distance is determined from DT=D/VEL where VEL is the velocity of the particle. Furthermore, the flux Φ is equal to the number density n(E) times the velocity.

The energy averaged over the number density of particles is determined by

$$\frac{\int n(E)\cdot E\ dE}{\int n(E)dE}$$

or

$$\frac{\sum (WGT*DT*ERG)}{\sum (WGT*DT)}$$

$\mathcal{N}$34: The energy averaged over the flux density is

$$\frac{\int \Phi(E)\cdot E\ dE}{\int \Phi(E)dE}$$

or

$$\frac{\sum (WGT*D*ERG)}{\sum (WGT*D)}$$

It is very difficult, and perhaps meaningless, to determine an average energy because a large spectrum involving several orders of magnitude is frequently involved leading to the problem of representing this by one number. That's why it has been calculated by the two methods of items 29 and 30. If the number-averaged energy is significantly lower than the flux-averaged energy (as is true in this

problem), it indicates a large number of low-energy particles. As the energy cutoff in this problem is raised, these two average energies come into closer agreement.

*N*35: The average track weight has been determined by averaging over pathlength D as in item 30. However, in many problems examined the average weight is reasonably insensitive to the method of calculation. For example, the track weight averaged over collisions may be calculated by multiplying the collisions*weight figure times the number of histories and dividing by the number of collisions. For cell 17, this is 5.10×10^{-2} as compared to 4.79×10^{-2} averaging over pathlength.

*N*36: The average mean free path of a track in a cell is determined by again averaging over pathlength D. The average mean free path is not as insensitive to the method of averaging as is the average weight, but it is good enough to a first approximation. However, averaging over pathlength is probably the most appropriate way. Even so, the purpose of the averages in items 29 through 32 is not to provide a rigorous type tally but to give you a better idea (and hopefully understanding) of what is going on in your problem.

*N*37: For photons, the number-weighted energy and flux-weighted energy are equal because a photon has a constant velocity regardless of energy.

*N*38: The next three tables for each type particle show all possible ways a particle's weight may be changed in each cell. In addition to telling you what is happening to the particle and where, this information can be useful in debugging a problem.

Note that the weight entering cell 17 is 1.02×10^{-2} whereas in the previous Problem Activity table the average track weight in cell 17 is 4.79×10^{-2}. This apparent discrepancy is resolved by realizing that the average weight in the Problem Activity table is for a track while it is for a history in the Weight Balance tables. If the average track weight is multiplied by the tracks entering cell 17 (1148) and then divided by the number of source particles (4327), the two weights are in close agreement.

*N*39: The activity of each nuclide per cell can tell you very quickly how important various nuclides are to the problem - such as trace

elements. If a nuclide is relatively unimportant, you may consider changing cross-section evaluations to an evaluation with fewer words.

$\mathcal{N}$40: This table is produced only for Mode 1, and it gives you an idea of how many photons were produced in each cell and the energy spectrum of the photons averaged over the problem. Since photons are produced only at neutron collisions, there is a correlation between the number of collisions in a cell, the PWT card, and this table.

$\mathcal{N}$41: The first tally is a result of the F1 and FM1 cards and like all tallies (unless changed by TALLYX) is normalized to be per starting particle. Multipyling the number of histories 4314 times the total 6.7478×10^{-1} says that 2911 neutron tracks crossed surface 18. Each track contributes unity to this modified tally rather than the weight of the track.

Tally 1 is the specific tally related to the F1 general type tally of current across a surface. The actual tally may be called F1 (as it is here) or incremented by 20 such as F21 for neutrons.

$\mathcal{N}$42: This tally of flux across surface 18 says that between 0 and 1×10^{-4} MeV, the flux is $1.73 \times 10^{-7} \pm 16.8\%$ within one standard deviation. The flux integrated over all energy is $4.59 \times 10^{-7} \pm 5.4\%$.

$\mathcal{N}$43: Note that with this initial run there are two dumps on the RUNTPE. The first one came at the end of the XACT overlay. A continut-run will pick up from this second dump (unless you specify m to be otherwise with the C. m option) and then add a third dump to the RUNTPE when it finishes.

CTM on this same line is the computer time in minutes.

III. SHELL PROBLEM

This simple problem, called SHELL in MCNPEX, is a shell of concrete 390 cm. in outer radius and 360 cm. in inner radius. There is a 14.19 MeV point isotropic neutron source at the center. Gamma-ray production is ignored (i.e., this is a Mode 0 problem) the energy cutoff is 12.0 MeV, and the object is to calculate the flux transmitted through the sphere. The flux is calculated by both an F2 surface flux tally and a ring detector

around the shell. Both DD and PRDMP input cards are used because of the detector.

The problem ran 100000 source neutrons in 2.49 minutes of computer time. The F2 surface flux tally is 7.12×10^{-8} ± 0.9% and the ring detector flux is 6.47×10^{-8} ± 8.2%. The specified radius R_o of the detector's imaginary sphere in this run was −0.50.

Running the same problem without the ring detector and only the F2 tally took 1.38 minutes for 100000 histories, so the detector doubled the running time. Doing a continue-run for a total of ten minutes of computer time, 404723 histories were run with the F2 flux being 7.07×10^{-8} ± 0.4% and the ring detector flux being 6.73×10^{-8} ± 3.8%. At 260000 histories, the detector flux was 7.08×10^{-8} ± 5.1%.

An analog detector on the outer surface on the y-axis produced a flux of 4.79×10^{-8} ± 22.6%. The OMCFE detector in the same location yielded a flux of 1.34×10^{-7} ± 55.2%. The analog detector took 2.29 minutes of time for 100000 histories and the OMCFE detector took 2.97 minutes (R_o for the analog detector was −0.5 and −2.5 for the OMCFE.) No conclusions about analog versus OMCFE detectors should be drawn from this. When any detector has an error greater than 10%, believe the result to only within a factor of a few; greater than 25%, within an order of magnitude; greater than 50%, don't believe it at all.

Severely biasing the source toward the detector location to get more collisions in the vicinity helped. The source biasing parameters were 80% into a cone of ν=.985 which is extreme and in general not to be done. The high energy cutoff made the severe biasing more tolerable, however. The analog result was 6.26×10^{-8} ± 7.6%, and the OMCFE result was 6.31×10^{-8} ± 6.0%. Changing R_o to 5 and 25 for the analog and OMCFE detectors, respectively, the analog flux was 6.45×10^{-8} ± 7.0% and 5.64×10^{-8} ± 4.4% for the OMCFE in about the same running time.

Using DXTRAN around the analog detector with radii of 5 and 15 for the two spheres resulted in a flux of 6.96×10^{-8} ± 2.5% (still with extreme source biasing and R_o=5). The computer time was 5.57 minutes.

The point of all of this is beware of detectors. They are very useful for some problems but inappropriate for others – such as this one. Always use the PRDMP and DD cards with detectors. If the variance is not stable and decreasing as a function of the number of histories run or there are relatively few collisions in the vicinity of a detector, be suspicious.

There is another to this story: always take advantage of symmetry in problems. A true three-dimensional problem with no symmetry probably requires a point detector, two-dimensional symmetry is best done with a

ring detector, and a one-dimensional problem can probably be done most efficiently with a simple F2 surface tally.

Other ways to calculate a flux in a small region in this problem would be to do an F4 cell-flux tally in a small sphere in the region of interest or do an F2 surface-flux tally across a segmented surface in the region of interest. Using DXTRAN around the F4 or segmented F2 tallies would probably produce good results.

The SHELL problem uses the PRDMP and DD input cards. Looking at the ring detector flux and error at each 10000 histories, we see

HISTORY	FLUX $x10^{-8}$	% ERROR
10000	5.10	11.4
20000	5.18	11.0
30000	5.15	8.5
40000	6.46	17.7
50000	6.20	14.9
60000	6.37	12.6
70000	6.37	10.9
80000	6.35	9.8
90000	6.45	8.9
100000	6.47	8.2

Even though the error is not jumping around after 40000 histories, it doesn't mean it won't later. In fact, looking at the results to 400000 histories, there are several oscillations but no large jumps.

Note the perturbation between 30000 and 40000 histories. The second entry of 100 on the DD card says to print a diagnostic print for scores to the detector that are at least 100 times larger than the average of the first 200 scores to the detector. Using this we should be able to pinpoint the history that produced this perturbation after 30000 histories. History 36575 is the culprit.

Following is some of the printout after the 30000^{th} history. It includes the normal Tally 5 printout, the summary of contributions to the detector from the DD card, and the DD card diagnostic print for large contributions.

The direct contribution of $1.61x10^{-8}$ from the source to the detector can be verified by a simple hand calculation. From the Problem Activity summary page near the end of the printout, the average mean free path for a track in cell 2 is 8.625 cm. Since the distance from the source to

detector is 390 cm., the thickness of cell 2 is 30 cm., and cell 1 is a void, the direct contribution is $e^{-30/8.625}/4\pi 390^2$ which is 1.61×10^{-8}. The direct contribution is a part of the total flux of 5.15×10^{-8}.

It is good practice to do a similar hand calculation whenever a user-provided SOURCE subroutine and a detector are used to make sure the direct contribution is correct. Remember that if the source is not isotropic, you will more than likely have to provide another subroutine, SRCDX.

```
 tally   5        nps =    30000
 tally type 5          neutron flux at a detector.  units:    1/cm**2

ring detector symmetric about y-axis located at y = 0.          with radius = 3.90000e+02

  time:           1.0000e+05
   energy
 1.0000e+02   5.15006e-08  .0849

ring detector symmetric about y-axis located at y = 0.          with radius = 3.90000e+02     direct

  time:           1.0000e+05
   energy
 1.0000e+02   1.60995e-08 -.0000
```

mean free paths	accumulative fraction of number of tallies	tally	accumulative fraction of total tally
1.	.00045	1.02565e-08	.19915
2.	.00277	9.89825e-09	.39135
3.	.00913	9.59735e-09	.57770
4.	.61283	2.09342e-08	.98419
5.	.63673	5.98248e-10	.99580
6.	.66485	1.50200e-10	.99872
7.	.69249	4.54331e-11	.99960
8.	.71915	1.41579e-11	.99988
>8.	1.00000	6.32067e-12	1.00000

```
number of tallies     50825
average tally   3.03988e-08
largest tally   7.96033e-05
```

det	t	wgt	psc	amfp	ddetx	radius	erg	cell	nps	nch	ijk	an
1	7.7347e-06	7.6705e-03	4.8559e+00	1.4895e+00	1.3146e+01	-5.0000e-01	1.2596e+01	2	30798	4	5003255746531425	
det	t	wgt	psc	amfp	ddetx	radius	erg	cell	nps	nch	ijk	an
1	6.2596e-06	7.6070e-03	2.5244e-01	5.9479e-01	5.1899e+00	-5.0000e-01	1.2685e+01	2	30917	1	4214601340161571	
det	t	wgt	psc	amfp	ddetx	radius	erg	cell	nps	nch	ijk	an
1	2.3915e-05	1.5190e-02	7.7948e+00	1.5219e+00	1.3116e+01	-5.0000e-01	1.4188e+01	2	31723	1	1253223254707521	
det	t	wgt	psc	amfp	ddetx	radius	erg	cell	nps	nch	ijk	an
1	4.4026e-06	1.7846e-02	1.8250e+00	1.7074e+00	1.4611e+01	-5.0000e-01	1.3940e+01	2	32515	1	4625201130017461	
det	t	wgt	psc	amfp	ddetx	radius	erg	cell	nps	nch	ijk	an
1	1.8491e-05	7.5978e-03	7.8352e-01	6.2132e-01	5.2468e+00	-5.0000e-01	1.3814e+01	2	32877	1	7057554217675631	
det	t	wgt	psc	amfp	ddetx	radius	erg	cell	nps	nch	ijk	an
1	5.1912e-06	2.0717e-02	9.5468e+00	2.5389e+00	2.1881e+01	-5.0000e-01	1.4179e+01	2	32900	1	0532562107025715	
det	t	wgt	psc	amfp	ddetx	radius	erg	cell	nps	nch	ijk	an
1	4.3659e-06	1.3210e-02	5.5539e+00	2.1039e+00	1.8062e+01	-5.0000e-01	1.3962e+01	2	33256	3	1033566626447335	
det	t	wgt	psc	amfp	ddetx	radius	erg	cell	nps	nch	ijk	an
1	1.3864e-05	1.0924e-02	6.4186e+00	1.5307e+00	1.3197e+01	-5.0000e-01	1.4096e+01	2	33446	2	0170575145663565	
det	t	wgt	psc	amfp	ddetx	radius	erg	cell	nps	nch	ijk	an
1	1.4183e-05	1.0583e-02	5.5394e+00	1.4447e+00	1.2456e+01	-5.0000e-01	1.4095e+01	2	33623	2	3116042471712101	

det	t	wgt	psc	amfp	ddetx	radius	erg	cell	nps	nch	ijk	an
1	5.1470e-06	2.4065e-03	1.0604e-01	3.8629e-01	3.3345e+00	-5.0000e-01	1.2054e+01	2	33623	3	3116042471712101	
det	t	wgt	psc	amfp	ddetx	radius	erg	cell	nps	nch	ijk	an
1	7.2140e-06	5.6960e-03	6.5901e-01	7.2448e-01	6.3348e+00	-5.0000e-01	1.2318e+01	2	33880	3	7764663346520235	
det	t	wgt	psc	amfp	ddetx	radius	erg	cell	nps	nch	ijk	an
1	2.3057e-05	1.5398e-02	7.7942e+00	1.5436e+00	1.3303e+01	-5.0000e-01	1.4187e+01	2	34573	1	4117037204262031	
det	t	wgt	psc	amfp	ddetx	radius	erg	cell	nps	nch	ijk	an
1	5.2978e-06	9.8626e-03	2.9726e+00	1.5849e+00	1.3436e+01	-5.0000e-01	1.3782e+01	2	34964	3	4024271014340015	
det	t	wgt	psc	amfp	ddetx	radius	erg	cell	nps	nch	ijk	an
1	3.9576e-05	7.7202e-03	2.5952e-01	4.1654e-01	3.6333e+00	-5.0000e-01	1.3367e+01	2	34970	1	1645632603301505	
det	t	wgt	psc	amfp	ddetx	radius	erg	cell	nps	nch	ijk	an
1	3.4856e-06	4.2899e-02	4.0945e+00	2.6908e+00	2.3323e+01	-5.0000e-01	1.2732e+01	2	35386	2	3415423504756705	
det	t	wgt	psc	amfp	ddetx	radius	erg	cell	nps	nch	ijk	an
1	7.9177e-06	1.2679e-02	2.1725e+00	1.3890e+00	1.1750e+01	-5.0000e-01	1.3863e+01	2	35647	2	2251607230561141	
det	t	wgt	psc	amfp	ddetx	radius	erg	cell	nps	nch	ijk	an
1	6.6028e-06	2.3309e-02	7.1550e+00	2.3136e+00	1.9940e+01	-5.0000e-01	1.4173e+01	2	36278	1	2745501241354665	
det	t	wgt	psc	amfp	ddetx	radius	erg	cell	nps	nch	ijk	an
1	2.2285e-05	2.5149e-03	4.3712e-01	3.5873e-01	3.0889e+00	-5.0000e-01	1.2770e+01	2	36515	2	4532523147723661	
det	t	wgt	psc	amfp	ddetx	radius	erg	cell	nps	nch	ijk	an
1	4.2877e-04	3.5787e-03	5.8652e+00	4.1266e-01	3.5396e+00	-5.0000e-01	1.3605e+01	2	36575	2	2163361440211341	
det	t	wgt	psc	amfp	ddetx	radius	erg	cell	nps	nch	ijk	an
1	4.1577e-06	1.8399e-02	7.3398e+00	2.4501e+00	2.1120e+01	-5.0000e-01	1.4128e+01	2	37086	2	4500550132071525	
det	t	wgt	psc	amfp	ddetx	radius	erg	cell	nps	nch	ijk	an
1	2.8399e-05	9.6692e-03	2.5161e+00	8.7462e-01	7.5404e+00	-5.0000e-01	1.3996e+01	2	38464	1	5262645336745075	
det	t	wgt	psc	amfp	ddetx	radius	erg	cell	nps	nch	ijk	an
1	4.6714e-06	1.0851e-02	3.5060e-01	8.5292e-01	7.4321e+00	-5.0000e-01	1.3314e+01	2	38464	2	5262645336745075	
det	t	wgt	psc	amfp	ddetx	radius	erg	cell	nps	nch	ijk	an
1	2.0431e-05	1.2486e-02	6.2722e+00	1.4269e+00	1.2101e+01	-5.0000e-01	1.3883e+01	2	39079	2	7235527657541201	
det	t	wgt	psc	amfp	ddetx	radius	erg	cell	nps	nch	ijk	an
1	3.1151e-05	8.5879e-03	7.3385e+00	1.1776e+00	9.9586e+00	-5.0000e-01	1.3793e+01	2	39079	3	7235527657541201	
det	t	wgt	psc	amfp	ddetx	radius	erg	cell	nps	nch	ijk	an
1	6.6726e-06	1.6947e-02	5.9372e+00	2.0446e+00	1.7624e+01	-5.0000e-01	1.4143e+01	2	39211	1	2204632111554121	
det	t	wgt	psc	amfp	ddetx	radius	erg	cell	nps	nch	ijk	an
1	4.7921e-06	1.9669e-02	6.4190e+00	2.3362e+00	2.0136e+01	-5.0000e-01	1.4150e+01	2	39757	1	0657221445360431	

In the table summarizing the contributions to the detector up to 30000 histories, *Mean Free Paths* is contributions to the detector from collisions within one to eight mean free paths and greater than eight. The second column is the fraction of the contributions to the detector as a function of mean free paths from the detector. The third column is the sum of the contributions to the detector up to one mean free path away, from two to three mean free paths away, etc. The sum of this column is the Tally 5 flux. The last column is the accumulative fraction of the total tally contributions as a function of mean free path.

Examining this table, we see the reason for the trouble with detectors in this problem. There are relatively few contributions to the detector close to it. Only 0.28% of the contributions to the detector have come from within two mean free paths. The more collisions there are close to a detector, the better the detector performs. With few collisions, the detector is vulnerable to perturbations from an occasional large contribution like the one from history 36575. The analog detector is more vulnerable than the OMCFE detector. Having few collisions close to the detector frequently causes the detector result to be too low.

One use of this summary table is to help you set λ_{max}, the first entry on the DD card. Beyond λ_{max} mean free paths, Russian roulette is played for contributions to the detector which can account for a significant savings in running time in some problems (but not in this particular one).

In this particular problem an obvious place (from looking at the second and last columns) to set λ_{max} is at four mean free paths. We see from the last column that over 98% of the tally comes within four mean free paths, but from the second column 40% of the contributions come from beyond four. Setting λ_{max} at four, only 5% of the contributions come from beyond four mean free paths. In this particular problem very little running time is saved. However, if there had been some complicated geometry with several different cells, surfaces, and materials beyond λ_{max}, the effect could have been significant.

We see that as of 30000 histories, there have been 50825 contributions to the detector, the average contribution has been 3.040×10^{-8}, and the largest contribution has been 7.960×10^{-5}.

Looking through the diagnostic printout for the contributions at least 100 times greater than the average, we see the largest is 4.288×10^{-4} for the 36575^{th} history. This remains the largest until 240000 histories when a slightly larger one comes along.

In this diagnostic print, DET is the number of the detector involved, T is the actual contribution to the detector, WGT is the weight of the

track making the contribution, PSC is the probability of scattering toward the detector from the collision point, AMFP is the number of mean free paths between the collision point and the detector, DDETX is the distance from collision to detector, RADIUS is the radius of the imaginary sphere around the detector, ERG is the energy of the contributing track, CELL is the cell in which the collision occurred, NPS is the number of the history, NCH is the number of collisions in the history so far, IJK is the starting random number of the history (KRNT on the DBCN card to restart that history), and AN signifies this was an analog contribution (which is the only type ring detector available) versus an OMCFE contribution. This AN flag is really useful only for a OMCFE detector where the contribution can be either once-more-collided or analog depending on the previous collision.

IV. EVENT LOG AND GEOMETRY ERRORS

If a geometry is improperly specified, it will probably be caught in the IMCN overlay. Even though IMCN catches no errors is no reason to assume the geometry is correct. Before doing a serious run, you should always run a few particles and also look at the geometry with the plotter.

When a particle gets to a place in the geometry that is not correctly specified, it gets lost - it simply does not know where to go next. If this happens, you will get on the output file a debug print and event-log print for each of ten lost particles before MCNP terminates. The default of ten lost particles for printing and termination can be changed with the LOST card, but this is generally an unwise thing to do.

A. Event Log

An event-log print is produced by a lost particle (MCNP backs up a lost particle and reruns it, producing the event log on the second go around - which will make some of the summary information slightly off) and also by the third and fourth entries on the DBCN card. SHELL2 in MCNPEX is the same as SHELL but all tallies were taken out, the problem runs only two histories, an event log is forced by the DBCN card, and the shell is given an importance of two to cause a source particle to split when it leaves the source cell and enters the shell. The event log is reproduced in Figure 5.3.

In the event log, SRC is source, S is surface, C is collison, T is

termination, and BNK is return a track from the bank. TYR refers to the reaction type used in COLOUT(3), see page 323.

event log for particle history no. 1

	x	y	z	u	v	w	erg	wgt	tme	cell	
src	0.000+00	0.000+00	0.000+00	2.724-01	9.578-01	-9.137-02	1.419+01	1.000+00	0.000+00	1	
s	9.806+01	3.448+02	-3.289+01	2.724-01	9.578-01	-9.137-02	1.419+01	5.000-01	6.910+00	2	surf= 1 npa= 1
c	1.023+02	3.598+02	-3.432+01	4.159-01	7.449-01	5.217-01	1.380+01	4.437-01	7.210+00	2	8016.04 tyr=-99
c	1.030+02	3.610+02	-3.345+01	1.642-01	8.223-01	-5.448-01	6.853+00	3.921-01	7.243+00	2	8016.04 tyr= -1
t	1.030+02	3.610+02	-3.345+01	1.642-01	8.223-01	-5.448-01	6.853+00	3.921-01	7.243+00	2	energy cutoff
bnk	9.806+01	3.448+02	-3.289+01	2.724-01	9.578-01	-9.137-02	1.419+01	5.000-01	6.910+00	2	import. sampling
c	1.007+02	3.542+02	-3.379+01	-1.824-01	-1.341-01	-9.740-01	5.715+00	4.437-01	7.098+00	2	8016.04 tyr= -1
t	1.007+02	3.542+02	-3.379+01	-1.824-01	-1.341-01	-9.740-01	5.715+00	4.437-01	7.098+00	2	energy cutoff

event log for particle history no. 2

	x	y	z	u	v	w	erg	wgt	tme	cell	
src	0.000+00	0.000+00	0.000+00	3.056-01	3.129-01	-8.993-01	1.419+01	1.000+00	0.000+00	1	
s	1.100+02	1.127+02	-3.237+02	3.056-01	3.129-01	-8.993-01	1.419+01	5.000-01	6.910+00	2	surf= 1 npa= 1
c	1.133+02	1.160+02	-3.334+02	3.008-02	9.422-01	-3.338-01	6.699+00	4.437-01	7.116+00	2	8016.04 tyr= -1
t	1.133+02	1.160+02	-3.334+02	3.008-02	9.422-01	-3.338-01	6.699+00	4.437-01	7.116+00	2	energy cutoff
bnk	1.100+02	1.127+02	-3.237+02	3.056-01	3.129-01	-8.993-01	1.419+01	5.000-01	6.910+00	2	import. sampling
t	1.138+02	1.165+02	-3.348+02	3.056-01	3.129-01	-8.993-01	1.749+00	4.644-01	7.147+00	2	energy cutoff

Figure 5.3

The first neutron starts with the correct parameters and immediately crosses surface 1 into cell 2 as we would expect since cell 1 is a void. Since the cell importance increases to 2 in cell 2, the original particle is split into two tracks, one of which is put in the bank (NPA=1) and the other followed. If there had been a four-for-one split instead of two-for-one as we have here, NPA would be 3 indicating one entry into the bank representing three tracks.

The next event is a collision for the track that is being followed. It has an elastic collision in the center of mass system (TYR is -99) with oxygen in cell 2. Its energy after the collision is 13.8 MeV. A second collision follows that is again with oxygen in the center of mass system, but this time it is inelastic with one neutron out. The energy after collision is 6.853 MeV which results in a termination since the energy cutoff in the problem is 12 MeV.

At this point the bank is checked for any tracks and one is found that got there as a result of importance sampling. That track is started at the point where it was created, and it has an inelastic collision with oxygen that results in its termination due to energy cutoff.

The second source particle is started. It is split, has one collision with oxygen, and dies because of energy cutoff. The second track of this second source neutron is returned from the bank. According to the event-log print the next thing that happens is termination from energy cutoff. There is a path in the code that can skip over the place where the event log would have registered a collision. This second track took that path, but it is obvious it had a collision since there was a loss of energy resulting in energy cutoff.

By default only 600 lines of the event log are printed for each history. This can be changed by the fifth entry on the DBCN card.

B. Debug Print

In addition to getting the event-log print for a lost particle, you will also get a debug print which will give you additonal information. In particular it will tell you what the geometry description is in terms of cell/surface relations at the point the particle get lost. Frequently the problem is an incorrectly specified sense.

As an example of a debug print, if the geometry of Example 2 in Chapter 4, page 178, is specified incorrectly such that the undefined tunnel going off to the right of surface 5 remains, you will get the following print:

```
   lost particle no.  1     no cell found in subroutine newcel

 the neutron currently being tracked has reached surface     5.  there
 appears to be no cell on the other side of the surface at that point.

 the neutron is in cell     2.

 x,y,z coordinates:          5.68757e-01     2.00000e+00    -1.90774e-01
 u,v,w direction cosines:    2.72389e-01     9.57840e-01    -9.13653e-02
 energy = 1.40000e+01    weight = 1.00000e+00     time = 4.03507e-02
 sqrt(z**2+x**2) = 5.99900e-01
 the distance to surface    5 from the last event is 1.04402e+00
 the distance to collision from the last event is 8.33256e+02
 the track is part of history no.         1
 the number of collisions so far in this history is     0.
 the starting random number of this history was 7062511534100051b

 the cells on the other side of surface    5 of cell    2
 (and the surface with respect to which the point x,y,z had
 the wrong sense) are: (see chapter 5 of the mcnp manual.)

    3(
```

The last line of 3(would be 3(*surface number*) if a surface were involved with a wrong sense which is not the case in this problem. In this example, otherside-cells were specified in the geometry description. By not specifying otherside cells, the last four lines in the debug print for this particular problem would not appear since (at the time of this lost particle), the code does not know of any cells on the otherside of surface 5.

V. KCODE

Following is a portion of the output from a KCODE calculation of a GODIVA critical assembly.

```
 mcnp   version 2    07/10/79                    u 07/27/79  08:27:19
 ******************************************************************
       inp          =    godiva

    1-       godiva
    2-            1 51 4.799-2 -1,2
    3-            2 0 1,1
    4-
    5-            1 so 8.7407
    6-
    7-           in 1 0
    8-          m51 92235. 9.376954-1 92238. 5.205251-2 92234. 1.025214-2
M1  9-       kcode 3000 1. 5 10 4500
M2 10-       ksrc  0 0 0
   11-        ergn 0
   12-        cutn 1.0e8 0 .4 .04
   13-        ctme 5.
   14-       print
   15-        mode 0
   16-
warning.  there are no tallies in this problem.
```

initial source from ksrc card.

N3	original number of points		1
	points not in any cell	0	
	points in cells of zero importance	0	
	points in void cells	0	
	total points rejected		0
	points remaining		1
	points after expansion or contraction		3000
	nominal source size		3000
	initial guess for k(eff.)		1.000000
	cycles to skip before tallying		5

source distribution written to file srctp

```
   neutron cross sections

   nuclide    storage

nuclides from file rmccs          rmccs   04/12/78
 92234.10      2246     u-234  endl 7167 13 apr 1976    t=0.0                prompt nu          ( 7167) 15 apr 76
 92235.19     16174     u-235 endf/b-iv t=3*e+4  24 february 1976           prompt nu          ( 1261)  8 mar 78
 92238.13     27087     u-238   endf/b-iv  (t409)  t=3.e+4.      26jan76    prompt nu          ( 1262)  8 mar 78

   total      45507


apportionment of lcm2
 tallies                    0
 kcode source           22505
 bank                    2200
 variable common backup   413
 user file buffers          0
 photon cross sections      0
 neutron cross sections 45507


**************************************************************************
dump no.  1 on file runtpe          nps =       0          ctm =   0.00

       1 warning messages so far
```

starting mcrun. lcm field length = 141032 = 0423350b cp0 = .05

godiva

nps	x	y	z	ia	ja	u	v	w	tme	wgt	erg
1	0.	0.	0.	1	0	2.724e-01	-9.578e-01	-9.137e-02	0.	1.000e+00	2.209e+00
2	0.	0.	0.	1	0	3.056e-01	3.129e-01	-8.993e-01	0.	1.000e+00	4.904e+00
3	0.	0.	0.	1	0	5.717e-01	-7.962e-01	1.982e-01	0.	1.000e+00	3.809e-01
4	0.	0.	0.	1	0	-2.503e-01	-4.739e-01	-8.442e-01	0.	1.000e+00	1.331e+00
5	0.	0.	0.	1	0	9.077e-01	-3.611e-01	2.136e-01	0.	1.000e+00	1.902e+00
6	0.	0.	0.	1	0	6.862e-01	6.811e-01	2.553e-01	0.	1.000e+00	4.410e-01
7	0.	0.	0.	1	0	4.628e-01	7.977e-01	3.866e-01	0.	1.000e+00	4.750e-01
8	0.	0.	0.	1	0	-9.478e-01	-1.175e-01	-2.965e-01	0.	1.000e+00	4.136e+00
9	0.	0.	0.	1	0	-6.401e-01	7.540e-01	-1.479e-01	0.	1.000e+00	7.453e-02
10	0.	0.	0.	1	0	8.672e-01	-2.841e-01	-4.091e-01	0.	1.000e+00	3.128e+00
11	0.	0.	0.	1	0	-8.584e-01	2.292e-01	-4.588e-01	0.	1.000e+00	1.014e+00
12	0.	0.	0.	1	0	1.116e-02	2.035e-01	-9.790e-01	0.	1.000e+00	1.395e+00
13	0.	0.	0.	1	0	8.237e-01	5.421e-01	-1.661e-01	0.	1.000e+00	7.748e-01
14	0.	0.	0.	1	0	2.086e-01	-8.823e-01	-4.220e-01	0.	1.000e+00	1.101e+00
15	0.	0.	0.	1	0	4.248e-01	-9.006e-01	9.153e-02	0.	1.000e+00	1.951e+00
16	0.	0.	0.	1	0	-7.764e-01	-3.462e-01	-5.266e-01	0.	1.000e+00	2.186e+00
17	0.	0.	0.	1	0	-3.403e-01	-3.683e-01	8.652e-01	0.	1.000e+00	1.865e+00
18	0.	0.	0.	1	0	-1.028e-01	3.632e-01	9.260e-01	0.	1.000e+00	1.229e+00
19	0.	0.	0.	1	0	1.875e-01	-6.654e-01	-7.226e-01	0.	1.000e+00	1.305e+00
20	0.	0.	0.	1	0	-9.296e-01	3.559e-01	9.599e-02	0.	1.000e+00	1.000e+00
21	0.	0.	0.	1	0	1.346e-01	-2.447e-01	9.602e-01	0.	1.000e+00	3.990e+00
22	0.	0.	0.	1	0	-8.666e-01	3.015e-01	-3.977e-01	0.	1.000e+00	2.665e-01
23	0.	0.	0.	1	0	6.324e-01	8.105e-02	-7.704e-01	0.	1.000e+00	1.156e+00
24	0.	0.	0.	1	0	7.391e-01	6.198e-01	2.636e-01	0.	1.000e+00	2.669e+00
25	0.	0.	0.	1	0	-4.460e-01	8.107e-01	-3.794e-01	0.	1.000e+00	2.185e+00
26	0.	0.	0.	1	0	4.875e-01	8.662e-01	-1.099e-01	0.	1.000e+00	4.225e+00
27	0.	0.	0.	1	0	-1.177e-01	-8.874e-01	-4.456e-01	0.	1.000e+00	1.079e+00
28	0.	0.	0.	1	0	9.988e-01	2.035e-04	4.894e-02	0.	1.000e+00	3.461e+00
29	0.	0.	0.	1	0	-6.569e-01	1.207e-01	-7.443e-01	0.	1.000e+00	1.836e+00
30	0.	0.	0.	1	0	9.987e-01	-1.801e-02	4.691e-02	0.	1.000e+00	4.556e-01
31	0.	0.	0.	1	0	-6.446e-02	8.113e-01	5.810e-01	0.	1.000e+00	6.415e-01
32	0.	0.	0.	1	0	1.705e-01	8.931e-01	-4.163e-01	0.	1.000e+00	2.764e+00
33	0.	0.	0.	1	0	3.075e-01	9.350e-01	1.767e-01	0.	1.000e+00	2.785e-01
34	0.	0.	0.	1	0	-7.639e-01	4.272e-01	-4.836e-01	0.	1.000e+00	9.097e-01
35	0.	0.	0.	1	0	3.773e-01	6.858e-03	9.261e-01	0.	1.000e+00	3.360e-01
36	0.	0.	0.	1	0	-3.060e-01	-5.581e-01	7.713e-01	0.	1.000e+00	6.376e-01
37	0.	0.	0.	1	0	-3.715e-01	-2.465e-02	9.281e-01	0.	1.000e+00	2.186e+00
38	0.	0.	0.	1	0	-6.938e-02	7.584e-01	-6.480e-01	0.	1.000e+00	7.314e-01
39	0.	0.	0.	1	0	-3.150e-02	1.064e-01	9.938e-01	0.	1.000e+00	2.997e-01
40	0.	0.	0.	1	0	4.710e-01	-8.627e-01	1.843e-01	0.	1.000e+00	1.444e+00
41	0.	0.	0.	1	0	-9.073e-01	1.742e-01	3.827e-01	0.	1.000e+00	1.914e+00
42	0.	0.	0.	1	0	-6.797e-01	-5.789e-01	-4.504e-01	0.	1.000e+00	1.502e+00
43	0.	0.	0.	1	0	-8.193e-01	-3.227e-01	-4.740e-01	0.	1.000e+00	5.971e+00
44	0.	0.	0.	1	0	-7.368e-01	-1.367e-01	-6.621e-01	0.	1.000e+00	1.827e+00
45	0.	0.	0.	1	0	-2.760e-01	-9.374e-01	-2.124e-01	0.	1.000e+00	1.928e+00
46	0.	0.	0.	1	0	1.343e-01	6.070e-01	-7.833e-01	0.	1.000e+00	1.351e+00
47	0.	0.	0.	1	0	-4.385e-01	-2.868e-01	-8.518e-01	0.	1.000e+00	2.288e+00
48	0.	0.	0.	1	0	7.132e-01	-2.280e-01	-8.628e-01	0.	1.000e+00	1.230e+00
49	0.	0.	0.	1	0	-7.738e-01	2.381e-01	-5.871e-01	0.	1.000e+00	1.433e+00
50	0.	0.	0.	1	0	-7.912e-01	5.868e-01	1.657e-01	0.	1.000e+00	6.572e-01

cycle 1 k(collision) 1.353042 removal lifetime(abs) 8.0982e-01 source points generated 4030

cycle 2 k(collision) 1.134063 removal lifetime(abs) 6.6897e-01 source points generated 2525

cycle 3 k(collision) 1.078055 removal lifetime(abs) 6.3271e-01 source points generated 2838

```
      cycle    4  k(collision)  1.026705  removal lifetime(abs)  5.8724e-01  source points generated   2868

      cycle    5  k(collision)  1.027925  removal lifetime(abs)  5.9982e-01  source points generated   2983

      cycle    6  k(collision)  1.000006  removal lifetime(abs)  5.8809e-01  source points generated   2942
```

N6

estimator	cycle 7	ave of 2 cycles		combination	simple average		combined average		corr
k(collision)	1.008255	1.004131	.0041	k(col/abs)	0.000000	0.0000	0.000000	0.0000	0.0000
k(absorption)	1.007082	1.003773	.0033	k(abs/tk ln)	0.000000	0.0000	0.000000	0.0000	0.0000
k(trk length)	1.017660	1.018455	.0008	k(tk ln/col)	0.000000	0.0000	0.000000	0.0000	0.0000
rem life(col)	5.8454e-01	5.8644e-01	.0032						
rem life(abs)	5.8535e-01	5.8672e-01	.0023	life(col/abs)	0.	0.0000	0.	0.0000	0.0000

N7 source points generated 3034

N8

estimator	cycle 8	ave of 3 cycles		combination	simple average		combined average		corr
k(collision)	.977773	.995345	.0091	k(col/abs)	.995475	.0088	.998168	.0049	.9994
k(absorption)	.979270	.995605	.0084	k(abs/tk ln)	1.002602	.0085	.993109	.0301	.9606
k(trk length)	.991890	1.009600	.0088	k(tk ln/col)	1.002472	.0089	1.006217	.0283	.9503
rem life(col)	5.7424e-01	5.8237e-01	.0072						
rem life(abs)	5.7489e-01	5.8278e-01	.0069	life(col/abs)	5.8258e-01	.0071	5.8547e-01	.0090	.9978

source points generated 2999

estimator	cycle 9	ave of 4 cycles		combination	simple average		combined average		corr
k(collision)	.984685	.992680	.0070	k(col/abs)	.992572	.0069	.991859	.0079	.9950
k(absorption)	.983040	.992464	.0068	k(abs/tk ln)	.998378	.0074	.981232	.0122	.9503
k(trk length)	.988367	1.004292	.0082	k(tk ln/col)	.998486	.0074	.988502	.0121	.9173
rem life(col)	5.6503e-01	5.7804e-01	.0091	k(col/abs/tk ln)	.996478	.0072	.958446	.0216	
rem life(abs)	5.6472e-01	5.7826e-01	.0092	life(col/abs)	5.7815e-01	.0092	5.7714e-01	.0111	.9985

source points generated 3020

estimator	cycle 10	ave of 5 cycles		combination	simple average		combined average		corr
k(collision)	.989666	.992077	.0055	k(col/abs)	.992056	.0054	.991916	.0058	.9947
k(absorption)	.990317	.992034	.0053	k(abs/tk ln)	.997668	.0058	.981469	.0085	.9485
k(trk length)	.999337	1.003301	.0064	k(tk ln/col)	.997689	.0058	.987741	.0087	.9179
rem life(col)	5.6453e-01	5.7534e-01	.0085	k(col/abs/tk ln)	.995804	.0056	.969834	.0160	
rem life(abs)	5.6448e-01	5.7551e-01	.0086	life(col/abs)	5.7542e-01	.0086	5.7423e-01	.0095	.9990

source points generated 3046

N9 source distribution written to file srctp

Notes:

$\mathcal{N}1$: This card signals a KCODE calculation with a nominal source size of 3000 particles, an initial k_{eff} of 1.0, and skip five cycles before tallying or averaging.

$\mathcal{N}2$: The initial source points are at x=y=z=0.

$\mathcal{N}3$: This table gives detailed information on the source such as points rejected and accepted plus other information related to the KCODE.

$\mathcal{N}4$: Note that prompt $\bar{\nu}$ values were used as a result of no TOTNU input card.

$\mathcal{N}5$: Five cycles are skipped before printing is started at the end of the sixth cycle.

$\mathcal{N}6$: There are three k_{eff} estimators: collision, absorption, and track length. Which one is more appropriate is problem dependent; averages of the three in various combinations are frequently used (see page 105 in Chapter 2). A rule of thumb is to take the result with the lowest variance.

$\mathcal{N}7$: The number of source points generated in a particular cycle.

$\mathcal{N}8$: This is the correlation coefficient between corresponding pairs of estimators. In this particular problem, the results are highly correlated, so one result is as good as another.

$\mathcal{N}9$: The source file SRCTP, as opposed to the RUNTPE, has been written and can be used in the future for continuation of this problem or to start another problem.

To do a continue-run with the source file SRCTP produced by the above initial run, the input file becomes

```
GODIVA WITH USE OF SOURCE TAPE
    1 51 4.799-2 -1,2
    2 0 1,1

    1 SO 8.7407

   IN 1 0
  M51 92235. 9.376954-1 92238. 5.205251-2 92234. 1.025214-2
KCODE 3000 1. 5 10 4500
 ERGN 0
 CUTN 1.0E8 0 .4 .04
 CTME 5.
PRINT
 MODE 0
```

Because of the absence of the KSRC card, the source file SRCTP of the initial run is used. The number of source points beginning this continue-run is 3046, the same as in Cycle 10 at the end of the initial run. At the conclusion of the continue-run, a new distribution of source points is written on SRCTP.

Appendix A

HOW TO USE MCNP ON LTSS

This appendix is divided into four sections:

(A) MCNP Quick and Easy
(B) Running MCNP
(C) Updating MCNP
(D) DBCTRL

Some example XEQ and ORDER files for automating the following procedures and for running unattended production jobs are illustrated in this Appendix, but in no way are these examples you should necessarily use. They are merely possible ways of doing things, and you should rewrite them to do your particular job in your particular environment. However, you will probably find much in the example files useful to you. Any items you will most likely need to change are written in italics.

The executable binary file for Version 2 is called MCNP which is a public file that the user does not have to get.

All of the XEQ and ORDER files plus examples of MCNP input files mentioned in this Appendix can be obtained from the LIX file MCNPEX that is under /X6CODE on the Common File System (CFS) which is accessed with the MASS utility routine. Other individual files also on disk under /X6CODE include:

MCNPID	Code with UPDATE identifiers and index of subroutines
MCNPIX	Master index of MCNPID
MCNPMP	LOD map
MCNPPL	Program library (OLDPL)
MCNPLF	LOD instruction file
MCNPBL	Binary library (relocatable) from LIBMAK

To print the code and index files:

```
ALLOUT MCNPID SEQ. CCSP. PPF. 800 LONG. BOX yourname
ALLOUT MCNPIX CC. LONG. BOX yourname
```

The latest MCNP Version 2 master file (normally not needed by the user) is a LIX file called LIXn where n is the largest integer found when listing the files under /X6CODE/MCNP2D on CFS:

```
MASS LIST /X6CODE/MCNP2D
```

This file contains backups for the public file MCNP, backups to other files on disk under /X6CODE, plus other related files. Included is also a file called INFO that describes all the files in LIXn.

The public cross-section files are RMCCS, AMCCS, DRMCCS, TMCCS, and MCPLIB. XMCCS and UMCCS are on disk under /X6CODE and are not public files. Backups for all these files are also under /X6CODE under the respective directory names RMCCSD, AMCCSD, DRMCCSD, TMCCSD, MCPLIBD, XMCCSD, and UMCCSD. The latest version of a backup file can be obtained by listing the contents of a particular directory like RMCCSD and taking the file with the largest integer in its name such as R6.

If a public file ever suffers from a disk problem (which does happen from time to time with the large cross-section files) and you need it before X-6 can get it replaced, simply get the backup file in your local file space and continue as usual (don't forget to switch the name R6 to RMCCS, however).

All the above files have universal read-only access.

A. MCNP Quick and Easy

This section is for someone who has never run MCNP and knows nothing about it. A basic knowledge of LTSS is assumed, however. If this assumption is wrong, contact the Consultants in C-Division. This elementary primer will show you how to run a simple job and then refer you to specific pages in the Manual where you can find a description of the basics you will need to know to set up your own simple jobs. Quickly learning how to set up and run simple jobs may give you a false sense of confidence, however. Before attempting more difficult jobs (especially those requiring variance reduction techniques) you should spend time studying appropriate sections of Chapter 2 and come to X-6 for guidance. This cannot be emphasized too strongly when using detectors; unfortunately they are very easy to use but can get you into a lot of trouble.

It is implicit in the following examples that at the end of each input line the RETURN key is hit.

APPENDIX A
Quick and Easy

To run a job with 10 MeV neutrons starting isotropically from the center of a bucket of water where you want to know the current of neutrons getting out the top and also the side of the bucket, get the file BUCKET from the LIX file MCNPEX under /X6CODE. Then simply type in

```
MCNP INP=BUCKET / 1 p
```

This will produce an output file OUTP that you can examine at your terminal and/or send to a printer by

```
ALLOUT OUTP CC. BOX yourname
```

Looking at the output, you will see that 1.918E-01 ± 1.2% neutrons per starting neutron get out the top and 5.403E-01 ± 0.5% get out the side. Adding E1, C1, and T1 input cards with appropriate entries on them will make the results energy, direction, and time dependent, respectively. Note that this bucket has no real walls; the geometry is a chunk of water in the shape of a bucket.

To take a look at the geometry with the PLOT overlay, simply type in

```
MCNP INP=BUCKET IP / 1 p
```

After you are prompted with a question mark, if you are at an interactive graphics terminal type in

```
PX=0 EX=20
```

and at the next prompt type in END. If you want to do more with PLOT, see the instructions in Appendix B. If you are not at an interactive graphics terminal precede the above line with

```
TERM=0 DISP=GIVE
```

so your plots will come out on 35-mm film.

To understand the input file BUCKET and gain some insight into setting up your own problems, see pages 11 through 24 about cells and surfaces, thumb through Chapter 3 to get confused about (but exposed to) all the available input cards, and refer to Chapter 4 for examples of specifying tallies and more exotic geometries. After a little expertise has been gained in the above areas (try doing other things with BUCKET like adding

different tallies), spend some time in the remainder of this Appendix to gain a little more versatility in the mechanics of running MCNP.

A more realistic bucket than the preceding one is set up in BUCKET2 that is also in MCNPEX. This bucket has real walls. It would be instructive to plot it and to decipher the input file. This geometry can be plotted exactly like the BUCKET geometry, but you will get a better plot if you use

```
B=0 1 0 0 0 1 EX=13 O=0 0 10
```

One final example in MCNPEX is the input file LOGO. It is a legitimate geometry consisting of two cells and fifteen surfaces. Plot it with the command

```
B 1 0 0 0 1 0 O 2800 0 0 EX 3700 L 0
```

Note that equal marks are omitted from this command and the character just before the 2800 is the letter *O*. After you have deciphered and understood this input file, contact X-6 for your diploma (with honors) from the MCNP *geometry school*.

B. Running MCNP

This section consists of specifying the MCNP execution line, interrupts, a simple controller, and a BATCH controller.

1. Execution Line:

The MCNP execution line has the following form:

```
MCNP Files Options / t p
```

where *Files* and *Options* are described below (the order on the execution line is irrelevant).

Files: MCNP uses several files for input and output. The files most frequently encountered by the user are listed in Table A.1. A complete list of the MCNP files is found at the beginning of the main overlay MCNP in MCNPID.

Table A.1
MCNP Files

Default Name	Description
INP	Problem input specification
OUTP	BCD output for printing
SRCTP	KCODE source distribution
RUNTPE	Binary start-restart data
RMCCS	Neutron cross sections
AMCCS	"
XMCCS	"
UMCCS	"
DRMCCS	Discrete reaction cross sections
TMCCS	$S(\alpha,\beta)$ data
MCPLIB	Photon cross sections
DUMN1	Dummy file name for user
DUMN2	Dummy file name for user

The default name of any of the files in the above table can be changed on the MCNP execution line by entering

Default Name_1=User File Name_1 Default Name_2=User File Name_2 etc.

For example, if you have an input file called MCIN and wish the BCD output file to be MCOUT, the execution line would be

```
MCNP INP=MCIN OUTP=MCOUT / t p
```

If there are no changes in default names, then nothing is entered for *Files*.

If a file exists in your local file space with the same name of a file MCNP needs to create, the file is created with a different unique name by changing the last letter of the name of the new file to the next letter in the alphabet. For example, if you already have an OUTP, MCNP will create OUTQ.

Options: There are two kinds of options: (a) overlay execution options and (b) "period" options.

(a) Execution: MCNP consists of up to four distinct operations, each performed by a single overlay. These operations and their corresponding

overlays are listed below along with a one-letter mnemonic for each operation.

Table A.2
Execution Options

Mnemonic	Overlay	Operation
I	IMCN	Process problem input file
P	PLOT	Plot or move geometry
X	XACT	Process cross sections
R	MCRUN	Particle transport

By omitting *Options*, the default is IXR. The execution of the overlays can be controlled by entering the proper mnemonic on the execution line. If more than one operation is desired, the single characters (in any order) may be combined to form a string.

Examples of use are I to have IMCN look for input errors, IP to debug a geometry by plotting, and IX to see how much cross-section space is required.

The above mnemonics cannot be used for a continue-run.

(b) Period: The following "period" options add more flexibility in running MCNP.

Table A.3
Period Options

Mnemonic	Operation
ALL.	Equivalent to IPXR
C. m	Continue a run starting with m^{th} dump. If m is omitted, last dump is used. See page 117.
DBUG. n	Write debug information of NMBD (see DBCN card) every n particles
GROSS.	Do not reduce LCM field length (see page 340)
NOTEK.	Assumes use of a TTY versus a graphics terminal; PLOT output is in a local film file FR80S. Equivalent to TERM=0 (see page 295)

FATAL.	Transport particles even if fatal errors are found by IMCN
D.	Destroy drop file unless there is an abnormal termination of MCNP. Default is do not destroy drop file.
PRINT.	Create the full output file; equivalent to PRINT card (see page 170)
MCT. n	Change the default time interval to save RUNTPE on magnetic tape from 120 minutes to n minutes

As an example of an execution line,

```
MCNP DBUG. 1000 INP=MCIN PRINT. / t p
```

results in file MCIN being used for input, all overlays but PLOT being executed, a full output file OUTP, and a debug statement being written to OUTP every 1000 particles.

One final item valid on the execution line is XS which can be used to specify user-supplied cross-section libraries and/or alter the order of search through the standard libraries (see page 157).

After a job has been run with MCNP, the BCD print file OUTP in your local file space may be examined with TRIX AC and/or sent to a printer and/or microfiche with ALLOUT (use the CC. option to obtain carriage controls).

2. Interrupts

MCNP allows four interactive interrupts while it is running:

(ctrl e) IS	MCNP status
(ctrl e) IT *time*	Change time remaining to *time* in minutes
(ctrl e) IQ	Terminates MCNP
(ctrl e) IP	Pause; continue with Return key

The status of MCNP consists of (1) overlay being run, (2) computer time used so far, and (3) if in MCRUN the number of particles run so far.

The IQ message simply stops MCNP if it is not in the MCRUN overlay. However, if MCRUN is being executed, this command causes the code to stop

after the current particle history. The process allows MCNP to terminate "gracefully," thus producing a RUNTPE and a final print output file.

3. A Simple Controller

MCNP does not have a standard controller. Except for unattended production runs, MCNP is most commonly run by manually getting the necessary input files in your local files space, simply executing it as illustrated in the Quick and Easy section, and then disposing of the output file as desired.

If a controller is desired to automate the process somewhat, something similar to the XEQ file XEQ1 in Figure A.1 could be used (XEQ is superior to ORDER and simpler to use).

Figure A.1
XEQ1

```
$BYP3
$MASS GET /X6CODE/MCNPEX
$LIX MCNPEX \ GET TEST1 \ END
$MCNP INP=TEST1 OUTP=MYOUT PRINT.
$COMPACT MYOUT MYOUT
$ALLOUT PRINTER MYOUT CC. BOX yourname
```

The above file gets an input file TEST1, executes MCNP with the appropriate options, and sends the output file MYOUT to the printer (or microfiche if FR80 is substituted for PRINTER in the ALLOUT line). To use this controller, simply type in (assuming you have XEQ1 in your local files)

```
XEQ XEQ1 / t p
```

MCNP generates the OUTP file (renamed MYOUT here) at a length of 100000_8 words. If that length is exceeded, it spills over into OUTPA, etc. (or in this case MYOUTA, etc.). The COMPACT utility will combine this sequence of files (if it exists) into a single file.

4. BATCH

Only ORDER files can be submitted to BATCH. Therefore, an ORDER file must be created to perform all necessary functions or to get and execute an XEQ file. For BATCH, you will probably find ORDER more appropriate than XEQ since many of the safety features to be discussed are not available with XEQ. See the BATCH writeup for an explanation of the $BATCH line parameters.

All production jobs should be run under MONITOR (see CCF Newsletter Vol. 37, p17) to preserve a record of everything that happened while your job was being executed. This information can be vital in determining why a job did not run; the dayfile from ORDER is generally insufficient for this purpose. To do this, ask the operator to run under MONITOR by adding a comment line right after the $BATCH line.

A good example of an ORDER file (called ORD and available from MCNPEX) for production use is shown in Figure A.2. The ORD tape writing is done independently of and in addition to any that may be done by MCNP itself during execution. ORD makes use of the fact that, by default, the drop file name is always +MCNP, the run file is RUNTPE, and the BCD output file is OUTP.

Figure A.2
ORD

```
$BATCH UA nnnnnn aaa bb hh mm zzzz       COMMENT
$ OPERATOR - PLEASE RUN THIS JOB UNDER MONITOR
*ID   account#                                yourname          box ###
*XEQ MASS
*XEQMES GET /X6CODE/MCNPEX
*NXT
*XEQ LIX
*XEQMES MCNPEX \ GET TEST1 \ END
*NXT
*BRANCH SAVE,SAVE,SAVE,SAVE,SAVE
*TIME n
*XEQ MCNP
*XEQMES INP=TEST1 PRINT.
*NXT  account# SAVE                           yourname          box ###
*XEQ COMPACT
*XEQMES OUTP OUTP
```

```
*XEQ MCT
*XEQMES ID. yourname group CHECK FILES RUNTPE +MCNP OUTP
*XEQMES TAPES * END
*NXT
*XEQ ALLOUT
*XEQMES PRINTER OUTP CC. BOX yourname
*NXT
*XEQ ALLOUT
*XEQMES FR80 OUTP CC. BOX yourname
```

The MCT line is simply good, cheap insurance because the necessary files are saved in case the output is lost, a subsequent continue-run is deemed necessary, or a problem needs to be tracked down with DBCTRL. The tape can be released when everything is satisfactory. Another safe and recommended procedure for production jobs (which is also illustrated in Figure A.2) is to have an intermediate time cutoff to make sure there is enough time to wrap up miscellaneous jobs such as printing and tape writing. This is accomplished by the *TIME n line where n is the time for MCNP to run in minutes. This time should be a couple of minutes less than the time specified on the BATCH line. The *BRANCH line causes the flow to go to the *NXT SAVE line regardless of termination type. Otherwise, by default an operator interrupt by Sense Switch 1 will cause the flow to go to the last *NXT line and skip the tape writing or other intervening functions. Finally, the COMPACT line combines a sequence (if one exists) of output files into one file.

C. Updating MCNP

When updating MCNP, three files (in addition to an UPDATE patch) are required in your local file space:

MCNPPL	UPDATE program library (OLDPL)
MCNPBL	Relocatable binary library
MCNPLF	LOD instructions

These files are available from /X6CODE. The UPDATE patch is prepared based on the identifiers from the file MCNPID. Note that each subroutine in MCNP is a separate UPDATE deck. Both a sequential and alphabetical list of all MCNP subroutines are on the last page of the MCNPID file. A list of all subroutines as a function of overlay is on page 285 of this Appendix.

Once you have the necessary files, to get a new executable binary called MCNP,

```
UPDATE2 (I=patchname, P=MCNPPL) / 1 p
FTN (CNAME=MCNP,OPT=2,LF=MCNPLF,SET,SYM=^) / 2 p
```

An XEQ file called UPDATE is available from MCNPEX that gets the three necessary files from /X6CODE and executes the above UPDATE2 and FTN lines. The file assumes the name of your patch is PATCH.

The above FTN line creates a compiler listing output file called LISTFTN which will contain any FORTRAN error messages. You can find your errors by opening the file and doing a pattern search on *space*FE*space* with TRIX AC.

An example of a simple patch is in Figure A.3 that puts two WRITE statements near the beginning of the HSTORY subroutine.

Figure A.3

```
*IDENT WLT
*I,HS.8
      WRITE(4,1000)
      WRITE(59,1000)
 1000 FORMAT(7HTESTING)
```

Note that Unit 4 refers to the BCD output file OUTP and Unit 59 is for BCD input/output teletype messages. Unit 2 is reserved for the BCD input file INP. The unit number 4 is not assigned until MC.224, and unit number 2 is not assigned until MC.231. Make sure a patch you write does not require these I/O units before they are assigned.

If you add or replace an entire subroutine (such as SOURCE), you must have *CALL CM (beginning in column 1) right after the beginning of your new subroutine to get the COMMON blocks in. If you change COMMON simply make the change to the CM deck (the first few lines of MCNPID) with your patch; you do not have to do anything else. There is never a need for a *COMPILE card in your patch; the function of this item is taken care of automatically.

To add a SOURCE subroutine, you may replace everything from SO.2 through SO.13 with a complete new subroutine (including *CALL CM) or you may simply replace SO.11 with the necessary FORTRAN lines to define at

least X,Y,Z,IA,JA,ERG,WGT, and TME (see page 140). Note that SO.11 must be removed.

To add an SRCDX subroutine, you may also replace everything that is now in MCNP (SX.2 through SX.8) or just replace SX.6 with the necessary FORTRAN lines to define PSC (see page 199 in Chapter 4) for each detector and/or DXTRAN sphere. In any event, SX.6 must be removed.

The random number generator is the FTN function RANF(1). The number of random numbers called must be incremented manually by adding NRN=NRN+1 after (or just before) every call to RANF(1).

The COMPILE file from UPDATE2 is a FORTRAN listing of only the MCNP subroutines you modified including your modifications and your new identifiers. If you don't change COMMON but want a listing of the complete code including your changes, adding F to the UPDATE2 execution line will cause an update of the full code.

Naturally the individual steps outlined above (obtaining the necessary files, updating, compiling, and loading) can be automated in an XEQ file as illustrated in Figured A.4. This example, XEQ2, adds a SOURCE subroutine (patch is PSOURCE) and then runs a job with the new code *yourcode*. The new code is saved for future use.

Figure A.4
XEQ2

```
$BYP3
$MASS
C DEFAULT DIR=/X6CODE
C GET MCNPEX MCNPPL MCNPBL MCNPLF
C END
$LIX MCNPEX \ GET TEST2 PSOURCE \ END
$UPDATE2 (I=PSOURCE, P=MCNPPL)
$FTN (CNAME=MCNP,OPT=2,LF=MCNPLF,SET,SYM=^)
$SWITCH MCNP yourcode
$MASS SAVE /yourplace/yourcode
$yourcode INP=TEST2
$COMPACT OUTP OUTP
$ALLOUT FR80 OUTP CC. BOX yourname
```

It is strongly discouraged to modify MCNP and then immediately make a long production run. Modifications should be made and followed with a

short test run first. Then if everything is satisfactory, a long run can be made with the saved executable binary and without another UPDATE.

Note that in XEQ2 the FTN line has CNAME=MCNP rather than CNAME=*yourcode*. The latter is perfectly acceptable; however, the main overlay will be *yourcode* and not MCNP. This should be kept in mind when using DBCTRL.

Caution: The above procedures for modifying MCNP are for relatively simple modifications. If more extensive changes are required (such as adding a COMMON block), see Appendix C. Furthermore, MCNP follows a convention for naming variables to help you avoid conflicts with variables in COMMON blocks which are in almost all subroutines; this is also explained in Appendix C.

If you modify MCNP and get the message *FLS is larger than allowed by the system* while loading, your patch requires more storage than is available in SCM. The unmodified MCNP requires 147366_8 words of SCM and the upper limit is 157760_8 words, leaving 4346_{10} extra. How much you used is printed at the conclusion of loading as the second number in the *FLD LGTH* line.

D. DBCTRL

MCNP is set up to use the DBCTRL debug routine on LTSS. To use DBCTRL, the user should get the latest DBCTRL manual from the CCF Program Library.

There are two ways to use DBCTRL:

Static	DBCTRL +MCNP MCNP / t p
Dynamic	DBCTRL MCNP MCNP / t p

Care must be used with FTN and DBCTRL because at FTN OPT=2, an amazing number of intermediate variables are not stored and therefore cannot be interrogated by DBCTRL. Furthermore, OPT=2 can leave out some labels and even add some of its own such as)AA and)AB. For debugging purposes, you may find it advantageous to compile with OPT=0 in order to have more variables available for DBCTRL. After you have finished debugging, recompile with OPT=2 for normal running.

When running MCNP, note that the name of the drop file for your particular job is printed out in the form +MCNP (or +MCNPA, etc., if other drop files of the first name already exist). The symbol table is loaded at the end of the file MCNP (make sure this is your MCNP and not the public

MCNP), and the possible overlays to be entered are MCNP, 1C0, 2C0, 3C0, or 4C0. For DBCTRL, the names of the overlays are followed by semicolons, and names of subroutines are followed by colons.

An alphabetical list of each MCNP subroutine followed by the name of the overlay it is in is in Table A.2. Note that in this table a few subroutines appear in more than one overlay. The last page of the MCNPID file contains two additional lists of subroutines - one in alphabetical order and the other in order of occurrence in MCNP. The numbers in the latter two tables correspond to the line numbers in MCNPID where the subroutines appear.

In Table A.2 and for purposes of DBCTRL, the names of the overlays are MCNP, 1C0, 2C0, 3C0, and 4C0 which refer to the MCNP, IMCN, PLOT, XACT, and MCRUN overlays, respectively.

Table A.2

Subroutines and Overlays

ROUTINE	OVERLAY	ROUTINE	OVERLAY
ACECAS	4C0	BANKIT	4C0
ACECOL	4C0	BIN	4C0
ACECOS	4C0	BNDCEL	2C0
ACEERG	4C0	CALCA	1C0
ACEFCN	4C0	CALCPS	4C0
ACEFIL	3C0	CALCVA	1C0
ACEFPT	3C0	CBIAS	4C0
ACEGAM	4C0	CELSRF	1C0
ACENU	4C0	CHEKCS	1C0
ACERIN	3C0	CHGMEM	MCNP
ACESAB	3C0	CHGNAM	1C0,4C0
ACETBL	4C0	CHKCEL	1C0,2C0,3C0,4C0
ACETOT	4C0	CHKSRC	4C0
ACTION	4C0	CHQCEL	1C0
ADVPAR	2C0	COLIDK	4C0
AMATRX	2C0	COLIDP	4C0
ANALYS	4C0	COLIDN	4C0
ANGLE	1C0,2C0,3C0,4C0	CORNER	1C0
AXIS	1C0	CRSPRO	2C0

APPENDIX A
DBCTRL

DDDET	4C0	QUART	4C0
DRAW	2C0	RANDY	2C0,3C0,4C0
DXDIAG	4C0	RDPROB	1C0
DXTRAN	4C0	REGULA	2C0
ERGSMP	4C0	ROTAS	4C0
ERPRNT	1C0,3C0	SABCOL	4C0
EVENTP	4C0	SECND	4C0
EVLGC	1C0,3C0,4C0	SETUP	2C0
EXPIRE	MCNP	SKCODE	3C0
EXPUNG	3C0	SOURCA	4C0
FISP	4C0	SOURCK	4C0
GETXS	4C0	SOURCE	4C0
HSTORY	4C0	SQQINT	1C0
IFISP	4C0	SRCDX	4C0
IMCN	1C0	STAR	1C0
INTER	2C0	STUFF	1C0
ISOS	4C0	SUMARY	4C0
ISOURC	1C0	SURFAC	4C0
KCALC	4C0	SWAP	2C0
KLEIN	4C0	TALLY	4C0
KSRCTP	1C0,3C0,4C0	TALLYH	4C0
LEGERE	2C0	TALLYX	4C0
LINEAR	2C0	TALLYD	4C0
MCNP	MCNP	TALLYP	4C0
MCRUN	4C0	TALSHF	4C0
MOVIT	2C0	TAREA	1C0
NEWCEL	4C0	TEKDVR	2C0
NORMA	2C0	TESTPT	2C0
OMCFE	4C0	TIMINT	4C0
PHOCRS	3C0	TORUS	4C0
PHOTC	4C0	TPEFIL	3C0,4C0
PHOTOT	4C0	TPEMCT	4C0
PIECES	1C0	TRACK	4C0
PLOT	2C0	TRANSM	4C0
PRLOST	4C0	TTYINT	MCNP
PRNTF	MCNP	UFILES	1C0,3C0
PRSRF	1C0,3C0	UNORM	2C0
PUTLBL	2C0	VIEW	2C0
PUTNQ	1C0	VOLUME	1C0
QMULT	2C0	WHATIS	2C0

WRAPUP	2C0	XACT	3C0
WTMULT	4C0		

When using DBCTRL with dynamically-dimensioned arrays that are equivalenced (such as equivalenced with the DAS array) you need to be careful about addresses. For example, the track bank is the third block of data, IBNK, in the DAS array. However, DAS is apportioned differently for each different problem as shown in the *Apportionment of LCM2* table at the end of the XACT overlay output. From this table, the first word of the bank starts in the DAS array at the sum of the storage for the two blocks ahead of the bank plus one, say 687+0+1=688. So you can get at the first word in the bank by DAS(688) or by IBNK(688). This location is also given by the COMMON variable LBNK which will be set by the time the MCRUN overlay starts.

If you are interested in quantities in the bank, MB*LPBB+LBNK in the BANKIT subroutine is the location of the beginning of the information for the last track put in the bank. NBNK is the number of entries in the bank, and a single entry of LPBB quantities represents PBL(14)=NPA tracks. For example, a 4-for-1 geometry split makes one bank entry with NPA=3. Each time a track is banked, LPBB (which is at least twenty two) quantities are stored. (The three words SPARE1, SPARE2, and SPARE3 are spare words available for the user.) These LPBB items are equivalenced to the PBL array. If a OMCFE detector is used in the problem, NDET more than the twenty two are stored in the bank. See page 328 in Appendix C for a description of the PBL array. Finally, the last track put into the bank will be the first one taken out.

1. Static DBCTRL

An MCNP execution may be statically examined with the DBCTRL facility by starting with the following teletype entry:

```
DBCTRL +MCNP MCNP / t p
```

where +MCNP is the drop file from an error-terminated or manually-terminated MCNP run. The second file name is the name of the associated symbol table. For example, the contents of the variable MXA in subroutine RDPROB in overlay 1C0 may be examined by entering

1C0; RDPROB: MXA

This entry sets the "current overlay and subroutine." Other variables in that specified subroutine may be examined by entering a blank delimited list of variable names. Several elements of an array may be printed out by following the array element with a comma and the number of elements desired [i.e., RHO(17),4 will print out RHO(17), RHO(18), RHO(19), and RHO(20)]. To examine variables in other overlays and subroutines, it is necessary to reset the current overlay and subroutine by entering the new overlay name followed by a semicolon and a new subroutine name followed by a colon (i.e., 4C0; HSTORY:).

An example of static DBCTRL is in Figure A.5 where the input file TEST2 from MCNPEX is used along with a modified PSOURCE patch. PSOURCE has been modified for this example by adding the two lines B=SRC(2) and X=W/B just before the RETURN statement.

The job aborts since SRC(2) has not been set on the SRC card in TEST2. The location of the error (072031) is found in the P register of the exchange package. This address is printed at the teletype terminal.

The user input in this example is in italics.

Figure A.5
Static DBCTRL

```
DBCTRL +MCNP MCNP
 DBCTRL VER.4 11/17/78
? 4C0;
? FIND. 072031
 072031 IS 000014B LOCATIONS AFTER THE ORIGIN OF ROUTINE SOURCE
      IN CODE BLOCK 4C0
? SOURCE:
? W IA JA WGT ERG B X
 W (0071641) = -9.136531E-02
 IA (0071647) = 1
 JA (0071650) = 0
 WGT (0071643) = 0
 ERG (0071642) = 0
 B (0072035) = 1.
 X (0071634) = 0
? END
 ALL DONE
```

At the first prompt from DBCTRL, 4C0; was entered since we know from the messages sent back from MCNP that IMCN and XACT had been completed and MCRUN started. The FIND. command locates the error somewhere near the beginning of the SOURCE subroutine. Several variables are then interrogated. The most suspicious-looking variable is B which is shown to be I. This is the key; B is set by the second entry on the SRC card in TEST2, and this entry was left blank.

Note that in the above example the value of ERG is 0 even though SRC(1) is set to 14.2 on the SRC card in TEST2. If the modified code had been compiled with OPT=0, the values for ERG, B, and X would have been 14.2, 0, and R., respectively.

2. Dynamic DBCTRL

In the dynamic mode, an in-progress MCNP execution may be interrupted for interrogation at various preset breakpoints by use of DBCTRL. The teletype entry

```
DBCTRL MCNP MCNP / t p
```

will begin a dynamic DBCTRL execution. Breakpoints may be set initially or after any pause, and execution is resumed between breakpoints by the RUN. command. Execution of MCNP from a static DBCTRL examination may also be continued with the RUN. command. Breakpoints may be set and held at the beginning of particular subroutines, at statement labels, and at particular octal addresses by the commands BKPSUB., HBKP., and HBKPA. . Corresponding commands can be used to remove breakpoints. The TRACE. command enables a breakpoint to be set after passing a statement label several times (i.e., in the middle of a loop such as after every tenth history). All currently set breakpoints can be listed by the command LISTBKP. Note that the MCNP input card DBCN provides a loop feature in HSTORY for the number of collisions.

The MCNP execution line information (such as the name of an input file) is entered with the RUNM. command.

It should be noted that variables appearing in an MCNP PARAMETER statement and variables in COMMON that are not referenced in the current subroutine may not be available under DBCTRL.

Several other DBCTRL features may also be useful, including the FIND. and REPLACE. commands. The BYON. command will allow you to use the MCNP CTRL-E interrupts such as IP. Detailed examples of the use of DBCTRL

can be found in the C-Division documentation LTSS-529. Note that by default DBCTRL sets a time limit for its controllee (i.e., MCNP) at one minute. If this is not sufficient, it can be changed with the TIME. command.

An example of using DBCTRL dynamically on the TEST1 problem of MCNPEX is found in Figure A.6. This example illustrates setting breakpoints, looking at several variables, and manually changing variables (such as NPP). The user's input is shown in italics.

Figure A.6
Dynamic DBCTRL

```
DBCTRL MCNP MCNP / t p
 DBCTRL VER.4  11/17/78
? COPY. BYON.
 THE CONTROLEE IS NOW MCNP+
? BKPCBALL.
? RUNM. INP=TEST1 PRINT.
 HARDWARE ABORT INTERCEPTION HAS BEEN DISABLED
      FOR THE CURRENT CONTROLLEE
MCNP  VERSION 2       U 10/03/79  15:16:01   DROP FILE IS  +MCNP
 THE CONTROLLEE IS NOW +MCNP
 BREAKPOINT HIT AT THE BEGINNING OF CODE BLOCK 1C0
? NPP NPS WC1(1)
 NPP (0066524) = 0
 NPS (0065427) = 0
 WC1(1) (0066557) = 0
? RUN.
 WARNING.  ENERGY BIN LIMITS ADJUSTED FOR TALLY   2
IMCN  (1C0) IS DONE
 BREAKPOINT HIT AT THE BEGINNING OF CODE BLOCK 3C0
? NPP WC1(1) IPT
 NPP (0066524) = 100000
 WC1(1) (0066557) = -5.E-01
 IPT (0071653) = 0
? RUN.
DUMP NO.  1 ON FILE RUNTPR      NPS =      0    CTM =   0.00
XACT  (3C0)  IS DONE
 BREAKPOINT HIT AT THE BEGINNING OF CODE BLOCK 4C0
```

```
? BKPSUB. MCRUN
? HSTORY: TRACE. 60 3
? TRACE(5),10
 TRACE(5) (0061542) = 0, 2800, 0, 0, 0, 0, 3441, 3442, 0, 0
? RUN.
 BREAKPOINT HIT AT THE BEGINNING OF SUBROUTINE MCRUN
      IN CODE BLOCK 4C0            (ADDRESS - 100546)
? NPS IPT OCT. IJK
 NPS (0065427) = 0
 IPT (0071653) = 0
 IJK (0066440) = 0B
? DEC. RUN.
LCM FIELD LENGTH = 194154 = 0573152B     CP0 =    .19
 BREAKPOINT BEING TRACED AT LABEL 60        OF SUBROUTINE HSTORY
      IN CODE BLOCK 4C0          (ADDRESS = 103666) HAS OCCURRED
           3 TIMES. TRACE FLAG REMOVED BUT BREAKPOINT HELD.
? NPS NPP IPT OCT. IJK DEC. NRN
 NPS (0065427) = 3
 NPP (0066524) = 100000
 IPT (0071653) = 1
 IJK (0066440) = 2475627207331461B
 NRN (0066525) = 229
? RBKPALL. MCRUN: TRACE. 170 25 RUN.
 BREAKPOINT BEING TRACED AT LABEL 170       OF SUBROUTINE MCRUN
      IN CODE BLOCK 4C0         (ADDRESS = 101152) HAS OCCURRED
           25 TIMES. TRACE FLAG REMOVED BUT BREAKPOINT HELD.
? NPS NPP NRN
 NPS (0065427) = 27
 NPP (0066524) = 100000
 NRN (0066525) = 13327
? REPLACE. NPP 35
? BKPSUB. SUMARY
? MCRUN: RBKP. 170 RUN.
 BREAKPOINT HIT AT THE BEGINNING OF SUBROUTINE SUMARY
      IN CODE BLOCK 4C0            (ADDRESS - 105671)
? NPS NPP
 NPS (0065427) = 35
 NPP (0066524) = 35
? RUN.
     RUN TERMINATED WHEN      35 PARTICLE HISTORIES WERE DONE.
```

```
DUMP NO.  2 ON FILE RUNTPE    NPS =     35   CTM =    .59
MCRUN (4C0) IS DONE
MCNP+       LTSS TIME     2.048 SECONDS
CPU =   1.953        SYS=      .096         I/O=     .000
 +MCNP       HAS TERMINATED
 WARNING - THE DROPFILE IS STILL CONSIDERED TO BE +MCNP
           IF THIS IS NOT SO,USE THE CHANGE. COMMAND.
 DYNAMIC COMMANDS ARE HEREWITH DISABLED FOR THE CURRENT CONTROLLEE
? END
 ALL DONE
```

Caution should be used when setting breakpoints in an overlay where the code has not stopped. If you do this, you will get the message *Return address storage could not be located; breakpoint may not be set at the proper location.* This may result in nothing more than a warning but could also be an indication that DBCTRL overwrote part of your code with its breakpoint. For example, the breakpoint that was set at the beginning of MCRUN with BKPSUB. could also have been set near the very beginning of the DBCTRL example of Figure A.6 by first specifying 4C0; before MCNP had even been started with the RUNM. message. The result would have been the warning message and in this particular case nothing more - but do not gamble on it.

Appendix B

THE PLOT OVERLAY

The PLOT overlay of MCNP has two very useful functions: (1) it can be used to plot the problem geometry specified in the INP file and (2) it can be used to rotate and/or translate geometries. The first feature is invaluable for debugging geometries. Furthermore, you should never set up a geometry and run a problem without first plotting the geometry to see if you set up what you think you did.

The second feature, MOVE, can greatly simplify geometry setups. For example, if an ellipse is needed somewhere off the coordinate system axis and canted at 35°, it is a much simpler job to specify the ellipse centered at the coordinate system origin and parallel to an axis. Then you simply translate and rotate it. The MOVE feature of PLOT will do this translation and rotation very conveniently; new surface coefficients and input cards are determined for you automatically.

A. INPUT

The plotter requires a run file (RUNTPE) that was produced by executing the initiation overlay with a regular INP input file. The most common way to use the plotter is at an interactive graphics terminal (such as a Tektronix) where

MCNP INP=*filename* IP / t p

is entered followed by the appropriate PLOT commands when prompted with a question mark. The PLOT commands are in the form of Options and Keywords followed by appropriate data that are input to PLOT interactively.

If you are not using an interactive graphics terminal, you should use the NOTEK. option on the MCNP execution line or set TERM=0 when first prompted by PLOT. The plots will be put in a local film file called FR80S.

Input lines are terminated by hitting the RETURN key on the terminal. Lines can be continued by using the & symbol. Between a Keyword and its data, you may use a blank or an equal sign (=). You will always be prompted with a question mark (?) for input. Decimal points are not required to be entered with real variables.

The error message *Retype the command* means retype the entire input line and not just the faulty section(s).

A film file FR80S is always produced, and by default it is kept in your local file space. A copy of every view you plot is put in FR80S. You can automatically give it to the system for processing by setting DISP=GIVE while you are in PLOT, or you can give it to the system manually after MCNP has ended by

```
CGSGIVE I=FR80S / t p
```

This is a way to get quality slides for a talk or plots for a report rather than using the poorer quality hard copies from a Tektronix.

It is important to remember that once an Option or Keyword is set, it remains in effect until you change it. For example, if you set L=0, you will get no surface labels on your plots until you reset it to 1. Current values may be examined by using WHAT. An exception to this is DEV which must be specified every time you want to use it. Another exception is that values for ORIGIN and BASIS set up by VIEW and ZOOM are not passed to MOVE. The MOVE option requires that you input values for ORIGIN and BASIS at least once after typing the option MOVE.

As an example, to get a plot parallel to the yz-plane at x=3 with an extent of 100 in all directions, at the first prompt just type in

```
PX=3 EX=100
```

This takes advantage of the Option VIEW being the default, and if the second EXTENT parameter is not entered the first value is taken for the second also. To get a similar second plot but you want to change the extent to 75, just type in

```
EX=75
```

at the next prompt.

(1) PLOT Options:

PLOT options are used to specify plotter tasks and to control code operation. The options (allowed abbreviations are indicated in parentheses) are:

VIEW (V) ZOOM (Z) MOVE (M)
WHAT (W) END

VIEW is the default. Options, if used, must be first on the input line.

(2) Keywords:

Following is a list of Keywords and allowed abbreviations, the Options they apply to, default values, and a short description.

Keyword	Option	Default	Description
PX	V		View plane at PX=C normal to x-axis
PY	V		View plane at PY=C normal to y-axis
PZ	V		View plane at PZ=C normal to z-axis
BASIS (B)	VM		Basis vectors
ORIGIN (O)	VM	0 0 0	Point in geometry about which plot is centered
EXTENT (EX)	V	0 0	Half the horizontal and vertical size of the plot
CENTER (C)	Z	0 0	Location of ZOOM plot relative to ORIGIN
FACTOR (F)	Z	1	Magnification factor for ZOOM; can be non-integer and less than one
THETA (T)	Z	0	Degrees to rotate a ZOOM plot counter clockwise
MAP	M	1	Denotes regular or primed coordinate system for MOVE
FR80	VZ	35	0 is no film; 35 is 35-mm film; 105 is microfiche
DEV	VZ		Versatec 36-inch plots
TERM	VZ	3	Terminal type: 1 for Tektronix 4012, 2 for 4014 without enhanced graphics, 3 for 4014 with enhanced graphics. 0 has same effect as NOTEK. on execution line
DISP	VZ	KEEP	Film disposition: KEEP to keep FR80S file and GIVE to give FR80S to system for processing

LS	VZ	960	Terminal linespeed (300 for a low-speed line)
LABEL (L)	VZ	1	1 for surface labels and 0 for no labels
HELP (H)			Lists the Options and Keywords

Plot Option: VIEW
Keywords: PX, PY, PZ, EXTENT(EX), ORIGIN(O), BASIS(B)

The VIEW option (which is the default) directs PLOT to draw the intersection of a rectangular plane section, a *window*, with the MCNP cell boundaries; that is, an arbitrary cross-sectional view contained within a specified rectangle may be plotted.

The rectangular cross-sectional view in MCNP (x,y,z) geometry space is defined by specifying the center of the window relative to the origin of the MCNP geometry, the orientation of the window, and by giving the extent of the window along each of its positive axes about the origin. The orientation is specified by two vectors, one along each axis of the window.

This information is related to the input variables as follows:

ORIGIN(I=1,3) = (x,y,z) coordinates of a point in the MCNP geometry about which the window, defined by BASIS and EXTENT, will be centered.

BASIS (I=1,3) = (x,y,z) components of a vector along the horizontal axis of the plot window

BASIS (I=4,6) = (x,y,z) components of a vector along the vertical axis of the plot window

EXTENT(I=1,2) = the size of the window about ORIGIN along the positive horizontal and vertical axes. The window extends from the origin in a positive horizontal direction EXTENT(1) cm and in a negative horizontal direction EXTENT(1) cm. Similarly, it extends in the positive and negative vertical direction each EXTENT(2) cm. If only one value is input for EXTENT, then EXTENT(2) is taken to be the same as EXTENT(1)

The words *horizontal* and *vertical* refer to the plot window and not to the geometry (x,y,z) space. The window may have any orientation in the (x,y,z) space.

Several examples of determining BASIS vectors are given on page 301.

Use of PX, PY, and PZ:

These input Keywords provide a shorthand way to define BASIS and ORIGIN for special (but frequently used) planes:

PX=C defines the plot window X=C [the plane parallel to the (y,z) plane through the point (C,0,0)] and directs PLOT to define
BASIS = 0 1 0 0 0 1 and
ORIGIN = C 0 0

PY=C defines the plot window Y=C [the plane parallel to the (x,z) plane through the point (0,C,0)] and directs PLOT to define
BASIS = 1 0 0 0 0 1 and
ORIGIN = 0 C 0

PZ=C defines the plot window Z=C [the plane parallel to the (x,y) plane through the point (0,0,C)] and directs PLOT to define
BASIS = 1 0 0 0 1 0 and
ORIGIN = 0 0 C

When using PX, PY, or PZ, the Keywords BASIS and ORIGIN are superfluous and should not be used.

Comments on input:

EXTENT must always be defined with the VIEW option, whether using a PX, PY, PZ, or using BASIS. EXTENT should not be made a lot larger than necessary since the geometry features will have to be small to get them all in the window. Expanded plots can be made by making EXTENT small or by using ZOOM.

ORIGIN is best placed in the middle of the region of the geometry to be plotted. The plot produced will be centered about ORIGIN.

PLOT Option: ZOOM
Keywords: FACTOR(F), CENTER(C), THETA(T)

The input Keywords for ZOOM are defined as follows:

FACTOR = Linear magnification (can be less than one)
CENTER = Horizontal and vertical coordinates of the center of the new plot window *relative* to the values of ORIGIN
THETA = Degrees of rotation, positive or negative, of the ZOOM plot (counter clockwise rotation is positive)

The ZOOM option is similar to VIEW; in fact, it is just another shorthand way to redefine some of the input to VIEW such as EXTENT. ZOOM cannot be the first option chosen in a plotting session because ZOOM does not initially define values needed by VIEW.

PLOT Option: MOVE
Keywords: MAP, BASIS(B), ORIGIN(O)

The MOVE option does no plotting. MOVE produces new surface input cards (in the local LTSS file SURFACE) for a transformation of the MCNP problem geometry space. The transformations allowed are mappings between the original geometry space and another Cartesian coordinate system. No scaling of coordinates is allowed. The input variables to MOVE are sufficient to define all rotations and translations as well as coordinate swapping.

For the purpose of describing the input of MOVE, consider two coordinate systems, namely the (x,y,z) system and the (xp,yp,zp) system, or primed system. The MCNP problem geometry may be considered to be in either of the systems. The following Keywords describe the relationship between the two systems:

MAP = 1 The input surfaces are in the (x,y,z) system and the output surfaces are in the (xp,yp,zp) system
MAP = −1 The input surfaces are in the (xp,yp,zp) system and the output surfaces are in the (x,y,z) system

ORIGIN(I=1,3)	(x,y,z) coordinates of the (xp,yp,zp) origin
BASIS (I=1,3)	(x,y,z) components of a vector along the xp-axis
BASIS (I=4,6)	(x,y,z) components of a vector along the yp-axis

The two cases MAP = 1 and MAP = -1 are included for convenience since either transformation or the inverse transformation may be known from direct geometric considerations.

MCNP rotates the geometry about the point (0,0,0) only. MAP = 1 is the default.

The values for ORIGIN and BASIS set up by VIEW and ZOOM are not passed to MOVE. The MOVE option requires that you input values for ORIGIN and BASIS at least once after typing the option MOVE.

Examples of MOVE and determining its BASIS vectors are given on page 305.

PLOT Option: WHAT
Keywords: None

The Keyword WHAT directos PLOT to list current values of all input values as well as the message from the MCNP execution line. Values listed for ORIGIN and BASIS are those applicable to the last PLOT option used. Since input values remain unchanged until reset, WHAT is a convenient way to inspect values set early in the plotting session and perhaps forgotten.

PLOT Option: END
Keywords: None

The PLOT option END terminates the PLOT overlay and returns control to MCNP which will either terminate or continue to another overlay, depending on the execution-line message.

B. VERSATEC Plots

To get 36-inch Versatec output (only the 36-inch size is allowed by MCNP) go through the following procedure:

(1) Plot something

(2) Type in DEV=VERS after each plot you want sent to the Versatec

(3) When you have finished all plotting and MCNP has ended:

```
MASS GET /C6USER/ESPOUT
```

(4) ESPOUT / t p

```
? type in major title
? type in sub-title
? carriage return for the comic default
? carriage return for 36-in. default
? type in pagesize such as square or long
```

(5) That's all. A magnetic tape is mounted to put the information from the PLOT file created by ESPOUT on tape for processing. If an operator should call you, say you want a PAGES tape. When you get your actual output, there may be some obscure error message on it; ignore it or call someone in C-6 for an explanation.

The Versatec presently allows no control over the label size on the output, so the surface numbers may get jumbled if several are in the same area. This will be changed whenever the size feature is available. If you don't want the surface numbers, set the Keyword LABEL (or just L) to 0:

```
DEV=VERS L=0
```

Warning: Because the Versatec output is a highly magnified view of what you see on a Tektronix screen, MCNP significantly reduces the plotting step size for good resolution - otherwise lines wouldn't meet, etc. This causes running time to increase substantially to frequently two or three minutes of CPU time per Versatec plot. However, you shouldn't be sending everything to the Versatec anyway, but only one or two final views after a lengthy interactive plotting session at the Tektronix.

C. $PLOT$:

The PLOT overlay produces a file $PLOT$ which is a binary file containing arrays of xy-coordinates of plotted points for each view made. There can be a maximum of fifty views. At the end is a 51-word directory

that points to the beginning of each array. This file can be used to display plots on graphical devices not supported by MCNP or used by installations other than at LASL. The user will have to supply his own processing code to read $PLOT$ and then create the necessary input for his particular device.

D. Use of BASIS Vectors in VIEW Option:

The ORIGIN, EXTENT, and BASIS vectors are all used to define the space called the plot window (in particular, the window appears on the Tektronix screen). The window is a rectangular plane twice the length and width of EXTENT and is centered about the point defined by ORIGIN. The first BASIS vector (called B_1) is a vector along the horizontal axis of the plot window and points toward the right side of the window. The second BASIS vector (called B_2) is a vector along the vertical axis of the plot window and points toward the top of the Tektronix screen.

To find the signs of the BASIS vector components, choose the cross-sectional view you wish to plot. The vector chosen as B_1 will point toward the right side of the screen, and B_2 will point toward the top of the screen. The signs are determined by the direction of the vectors; in particular, do the vector components point in the ±x, ±y, or ±z direction? After signs have been fixed, determine the magnitudes of the vector components.

Assume the vector is parallel to the x-axis. It has no y-component and no z-component so the vector would be 1 0 0. If there is no x-component but both y and z, and y and z have equal magnitudes, the vector would be 0 1 1. The vector does not have to be normalized. If the angle between the vector and the axes is known, you can use the sine and cosine of the angle to determine the magnitude of the components. A rough approximation will probably be sufficient.

The following examples illustrating the BASIS vector cannot be done with the shorthand notations PX, PY, or PZ.

Example 1. Consider the intersecting two-cylinder example (see page 186 and page 306). The radius of the cylinders is 60 cm., and the disk is 20 cm. thick and 30 cm. in radius. The center of the lower face of the disk is at (x,y,z)=(0,0,1500). Assume you want a plot plane parallel to the x-axis and across the corner where the cylinders intersect (represented by the dotted line in Figure B.1). B_1, the vector in the horizontal

APPENDIX B
PLOT

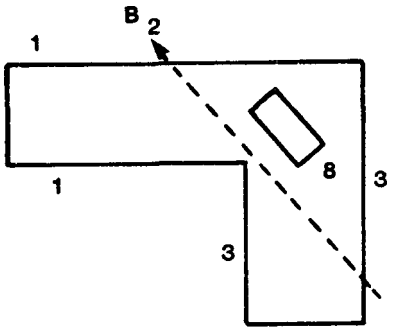


Figure B.1

direction of the plot window, is coming out of the paper parallel to the x-axis. The sign of the x-component is positive since the vector is pointing in the +x direction. There is no y nor z-component. Therefore B_1 is 1 0 0. B_2, the vector in the vertical direction of the plot window, is chosen to point in the direction of the arrow on the dotted line in Figure B.1. There is no x-component. The sign of the y-component is negative since the vector is pointing in the -y direction. The sign of the z-component is positive because the vector is pointing in the +z direction. The magnitude of the y and z-components can be 1 because they are equal. If the angle of rotation of the vector was not 45°, the y-component would be entered as $-\sin\theta$ while the z-component would be $\cos\theta$. B_2, therefore, is 0 -1 1 or 0 $-\sin\theta$ $\cos\theta$. ORIGIN defines the (x,y,z) coordinates of the center of the plot window. The amount of the geometry displayed is determined by EXTENT. The length along the horizontal axis of the plot window is twice EXTENT(1) and the length along the vertical axis is twice EXTENT(2). If EXTENT(2) is omitted, it is assumed to be the same as EXTENT(1).

Figure B.2a shows a plot with the origin to the left of the disk (which is at the intersection of the two cylinders) at 0 0 1475. In Figure B.2b, the origin is located in the disk at 0 0 1500 so the cylinderical portion of the disk is shown. The origin in Figure B.2c is to the right of the disk at 0 0 1529.

Example 2. Assume a plot plane is drawn parallel to the x-axis. B_1 will be the same as in Example 1. B_2, the vector pointing to the top of the plot window, is chosen to point in the direction of the arrow in Figure B.3. There is no x-component. The signs of the y and z-components are both positive since they point in the +y and +z directions, respectively. The magnitude is again equal so B_2 is 0 1 1 or 0 $\cos\theta$ $\sin\theta$ if B_2 is not at

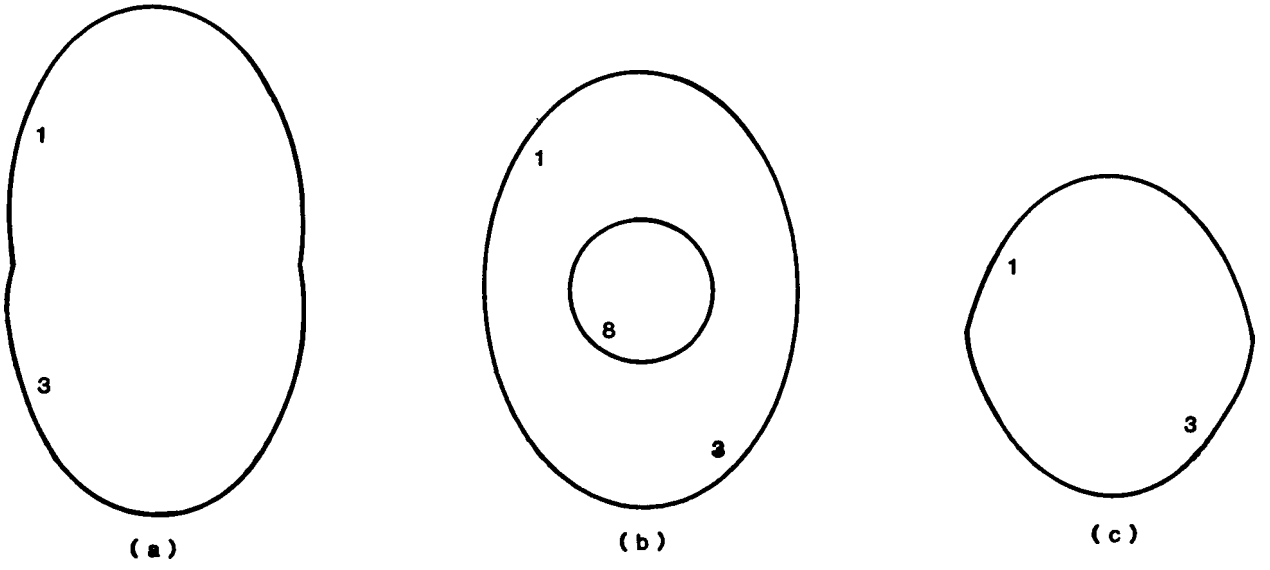


Figure B.2

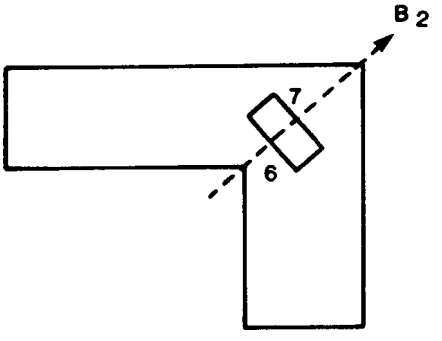


Figure B.3

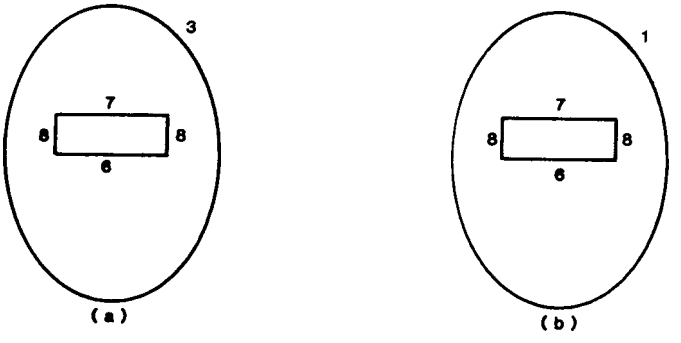


Figure B.4

45°. Figure B.4a shows a plot with the origin at 0 0 1499, below the line of intersection of the two cylinders, so surface 3 is the outside surface.

In Figure B.4b the origin is 0 0 1501, above the line of intersection, and

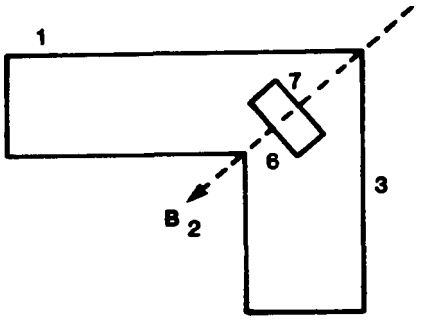


Figure B.5

surface 1 is the outside surface.

Example 3. This example (Figure B.5) is similar to the one of Figure B.4b except the direction of B_2 is reversed. The signs of the y and z-components of B_2 are negative because the vector points in the -y and -z

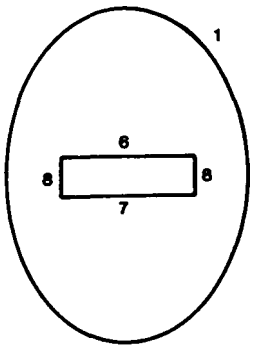


Figure B.6

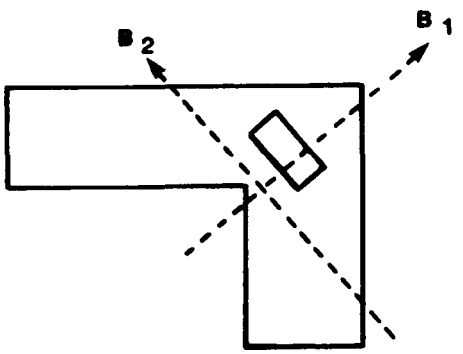


Figure B.7

direction. Note that the surfaces of the disk are reversed in Figure B.6.

Example 4. Assume two plot-window vectors drawn as illustrated in Figure B.7 with the origin at 0 0 1400. B_1 has no x-component; the y and z-components are both positive of equal magnitude. B_2 has no x-component; the y-component is negative, and the z-component is positive. The y and z-components are again of equal magnitude. The entries for BASIS are 0 1 1 0 -1 1. B_2 will point toward the top of the plot window as seen in Figure B.8. To rotate the geometry 90° counter clockwise, enter BASIS

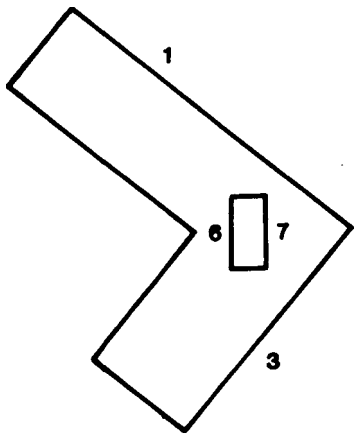


Figure B.8

vectors of 0 1 -1 0 1 1.

E. Use of BASIS Vectors in MOVE Option:

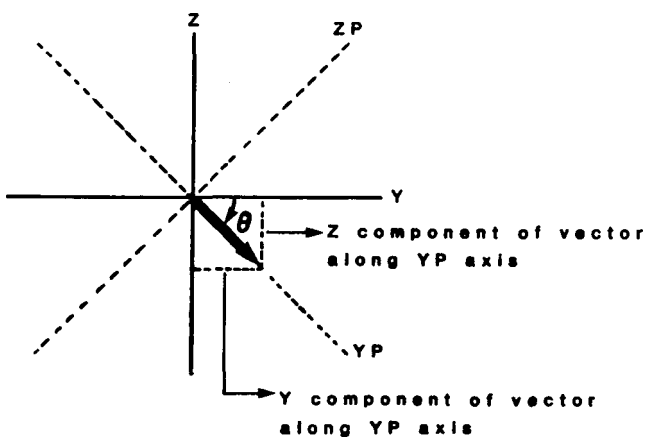


Figure B.9

Figure B.9 illustrates the default case where MAP=1. The input surfaces are in the (x,y,z) system and the output surfaces are in the primed (xp,yp,zp) system. A vector is drawn along the yp-axis as shown. The x-component of the vector is 0. The y-component is cos θ and the sign is positive because the component points in the +y direction. The z-component is sin θ and the sign is negative because it points in the -z direction. Therefore, BASIS(I=1,3) is 1 0 0 and BASIS(I=4,6) is 0 cosθ -sinθ.

If the primed coordinate system were rotated counter clockwise θ

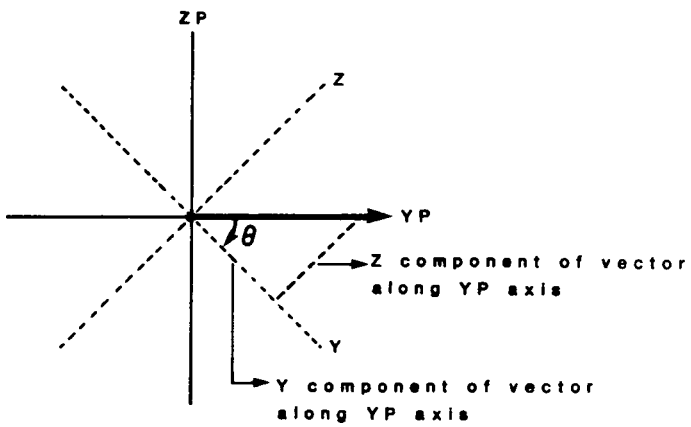


Figure B.10

degrees, BASIS(I=4,6) is 0 cosθ sinθ.

Figure B.10 illustrates the case where MAP is -1. The input surfaces are now in the (xp,yp,zp) system and the output surfaces are in the (x,y,z) system. Again a vector is drawn along the yp-axis. The x-component of the vector is 0. The y-component is cos θ and is positive because it points in the +y direction. The z-component is sin θ and is also positive. Therefore, BASIS(I=4,6) is 0 cosθ sinθ.

If the unprimed system is rotated counter clockwise θ degrees, BASIS(I=4,6) is 0 cosθ -sinθ.

Example: This example illustrates generating surface cards for a rotated disk away from the origin of the coordinate system by first specifying the disk at the origin parallel to an axis and then using MOVE.

This procedure was used in specifying the geometry of Example 10 on page 186 and in the preceding section of this Appendix.

The disk consists of a cylinder bounded by two planes. The cylinder has a radius of 30 cm and is 20 cm thick. The center of the bottom plane of the disk is to be located at z=1500 and rotated 45° clockwise. Initially an input file is set up consisting of two cells (the disk and the outside world) and the three surfaces of the disk. The 30-cm cylinder is centered about the z-axis, and one plane of the disk is at z=0 and the other at z=20. The complete input file is

```
DISK TO BE MOVED
1       0 -1 2 -3
2       0  1:-2:3

1       CZ 30
2       PZ 0
3       PZ 20

IN      0 0
```

To see the simple geometry as described, execute PLOT and type in

```
PX=0 EX=35
```

At the next prompt select the MOVE option with the following parameters:

```
MOVE O=0 0 1500 B=1 0 0 0 .70710678 -.70710678
```

This takes advantage of the default MAP=1 and the diagram of Figure B.9. An alternate input for BASIS is 1 0 0 0 1 -1 since the vectors do not have to be normalized. MCNP will create a file called SURFACE in your local file space that contains new surface cards at the new location for all the surfaces in the original input file. For this example, the SURFACE file is

```
1 GQ  1.000000000000e+00  5.000000000000e-01  5.000000000000e-01
      0.                  -1.000000000000e+00  0.
      0.                   1.500000000000e+03 -1.500000000000e+03
      1.124100000000e+06
2 P   0.                   7.071067811865e-01  7.071067811865e-01
      1.060660171780e+03
3 P   0.                   7.071067811865e-01  7.071067811865e-01
      1.080660171780e+03
```

MCNP rotates the geometry about the point (0,0,0) *only*. If the simple disk were centered at the origin so that the two planes were at z=10 and −10, MOVE would not produce the desired placement of the center of the bottom plane of the disk at z=1500 with what might appear to be an equally correct MOVE input line of

```
MOVE O=0 0 1510 B=1 0 0 0 1 -1
```

The surface cards from SURFACE can be copied into the input file for the rest of the geometry of the two intersecting cylinders with the minor modification that the surface numbers have to be changed to correspond to the numbers in the complete geometry.

Appendix C

Definition of COMMON and Some Useful Arrays

The first part of this appendix is a dictionary of items in the various MCNP COMMON blocks. The second part describes some important arrays in MCNP whose elements have separate definitions.

The following arrays are defined in the next few pages:

APPENDIX C
COMMON

COMMON DICTIONARY

The MCNP COMMON is divided into several sections: *fixed* which is constant after initial values are set in IMCN, *variable* in which the contents change during the course of a run and which is needed for a continue-run, *ephemeral* which is not needed for a continue-run, PBL that goes to the bank, tables of data, two LCM blocks, and finally a block for the PLOT overlay. The variables of the various COMMON blocks are merged and listed alphabetically in the following table. Some of the arrays, such as TRACE, are described in more detail in the second part of this appendix.

AAA	First word in variable Common
AID(10)	Problem ID
AID1(10)	Problem ID of continue-run
AJSH	Coefficient in the formulas for the surface area of a torus
ALSTAR	8H********
AMFP	Mean free paths from collision to detector
ANG(5)	Describes angle between surface normal and particle direction
AREA(JMAX)	Surface areas used for tallies
AREAS(2,JMAX)	Areas of surfaces calculated by MCNP
ATSA(2,MP9)	Tally segment volumes or areas
AWTAB(2,159)	Atomic weight ratio table
BBB(3,3)	Rotation matrix in volume calculator
BUF2(1028)	Input buffer Unit 2
BUF4(1028)	Output buffer Unit 4
CHCD	User charge code
CMP(NDET)	OMCFE collision multiplicity
COLOUT	See description of COLOUT array
CP1	Computer time consumed after beginning MCRUN
CPO	Computer time consumed up to MCRUN
CRT(MAXE)	Cumulative total cross section for nuclides or elements in a cell
CTME	Computer time cutoff
CTS	Computer charge, dump 1 to current dump
DAS(1)	Dynamically allocated storage area
DDET	Distance from collision point to detector

DDG(2)	Detector and DXTRAN diagnostics, λ_{max} and m
DEN(MAXA)	Cell gram densities
DLS	Distance to next surface
DOM(2,2,4)	Domain limits for plotting
DTI(100)	1. Distances to Godfrey cell surfaces in MCRUN 2. Array for command numbers in PLOT
DUMSP(MP6)	Spacing array in DAS
DXCP(2,MAXA)	DXTRAN cell probabilities for neutrons/photons
DXD(2,22,5)	Neutron/photon diagnostic tables (DXTRAN)
DXW(2,2)	DXTRAN neutron/photon weight-cutoff parameters
DXX(2,5,5)	DXTRAN neutron/photon sphere parameters
EAA	Average source energy
EBL(13)	Photon energy bins
EBR	Thermal energy cutin
ECF(2)	neutron/photon energy cutoff
EFS(35)	Boundaries of equally probable fission energy bands
EGO	Energy before collision, and also used to store mean free path in forced collisions
EMCF(2)	Neutron and photon energy thresholds for analog capture
EMX	Maximum neutron energy for cross sections
ERG	Particle energy (MeV)
ESPL(2,10)	Energy splitting data
ETH(MAXA,MAXT)	Cell thermal temperatures
FEBL(13)	Weight of photons generated, each of 13 bins
FIM(2,MAXA)	Neutron/photon cell importances
FIS	Importance of source cell
FME(MEMAX)	Atomic fractions of elements on material card
FPI	1/NPS
FRM(MXMT)	Material card data
FSO(1)	KCODE fission source array in DAS
GWT(MAXA)	Minimum gamma production weights
IA	Program number of cell in which track is located
IAP	New cell into which particle has crossed
IBNK(1)	Array for banked particles in DAS
ICRN(3,MP9)	Names of 2 surfaces forming corners, and corner label
IDET	Current detector index
IDMP	Dump number for continue-run

APPENDIX C
COMMON

IDTM(3)	Machine/date/time of run
IDUMMY(50)	Integer array for user, IDUM card
IDX	DXTRAN sphere number
IET	S(α,β) thermal nuclide index
IEX	Index of nuclide involved in last collision
IFIP	Non-zero if IP card used
IFL(MAXA)	1. (IMCN) Material numbers on cell cards
	2. (MCRUN) Cell tally flagging node
IGROSS	Set to 1 for maximum field length (LCM)
IJK	Starting random number of current history
IKZ	Number cycles to skip before tallying (KCODE)
ILN	Input data line counter
IMES(30)	Symbols from execute-line message
IMTX(MXMTX)	List of S(α,β) identifiers for each MT card
INDT	Count of MT cards plus MT card entries
INK	Flag used for PRINT feature
IOVR	Overlay index
IPAC2(MAXA)	0/1 if track has not/has previously entered cell
IPAN(MP1)	Sum of nuclides over cells
IPM	Counter for tally parameters
IPNT(2,16,NTALMX)	Array in DAS for reading in tally cards
IPSC	Scattering probability index, see PBL array
IPST(1)	Equivalenced to PST
IPTAL(IPTALJ,NTALMX)	Array for tally descriptor information in DAS
IRTP(1)	Equivalenced to RTP
ISABID(MAXE)	List of different S(α,β) material entries
ISOTT	Number thermal nuclides
ISRP(7)	Processed source parameters from SRC card
ISSAB	Count of live entries in ISABID
IST	Pointer for KCODE source block FSO
ISUB(NDEF)	Default file names
ISXSN(1)	Equivalenced to SXSN
ITAL	Tally index
ITDS(1)	Equivalenced to TDS
ITERM	Type of computer terminal
ITFXS	Flagged if problem needs total fission cross section
ITI(100)	Array for command keywords in PLOT
	Surface numbers for DTI distances in MCRUN
ITOTNU	Total fission $\bar{\nu}$ flag
ITRACE(NPREF,MAXE)	Equivalenced with TRACE

ITTFG(MAXE)	Index for nuclides in TCC (thermal treatment) array
IXAK	Pointer for KCODE source block FSO
IXRE	Cross section reaction index
JA	Program number of surface upon which track is located; JA=0 if none.
JAP	Next surface
JEV	Number of printed events in current history
JFL(JMAX)	Surface tally flag
JOVR(NOVR)	Overlay flags
KCSF	Flag to indicate if new source has already overridden old source (KCODE)
KCT	Number cycles performed before problem ends (KCODE)
KCY	K generation or cycle
KDB	Way the particle got lost
KFL	Flag for cell or surface tally flagging(0=none)
KLIN(8)	Data input line that goes to ERPRNT
KLINO(8)	Each data input card read into this array
KNOD	Dump number
KNTRLEE	Flags MCNP as controlee
KODVER(3)	Code and version identification
KOLOUT(1)	Equivalenced to COLOUT
KONRUN	Continue-Run flag
KOVR(NOVR,2)	Names of overlays
KPROD	Flag for production status
KRFLG	Event printing flag: 0 for no printing, 1 specified for debugging, 2 for lost particle
KSF(29)	Surface type symbols
KSR	Number of seconds before job time limit
KST(JMAX)	Surface type (1-24). Reflecting surfaces are negative
KUFIL(4,6)	User-file information
KX1	Constant used in index expressions for XSN & IXSN = 1
KX2	Constant used in index expressions for XSN & IXSN = 2 (see page 340 about large FTN arrays)
KZKF	KCODE flag for when to start tallying
LAJ(MP3)	Cells on other side of the surfaces in LJA
LBNK	Location of the bank in DAS
LCA(MP1)	Location in LJA of 1st surface entry for Cell IA
LCAJ(MP2)	Location in LJA of 1st cell other side of surface LJA(IA)

LCMS(100)	Scratch array in LCM
LCP(MAXE)	Element locator for photon cross-section tables in XSP
LECP(MAXE)	Index for PCR block for each nuclide
LFATL	If equal to 1, run problem with fatal errors
LFC	Length of fixed Common
LFCL(MAXA)	Fissionable material cell flag
LFL	Thermal treatment flag
LFLL	LCM field length
LFS	Length of KCODE source in DAS
LFSO	Location of KCODE source in DAS
LGC(100)	Boolean coefficients for present Godfrey cell
LIBXS(MP8)	Neutron cross-section library file names
LJA(MP2)	Surfaces with senses and logical operators
LJAV(JMAX)	List of cell descriptors in volume calculation
LJSV(JMAX)	List of surfaces in volume calculation
LKTP	Length of KCODE source file
LME(MEMAX)	List of nuclide locators for each cell
LMET(MEMAX)	List of photon locators for each cell
LOCCT(MAXA)	Location of cell tallies
LOCDT(2,MXDT)	Location of detector tallies
LOCST(JMAX)	Location of surface tallies
LOST(2)	Control for lost particles
LPBB	Number of words required for each particle in bank
LPBD	Number of words in PBL to be restored after point detector calculations
LPL	Length of tally pointers and descriptors in DAS
LSAT(NTALMX)	Location in ATSA array of segment areas and volumes in volume calculation
LSC(MP4)	Pointers to surface coefficients
LSS	Length of summary information
LTAL	Location of tally blocks in DAS
LTPE	Length of the RUNTPE file
LUF	Length of user-file buffers in DAS
LUFB	Location of user-file buffers in DAS
LVB	Length of variable COMMON backup in DAS
LVBU	Location of variable Common backup in DAS
LVC	Length of variable Common
LXN	Length of neutron cross sections in DAS
LXP	Length of photon cross sections in DAS

LXSN	Location of neutron cross sections in DAS
LXSP	Location of photon cross sections in DAS
MCD(MAXA)	Cells in which OCMFE detectors are located
MCTL(10)	Default MCT execution message
MCTM(20)	MCT card message
MEMORY	Total available machine memory
MIX	Number of items in FRM
MKE	Number of nuclides in the cell
ML1(MAXA)	First word address FME-LME blocks each cell
ML2(MAXA)	Last word address FME-LME blocks each cell
MMAT	Count of nuclides
MODE	Problem Mode
MSRK	Maximum number of source points (KCODE)
MXA	Number of cells
MXAFM	Number of cells and pseudo-cells for FM card
MXAFS	Number of cells and pseudo-cells for FS card
MXE	Number of different nuclides used
MXF	Total number of tally bins in DAS
MXF2	2*MXF
MXF3	3*MXF
MXJ	Number of surfaces
MXS	Number of source energy entries on SERG card
MXT	Number of thermal times on THTME card
NBHWM	Maximum number ever in bank
NBMX	Maximum bank particles before overflow to BANKFL
NBNK	Number of particles in bank
NBOV	No. source particles causing bank overfow to BANKFL
NBT(2)	Total number of neutrons/photons banked this problem
NCH(2)	Number of neutron/photon collisions so far in the current history
NCL(MAXA)	Problem number of cells
NCRN	Number of corners for a cell in volume calculator
NCT(2)	Total number collisions (neutron/gamma)
NDBG(20)	Debugging controls array
NDET	Total number point, ring, and OMCFE detectors
NDM	Dump cycle
NDP	Print cycle
NDRR(MAXE)	List of discrete reaction rejections
NDX(2)	Number of neutron/photon DXTRAN places
NERR	Number of lost particles

NETB(2)	Count of times neutron energy greater than EMX
NFER	Number fatal errors found by IMCN and XACT
NLAJ	Number of otherside cells on cell cards
NLJA	Total number of surfaces and logical operators on cell cards
NLT	Number of distance-to-surface roots for Godfrey cells
NODE	Track branching flag - see PBL(18)
NORDER	Indicator of whether the controller is ORDER
NOSC	Count of otherside cells in input
NPA	Number of tracks in same bank location
NPCLUN	Jerks per gram flag for photon summary print
NPP	Source particle cutoff
NPS	Number of particles started
NRN	Number of random numbers generated
NRPZ	Number times RPZERO used in torus calculations
NSA	Number of source points not yet started (KCODE)
NSCF	Number surface coefficients
NSEQ(MAXA)	Number of pieces of cell
NSF(JMAX)	Problem number of surfaces
NSJV	Number of cell descriptors in LJAV
NSR	Source identifier
NSRCK	Nominal source size for cycle (KCODE)
NSRC	Number of source coefficients
NSS	KCODE number source points stored
NST	Termination time flag
NSV	Number of surfaces in LJSV
NTAL	Number of tallies
NTBB(NTALMX,3)	Counter for tallies beyond last time bin
NTC	Control variable for time check
NTC1	second control variable for time check
NTER	Particle termination type
NTII	Used in processing 10-second time interrupts
NTPE	Current disk address of RUNTPE file
NUMF(2,MAXA)	Number neutron/photon forced collisions per cell
NWER	Number of warning errors
OSUM(3)	Sums used to calculate averages of three k_{eff} estimates (KCODE)
OSUM2(3,3)	Sums used to calculate covariance (KCODE)
PAC(2,10,MAXA)	Summary of neutron/photon activity each cell
PAN(2,3,MP7)	Summary information for activity of neutrons/photons

PAX(2,6,9)	Neutron/photon summary of particle creation and loss
PBL	See description of PBL array
PBLSAV(MXPD)	Temporary storage for particle quantities in detector routines
PCC(3,MAXA)	Photon creation summary
PCR(6,MAXE)	TRACE array for photons
PIE	3.1415926535898
PLE	Macroscopic cross section of present cell
PMF	Distance to next collision
PSC	Probability density function for scattering toward a detector
PST(1)	Array for PLOT information in DAS
PTAL(IPTALJ,NTALMX)	Equivalenced to IPTAL
PWB(2,13,MAXA)	Summary of neutron/photon weight balance each cell
QAX(2,MAXA)	Neutron/Photon path length stretching parameter for cells
QPL	Adjusted macroscopic cross section for exponential transformation
RDUMMY(50)	Real array for user, RDUM card
RHO(MAXA)	Cell atom densities
RKK	Initial guess for k_{eff} (KCODE)
RKS(1)	Temporary storage for KCODE source in DAS
RLT(2)	Removal lifetime by absorption and track length, (KCODE)
ROM	Radius of sphere around OMCFE detectors
RSCRN(2,MP9)	R and S coordinates of cell corners
RSUM(2)	Sums used to calculate two estimates of removal lifetime (KCODE)
RSUM2(2,2)	Sums used to calculate covariance of lifetime (KCODE)
RTP(MP6)	Temporary storage area for tally card data in DAS
SCF(MP5)	Surface coefficients
SCFQ(5,JMAX)	Q-form surface coefficients for volume calculation
SDL	Distance transversed by pseudo particle scoring at a detector
SDW	Constants for testing for word length truncation
SEF(MAXS,3)	Source data from SERG, SPROB, and SBIAS cards
SMCT	Interval between executions of MCT
SMUL(2)	System multiplication of particles

APPENDIX C
COMMON

SMUL1	System multiplication of current history
SRC(MAXS)	Raw source parameters from SRC card
SRP(20)	Processed source parameters from SRC card
SPARE1, SPARE2, SPARE3	Spares for user-supplied bank data
SUMK(3)	Total weight of fissions in a generation
SVWT	Weight of source particle used for producing photons
SWTM	Minimum source particle starting weight
SXSN(1)	Array for neutron cross sections; note XSN(I)=SXSN(I+1)
TAL(1)	Tally array in DAS
TCO(2)	Neutron/photon time cutoffs
TDS(1)	Array for descriptors of tallies in DAS
TMAX(2,3)	Average particle lifetime in shakes
TMCT	Time of last MCT tape dump
TM0	Presently unused
TME	Time in shakes
TOTM	Total microscopic cross section
TPD	See description of TPD array
TPP(20)	Temporary scratch storage array
TRACE	See description of TRACE array
TTC(5,NTTC)	Thermal treatment constants of low-Z materials
TTH(MAXT)	Times for thermal temperatures
TTTID(MAXE)	Thermal isotope ZAID numbers
U	X direction cosine of particle velocity
UFB(1)	Buffers in DAS for user supplied files
UOLD	Former value of U
V	Y direction cosine of particle velocity
VBU(1)	Variable Common backup array in DAS
VCO(55)	Form factors for photon scattering
VEL	Track velocity (cm/shake)
VIC(21)	Form factors for photon scattering
VOL(MAXA)	Cell volumes
VOLD	Former value of V
VOLS(MAXA)	Array for volumes of cells calculated by MCNP in DAS
W	Z direction cosine of particle velocity
WC1(2)	Neutron/photon first weight cutoff
WC2(2)	Neutron/photon second weight cutoff

WCO(55)	Used in form factors for photon scattering
WGT	Track weight
WOLD	Former value of W
WTO	Weight of each KCODE source point in current cycle
X	X-coordinate of track in geometry
XLIST(MAXE)	Ordered ZAID identifier list
XSDR(MAXE)	Nuclides for discrete reaction treatment
XSP(1)	Photon cross-section array in DAS
Y	Y-coordinate of track in geometry
Z	Z-coordinate of track in geometry
ZERO	1.0E-8
ZZEOF	Last word in variable Common
ZZL1	Last word in summary information
ZZZ	Last word in backed up variable Common

APPENDIX C
Arrays

TRACE(NPREF,MAXE) Array (NPREF = 40, MAXE = 40)
EQUIVALENCE (TRACE,ITRACE)

This array is loaded in the XACT overlay and used in the running overlay. For each problem nuclide, IEX, this array contains identifiers (TRACE(I,IEX),I=1,2) and pointers (TRACE(I,IEX),I=3,23) for the neutron cross-section data. But unlike other cross-section arrays, some words (TRACE(I,IEX),I=24,40) are variables in the running overlay. These words contain key cross-section data at the energy ERG for which the cross section was last looked up. Thus when a cross section is needed at the same energy more than once the time-consuming process of fetching its value is only required the first time; substantial computer time is saved by getting the values from the TRACE array all subsequent times until the energy changes. The entire TRACE array, which is also listed in Appendix E, is described here for nuclide IEX.

It should be noted that the locators in the ACE libraries are with respect to the beginning of the nuclide's table, but in MCNP they end up being indexes of XSN.

I	TRACE(I,IEX)	Description
1	ZAID	nuclide identifier number
2	AWR	atomic weight ratio
3	NTR	total number of reactions excluding elastic
4	NES	number of energies
5	NR	number of nonelastic reactions having secondary neutrons NR ≤ NTR
6	ESZ	loc. of energy table
7	MTR	loc. of ENDF MT's
8	TYR	loc. of reaction types
9	LSIG	loc. of table of cross-section locators
10	SIG	loc. of cross sections
11	LAND	loc. of table of cosine distribution locators
12	AND	loc. of cosine tables
13	LDLW	loc. of table of law locators
14	DLW	loc. of law data
15	GPD	loc. of gamma production data. 0 if none
16	LQR	loc. of reaction Q's
17	MGPT	type of neutron data set: ≥0, continuous energy; -1, discrete reaction; -2, S(α,β)

18	NU	loc. of fission ν data. 0 if no fission
19	FIS	loc. of total fission cross section. 0 if no fission
20	END	loc. of last word of this data set
21	-	unused
22	-	unused
23	-	unused
24	-	unused
25	σ_t	total cross section at energy ERG without thermal effects (as taken from ACE format input data)
26	ERG	energy prior to collision, LAB
27	σ_t	total cross section at energy ERG with thermal effects included
28	IERG	i such that $ES_i \leq ERG < ES_{i+1}$ (ES_i = energy table energy. See ESZ block)
29	RINT	interpolation fraction: $(ERG-ES_i)/(ES_{i+1}-ES_i)$
30	EHD	variable in thermal procedure
31	σ_a	absorption cross section at energy ERG with thermal effects included
32	σ_{el}	elastic cross section at energy ERG with thermal effects included
33	σ_f	total fission cross section at energy ERG without thermal effects
34	-	$S(\alpha,\beta)$ quantity
35	-	unused
36	-	unused
37	LTNN	thermal collision parameter
38	-	$S(\alpha,\beta)$ quantity
39	$\bar{\nu}$	average number of neutrons per fission at energy ERG
40	-	presently unused

PCR(6,MAXE) Array (MAXE = 40): Photon TRACE Array

This array is the "TRACE" array for photons and is used only in the running overlay. Analogous to the TRACE array for neutrons, the elements of the PCR array contain the photon cross sections at the energy ERG for which they were last locked up. Thus when a photon crosses a boundary, is fetched from the bank, etc., the time-consuming process of looking up and/or computing the cross section is avoided if the energy is unchanged. This feature has decreased the MCNP running time for some problems by as much as a factor of 5. The PCR array for nuclide IE is

I		PCR(I,IE)
1	σ_{inc}	where σ_{inc} = incoherent cross section at energy ERG
2	$\sigma_{inc} + \sigma_{coh}$	where σ_{coh} = coherent cross section at energy ERG
3	$\sigma_{inc} + \sigma_{coh} + \sigma_{fl}$	where σ_{fl} = fluorescence cross section at energy ERG
4	$\sigma_{tot} = \sigma_{inc} + \sigma_{coh} + \sigma_{fl} + \sigma_{pp}$	where σ_{pp} = pair production cross section at energy ERG and σ_{tot} = total cross section at energy ERG
5	H_p	photon heating at energy ERG (MeV/collision)
6	ERG	energy at which above data are fetched/computed

COLOUT(60) Array: Collision Description

EQUIVALENCE (COLOUT,KOLOUT)

The COLOUT array appears only in the running overlay and it describes the output of a collision. The elements of this array are

I	COLOUT(I)
1	unused
2	unused
3	TYR = reaction type. For neutrons, this is the value for this collision reaction from the TYR block in the cross section tables. TYR < 0 for center of mass scattering and TYR > 0 for lab system scattering. TYR = ± 99 for an elastic collision, ± 18 for type 18 fission collision, ± 19 for type 19 fission, or 0, ± 1, ± 2, ± 3, ... for 0,1,2,3 ... neutrons out. For photons, TYR = 1,2,3,4,5 for incoherent, coherent, single fluorescent, double fluorescent or pair production scattering, respectively
4	ENDF MT reaction number for neutrons. KOLOUT(4) is set but never used.
5	Real number of emergent neutrons from a neutron collision. COLOUT(5) is not used for photons
5*N+1	emergent energy of N^{th} particle in lab system
5*N+2	emergent cosine of N^{th} particle in lab system
5*N+3	emergent energy of N^{th} particle in CM system (given only if TYR < 0)
5*N+4	emergent cosine of N^{th} particle in CM system (given only if TYR < 0)
5*N+5	location in cross-section data of the appropriate equiprobable cosine bin distribution table. (0 for isotropic scattering)

APPENDIX C
Arrays

IPNT(2,16,NTALMX) Array (NTALMX=41)

This array is used only in the initiation overlay for reading in tally cards.

IPNT(1,1,K)=n	= tally number on Fn card for k^{th} tally
IPNT(2,1,K)	= 1,2,3,4,5,-1,-2,-3,-4,-5/F,FX,FY,FZ,FW,*F,*FX, *FY,*FZ,*FW = tally type for k^{th} tally
IPNT(1,J,K),J=2,16	= starting location in RTP array of data on tally card J for the k^{th} tally
IPNT(2,J,K),J=2,16	= last location in RTP array of data for tally card J for the k^{th} tally
J=2	F,FX,FY,FZ,FW card
3	E card
4	T card
5	C
6	FS
7	FC
8	CF
9	SF
10	FM
11	VA
12	EM
13	TM
14	PD
15	FU
16	CM

Example: Suppose the third tally in the input deck is

```
*FY45   6   4   1
 FM45   1   2   16
```

Then the tally is loaded as

IPNT(1,1,3) = 45
IPNT(2,1,3) = -3
(RTP(L), L=IPNT(1,2,3),IPNT(2,2,3)) = 6,4,1
(RTP(L), L=IPNT(1,10,3),IPNT(2,10,3)) = 1,2,16

IPTAL(IPTALJ,NTALMX) Array (IPTALJ=25,NTALMX=41)
EQUIVALENCE (IPTAL,PTAL)

This array is used in the initiation and main transport overlays to identify tallies, point to tally descriptions, and points to tally storage. The tallies are ordered ITAL=1,NTAL in order of how they are printed out (i.e., neutron tallies first, photon second). For tally ITAL, the values in IPTAL(M,ITAL) are

M	IPTAL(M,ITAL)
1	Location of word before tallies in TAL array
2	Location of energy bin array in TDS array (E card)
3	Location of time bin array in TDS array (T card)
4	Location of cosine bin array in TDS array (C card)
5	Location of surface/cell/detector array in TDS array (F card)
6	Location of time bin multipliers in TDS array (TM card)
7	Location of energy bin multipliers in TDS array (EM card)
8	Location of reaction information (FM card) + 1
9	Tally type (1,2,4,5,6,7), i.e., F5 is a type 5 tally
10	Tally type: 1=neutron,2=photon. Negative if * on F card.
11	Location of cell flagging cell numbers (CF card) if >0; if = 0, neither cell nor surface flagging; -1, surface flagging
12	NE+1=number of energy bins + 1
13	(NT+1)*(NE+1) where NT = number of time bins
14	NC*(NT+1)*(NE+1) where NC = number of cosine bins
15	NU*(NS+1)*NC*(NT+1)*(NE+1) where NS=number of segment bins (FS card) and NU=number of user-supplied bins (FU card)
16	NT+1=number of time bins + 1
17	Location of segment bin divisors (volumes or areas) in TDS array (VA card)
18	Multiplication factor for tally (first entry of FM card)
19	Location of user-supplied tally bins (FU card)
20	Tally number (i.e., n of Fn card)
21	Number of fictitious cell for which FS card data is loaded into the LCA and LJA arrays
22	Location of descriptive comment (FC card)
23	Location of surface flagging surface numbers (SF card). For tally type 5, location of PD card information.
24	NU*(NS+1)*NC=total number of time-energy arrays
25	Location of cosine bin multipliers (CM card)

APPENDIX C
Arrays

SRP(20) and ISRP(7) Source Parameter Arrays

The parameters describing the source are

I		SRP(I)
1	x	(x,y,z) Cartesian coordinates of source point
2	y	location
3	z	
4	w	source particle weight
5	p	sample y-axis direction cosine, v, within the
6	ν	cone $\nu \le v \le$ with probability p
7	w_1	Weight of particles starting in the cone $\nu \le v \le 1$
8	w_2	weight of particles starting in the cone $-1 \le v < \nu$
9	R_1	unused for SRC1 source radius of source surface for SRC2 & SRC3 sources inner source sphere boundary radius for SRC4 source window radius for SRC5 source
10	R_2	outer source sphere boundary radius for SRC4 source unused for SRC1, SRC2, SRC3, SRC5
11	u_3	unused for SRC1, SRC2, SRC3
12	v_3	direction cosines of axis of optional cylinder, SRC4
13	w_3	direction cosines of plane wave, SRC5
14	u_1	Rotation matrix elements for SRC4 cylindrical volume
15	v_1	and SRC5 plane wave sources. Unused for SRC1, SRC2, SRC3.
16	w_1	A point (x',y',0) is chosen uniform in area on the plane
17	u_2	centered at (x_o,y_o,z_o). The source particle starting
18	v_2	coordinates, (x,y,z) are then

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} u_1 & u_2 & u_3 \\ v_1 & v_2 & v_3 \\ w_1 & 0 & w_3 \end{bmatrix} \begin{bmatrix} x' \\ y' \\ 0 \end{bmatrix} + \begin{bmatrix} x_0 \\ y_0 \\ z_0 \end{bmatrix}$$

19	Area	Source SRC5 surface area

I		ISRP
1	IA_1	cell in which source particle is born
2	IA_2	window cell defining window for SRC4 cylinder and SRC5 only
3	JA	surface upon which source particle started (SRC2,SRC3,SRC5 only)
4		unused
5	tries	number of times source coordinates calculated
6	failures	number of times source coordinates not found in proper cell. Note that 1 - ISRP(6)/ISRP(5) = efficiency of source sampling
7	IPSC	index for type of source in order to compute scattering directly to point or ring detector

APPENDIX C
Arrays

PBL(LPBB) Array: (LPBB=MXPB, or LPBB=MXPB+NDET if there are OMCFE detectors; MXPB=22, NDET=number of any type of detectors)
EQUIVALENCE (PBL,X)

Splitting schemes and some collisions produce additional tracks. Because only one track can be treated at at time, the extra tracks are stored in the "bank," the IBNK array. Furthermore, detector calculations require that a description of a track and its last collision parameters be temporarily stored while the detector tally is being made - this is done in the PBLSAV array. To facilitate rapid copying to and from storage, the track parameters that must be saved are consecutive in COMMON and the array PBL is equivalenced to them.

Only items 1 through 22 in the following list are saved in the bank if there are no OMCFE detectors in the problem. If there are OMCFE detectors, items 1 through CMP(NDET) are saved. During detector calculations, items 1 through WOLD are temporarily stored in PBLSAV for non-OMCFE detectors and items 1 through COLOUT(10) for OMCFE detectors.

I	EQUIVALENCE	PBL(I)
1	X	x,y,z coordinates of particle in MCNP cartesian coordinate system.
2	Y	
3	Z	
4	U	direction cosines of the particle velocity
5	V	
6	W	
7	ERG	particle energy (MeV)
8	WGT	particle weight
9	TME	time (shakes = 10^{-8} sec)
10	VEL	particle velocity (cm/shake)
11	EGO	Energy before collision. Also used to store mean free path in forced collisions
12	IA	program number of cell in which particle is located
13	JA	program number of surface upon which particle is located. JA=0 if none.
14	NPA	Number of particles stored in bank. For example, rather than store 4 particles in the bank when a 5-for-1 split occurs, 1 particle with NPA = 4 is banked representing all 4 particles. Also, PBL(14)

		= - distance to next surface in the special case of forced collisions.
15	IEX	index of the nuclide involved in the last collision
16	NFC	number of forced collisions so far during present transversing of cell IA
17	IPT	particle type: 1=neutron, 2=photon
18	NODE	branch number (used in cell and surface flagging for tallies) of each track. NODE=1 for source track, NODE=2,3,4, ... for 2nd, 3rd, 4th tracks of a given history from the bank.
19	IDX	DXTRAN sphere number if particle is a DXTRAN particle; otherwise, zero
20	SPARE1	Spares for user-supplied bank data
21	SPARE2	
22	SPARE3	
23	(CMP(I), I=1,NDET)	Collision multiplicity: number of neutrons emitted from previous collision used by OMCFE detector.
MXDT+23	UOLD	former values of u,v,w direction cosines; in non-OMCFE context, these parameters are also used for other assorted temporary storage.
MXDT+24	VOLD	
MXDT+25	WOLD	
MXDT+26	IPSC	probability of scattering index =1 particle from source SRCn, =2 particle from monodirectional SRC5 source =3 particle from SRC0 (user supplied) source =4 neutron from thermal collision =5 neutron from non-thermal collision =6 photon from coherent scattering =7 photon from incoherent scattering =8 photon from isotropic event =9 neutron from isotropic event

		=10 neutron from non-thermal collision in thermal range
		=11 photon from double fluorescence
MXDT+27	TPD(I),I=1,6	see description of TPD array
MXDT+33	(COLOUT(I), I=1,10)	see description of COLOUT array

Note that the variables X,Y,Z,U,V,W,IA, and JA are the only exceptions to the rule that the names of local variables have 1 or 2 characters and the names of COMMON variables have 3 or more characters.

TPD(6) Array: Free-Gas Model Thermal Neutron Collision Parameters

In a thermal neutron collision using the free-gas model the scattered neutron direction cosines (u'v'w') and energy E' are related to the incident neutron direction cosines (u,v,w) and energy E by

$$E' = \frac{E}{(A+1)^2}\left(\bar{x}^2 + \bar{y}^2 + \bar{z}^2\right)$$

$$u' = \frac{\bar{x}}{\left(\bar{x}^2 + \bar{y}^2 + \bar{z}^2\right)^{1/2}}$$

$$v' = \frac{\bar{y}}{\left(\bar{x}^2 + \bar{y}^2 + \bar{z}^2\right)^{1/2}}$$

$$w' = \frac{\bar{z}}{\left(\bar{x}^2 + \bar{y}^2 + \bar{z}^2\right)^{1/2}}$$

where

$$\bar{x} = u + A\left(\delta u_o + \frac{x}{a} u_t\right)$$

$$\bar{y} = v + A\left(\delta v_o + \frac{x}{a} v_t\right)$$

$$\bar{z} = w + A\left(\delta w_o + \frac{x}{a} w_t\right)$$

$$\delta = \left(1 + \frac{x^2}{a^2} - \frac{2x\mu_t}{a}\right)^{1/2}$$

A = target nucleus atomic weight ratio (atom mass ÷ neutron mass)

x/a = appropriately sampled relative velocity (incident neutron velocity ÷ target atom velocity)

(y_o, v_o, w_o) = center of mass system isotropically sampled direction cosines

APPENDIX C
Arrays

μ_2 = appropriately sampled cosine between target and incident neutron direction-of-flight vectors

(u_t, v_t, w_t) = appropriately sampled target atom direction cosines

Variations of these parameters must be saved in the TPD array for possible later use in a point or ring detector calculation:

I	TPD(I)
1	$A\delta$
2	$E/(A + 1)^2$
3	temporary storage for the quantity $\sqrt{TPD(4)**2+TPD(5)**2+TPD(6)**2}$
4	$u + Axu_t/a$
5	$v + Axv_t/a$
6	$w + Axw_t/a$

PAX(2,6,9) Array Summary of Particle Creation and Loss

The PAX (I,J,K) array contains summary information for neutrons (I=1) and/or photons (I=2) which appears on the output summary page. J=1,6 are the six columns of the summary table and K=1,9 are the nine rows of the summary table. The values of PAX(I,J,K) are

J	PAX(I,J,K)
1	number of tracks created by creation mechanism K
2	weight created per source particle by creation mechanism K
3	average energy created per source particle by creation mechanism K
4	number of tracks lost by loss mechanism K
5	weight lost per source particle by loss mechanism K
6	average energy lost per source particle by loss mechanism K

K	Creation mechanism	Loss mechanism
1	source	escape
2*	-	time cutoff
3*	fission	energy cutoff
4*	(n,xn)	all capture
5	weight cutoff	weight cutoff
6	importance sampling	importance sampling
7	forced collision	exponential transform
8	energy splitting	scattering
9	DXTRAN	DXTRAN

*For photons these are

2	from neutrons
3	fluorescence
4	pair production

APPENDIX C
Arrays

PAC(2,10,MAXA) Array Summary of Problem Activity per Cell

The PAC(I,J,K) array contains information for the printed summary of neutron (I=1) and/or photon (I=2) activity in program cell K. This information is updated whenever a particle enters or collides in a cell. The parameters in the PAC array are

J	PAC(I,J,K)
1	tracks entering cell K
2	population of cell K ≡ number of tracks entering for the first time (re-entrant tracks excluded)
3	collisions in cell K
4	collisions * weight
5	energy * time * weight
6	energy * path length * weight
7	path length
8	mean free path (=1/macroscopic total cross section of all material in cell K) * path length * weight
9	time * weight
10	path length * weight

For the printed summary of problem activity in each cell, K, the listed quantities are:

tracks entering = PAC(I,1,K)
population = PAC(I,2,K)
collisions = PAC(I,3,K)
collisions * weight per history = PAC(I,4,K) ÷ number of histories
number weighted energy = PAC(I,5,K)/PAC(I,9,K)
flux weighted energy = PAC(I,6,K)/PAC(I,10,K)
average track weight (relative) = [PAC(I,10,K) * cell K importance] ÷ [PAC(I,7,K) * source cell importance]
average track MFP = PAC(I,8,K)/PAC(I,10,K)

Note that if a particle becomes lost erroneous information is entered into the PWB array and hence the summary of problem activity in each cell is slightly inaccurate.

PWB(2,13,MAXA) Array Summary of Weight Balance in each Cell

The PWB(I,J,K) array contains information for the printed summary of neutron (I=1) and/or photon (I=2) weight balance in each program cell, K. This information is updated every time weight changes in MCNP. The J=1,12 events summarized in the table and their components are

J	Table Heading	Quantity Stored in PWB(I,J,K)
	External	
1	source	weight of newly created source particle
2	entering	weight of particle entering cell K
3	time cutoff	weight of particle killed by time cutoff
4	energy cutoff	weight of particle killed by energy cutoff
5	exiting	weight of particle exiting cell K
	Physical (neutrons)	
6	fission	weight of new tracks created by fission
7	(n,xn)	weight of new tracks created by inelastic processes
9	capture	weight of neutrons lost to capture
	Physical (photons)	
6	from neutrons	weight of photons created in neutron collisions
7	pair production	weight of photons created by pair production
8	fluorescence	weight of photons created by double fluorescence
9	capture	weight of photons lost to capture

	Variance Reduction	
10	weight cutoff	net weight change caused by weight cutoff
11	importance sampling	net weight change caused by splitting and Russian Roulette in importance sampling
12	exponential transform	net weight change caused by use of exponential transform
13	DXTRAN	net weight change caused by using DXTRAN

Note that if a particle becomes lost, erroneous information is entered into the PAC array and hence the weight balance summary becomes slightly inaccurate.

PAN(2,3,MP7) Array (MP7 = 3*MAXA, MAXA=175)

The PAN(I,J,K) array contains summary information for the activity of neutrons (I=1) and/or photons (I=2) for the (K-IPAN(IA)+1)thnuclide in cell IA. The three types of information stored are

J	PAN(I,J,K)
1	number of collisions with nuclide K-IPAN(IA)+1 in cell IA
2	collisions * weight
3	weight lost to capture

Note that if a particle becomes lost erroneous information is entered into the PAN array and hence this table becomes slightly inaccurate.

PCC(MAXA) Array (MAXA=175) Photon Creation Summary

The PCC(I,J) array contains a summary of photon production in cell J. The photon production information is

I	PCC(I,J)
1	number of photons produced from neutron collisions in cell J
2	weight of photons produced
3	weight * energy of photons

Note that if a photon gets lost it is still considered created in the PCC array and hence this table becomes slightly inaccurate.

FEBL(13) Array Photon Generation Summary

The FEBL(I) array contains the weight of photons created by neutron collisions in all cells in each of I=1,13 photon energy groups. The lower bounds of the thirteen groups are (in MeV) 0, .01, .1, .5, 1, 2, 3, 4, 5, 6, 7, 8 and 9.

APPENDIX D

UPDATING AIDS

This appendix is for the serious updater making extensive changes to MCNP. The Update examples in Appendix A are for the casual updater and should be adequate for most needs, like adding a source subroutine. Beyond that there is not much else we can do because of the large variety of ways the code is altered by users. The serious updater should certainly be familiar with the uses and options in UPDATE, LOD, FTN, and LIBMAK so individual needs can be satisfied regardless of how we have set things up.

The MCNP naming convention for variables, a discussion of dynamic field length adjustment, and a way around the FTN compiler problem with very large arrays are covered in the first part of this Appendix. In the last part, the MCNP tally structure is outlined – this is the most frequently user-modified part of MCNP. Appendix E contains very helpful information if you require special cross-section related data. Finally, some especially important arrays are identified in Appendix C.

A master index of MCNP that shows where variables and subroutines are used throughout the code should be very useful to anyone making modifications. An index MCNPIX is available from MASS under /X6CODE (use the CC. and LONG. options with ALLOUT).

Note that only 4346_{10} extra words are available in SCM.

I. NAMING CONVENTION FOR VARIABLES

The COMMON blocks in MCNP are common to nearly all of the subroutines in the code. This practice substantially simplifies development and maintenance of the code, but it also creates a hazard that anyone who intends to modify the code needs to be aware of. Unless an updater checks the symbol that is to be used for the name of a new local variable against all of the variable names in the COMMON blocks, a COMMON variable may be inadvertently used locally with disasterous results.

Two practices are followed in programming MCNP to make this checking easier. The variables and arrays in each COMMON block are in alphabetical order, and, with only eight exceptions, all COMMON variables and arrays have names which are at least three characters long. The exceptions are X, Y, Z, U, V, W, IA, and JA which are all in the PBL array. Furthermore, all

local variables in MCNP are no more than two characters long. If someone modifying MCNP uses only one and two-character names for new local variables and avoids the eight special names in PBL, there will not be a symbol conflict with anything in COMMON.

II. DYNAMIC FIELD LENGTH ADJUSTMENT

The less memory a job uses, the better it competes with other jobs in a timesharing computer system. MCNP adjusts its LCM field length from time to time during the early part of a run in order to use only as much memory as it has to. The data arrays are laid out to facilitate dynamic field length adjustment, with the arrays used later placed higher in memory than arrays used earlier.

The major sections of memory used by MCNP are in this order, with their approximate octal lengths:

Main (level 1) overlay	72000
Level 2 overlays	max 55000
Misc. small LCM Common Blocks	3000
LCM1 COMMON Block	35000
LCM2 COMMON Block	rest of memory

LCM1 contains the BCD I/O buffers and some of the summary data arrays. During the execution of the IMCN (1c0) overlay, part of LCM2 is used for the KCODE source, for the tally descriptions, and for work space for the volume calculator. During the execution of the MCRUN (4c0) overlay, LCM2 is used for tally descriptions, tally blocks, the KCODE source, the particle bank, backup for variable COMMON, buffers for user-specified files, photon cross sections, and neutron cross sections.

The drop file for MCNP is created large enough for the code at its maximum possible size. At the beginning of the MCRUN (4c0) overlay it is trimmed to the size needed for the problem and is not changed thereafter. Immediately after creating the drop file, MCNP increases its LCM field length to the end of the VOLUME work space in LCM2, which is sufficient for the needs of the main overlay and of the level 2 overlay IMCN (1c0). If the PLOT overlay is used, it sets the field length to the end of a part of LCM2 which is used for the buffer for the SURFACE file and for storage of points to be plotted. If XACT is used, it sets the field length to the end

of a part of LCM2 which is used for buffers for the neutron cross-section tables.

In a continue-run, only the IMCN and MCRUN overlays are involved. IMCN sets the field length long enough to contain all the cross sections, and MCRUN does not change it.

If you modify MCNP, make sure that any LCM COMMON blocks you add are loaded before LCM2. This is accomplished by putting the name of the new COMMON block before LCM2 in the list of LCM COMMON blocks on the *LD line in the file of LOD commands (MCNPLF). If it is not used in the main overlay (0c0), you will have to put its name and length in the *LD line in the way that WINDBUF, for example, is done. Beware of inadvertently adding an LCM COMMON block as a result of calling a new FTNLIB subroutine. If a new FTNLIB subroutine declares an LCM COMMON block, you must also specify it on the *LD line before LCM2.

If for some reason you must add an LCM COMMON block after LCM2, use the GROSS. option in the execution line message of MCNP and do something about the length of LCM2. This option causes the LCM field length to be the same as the initial length of the drop file. Difficulties with out-of-bounds references will go away, but so will good timesharing in the initial phases of the run.

III. FTN LARGE ARRAYS

Although the CDC 7600 has half a million words of memory, the FTN compiler will usually generate code that will execute incorrectly when the index of an array is greater than 131071. This happens because the compiled code uses the 18-bit B-registers and SX instructions (which also truncate to 18 bits) in calculating indexes, even for LCM arrays.

By following the rules below, it is possible to fake out the compiler and make it write code that will execute correctly. The programmer need only be concerned with arrays whose indexes will be greater than 131071 at some time during execution:

(1) Use a statement function such as A(I)=B(I+1) to make the compiler omit the instruction SXi Xj-1. A is used in all executable statements. B is the actual array in LCM.

(2) Use BLKCPY instead of assignment statements to define the values of array elements.

(3) Replace all constants in index expressions by variables in COMMON defined by DATA statements. This includes constants in expressions in

statements that assign values to variables that are later used in index expressions.

(4) If a DO-loop index is used in any index expressions, replace any constants in the DO statement by variables defined by DATA statements, and, if there are no subroutine calls or function references in the loop, insert one to discourage the compiler from using a B-register for an array index to be incremented at the end of the loop.

In MCNP this problem arises only with the array XSN and its equivalent IXSN. The prospective modifier of the code should look at how the above rules have been applied in the ACE subroutines in the XACT and MCRUN overlays.

IV. MCNP TALLY STRUCTURE

MCNP uses a variably dimensioned tallying scheme. With a few exceptions, tally descriptions and data are loaded into the first portion of the DAS array. By use of an EQUIVALENCE statement this array also goes by the names IPTAL, PTAL, ITDS, TDS, RTP, and IPNT. The contents of this array are different at different points in the MCNP calculation. These equivalenced arrays are described in Sections A through C. Other tally variables and the routines which use them are described in Sections D and E.

A. Tally arrays during the reading of INP

The user's input file is read into MCNP by subroutine RDPROB. At this time all tally cards are copied into the RTP array and are located by the pointer array IPNT(I,J,K). The arguments of the IPNT (I,J,K) array are:

K = tally number (numbered in order read by RDPROB)
I = 1/2 = starting/ending address of data in TDS array raw data block
J = card type = 2,3,4,5,6,7,8,9,10,11,12,13,14,15,16 for cards
F,E,T,C,FS,FC,CF,SF,FM,VA,EM,TM,PD,FU and CM.
J = 2 for the FW, FX, FY and FZ (detector tally) cards,
IPNT(1,1,K) = tally number appearing on data card
IPNT(2,1,K) = F card description
= 1,2,3,4,5 for F,FX,FY,FZ,FW
= -1,-2,-3,-4,-5 for *F,*FX,*FY,*FZ,*FW

The IPNT and RTP arrays are equivalenced to the DAS array which at this time has the structure shown in Table D.1.

APPENDIX D
Tally Structure

Table D.1

Address	Array	Description
DAS(1)		unused space
DAS(L1)	IPNT(2,16,NTALMX)	Array of pointers to RTP array
DAS(L2)	RTP(MP6)	Unprocessed data from tally cards

NTALMX = 41
MP6 = 100*NTALMX
L1 = IPTALJ*NTALMX+MP6+1
L2 = IPTALJ*NTALMX+MP6+32*NTALMX+1
IPTALJ = 25

B. Tally arrays after processing by ITALLY

After the entire user-input file is read, the tally data are processed in subroutine ITALLY. This routine establishes pointers to the tally blocks (TAL array) where fluxes, currents, etc. will be scored in the transport portion of MCNP. This routine also processes the raw tally data from the RTP array into the TDS (tally descriptor) array. The processed data in the TDS array are located by information in the IPTAL array. These arrays are all equivalenced to the DAS array which now has the structure described in Table D.2. The IPTAL and TDS arrays are described in Tables D.3 and D.4. Note that the IPTAL, TDS, and TAL arrays overwrite the IPNT and RTP arrays which are no longer needed when subroutine ITALLY is finished.

Table D.2

Layout of DAS Array after Processing by ITALLY

DAS(1)	((IPTAL(I,ITAL),I=1,IPTALJ), ITAL=1,NTAL)	Pointers to data in TDS array (see Table D.3)
DAS(L3)	TDS(I),I=L3,L4	Tally descriptors (see Table D.4)
DAS(LTAL)	TAL(I),I=LTAL,L5	MXF words reserved for Tally block 1 See Section C
DAS(L6)	TAL(I),I=L6,L7	MXF words reserved for Tally Block 2 See Section C
DAS(L8)	TAL(I),I=L8,L9	MXF words reserved for Tally Block 3 See Section C

L3 = IPTALJ*NTAL+1
L4 = LTAL-L3
L5 = LTAL+MXF-1
L6 = LTAL+MXF
L7 = LTAL+2*MXF-1
L8 = LTAL+2*MXF
L9 = LTAL+3*MXF-1

APPENDIX D
Tally Structure

Table D.3

Layout of IPTAL(I,ITAL)
EQUIVALENCE(IPTAL,PTAL)

I=1 location of tallies in TAL array
I=2 location of energy bin array (E card) in TDS array
I=3 location of time bin array (T card) in TDS array
I=4 location of cosine bin array (C card) in TDS array
I=5 location of F card entries in TDS array
I=6 location of time bin multipliers (TM card) in TDS array
I=7 location of energy bin multipliers (EM card) in TDS array
I=8 location of reaction information (FM card) in TDS array
I=9 tally type (1,2,4,5,6,7) i.e., F5 is a type 5 tally
I=10 tally type: ±1 = neutron; ±2 = photon. +/−1 = F/*F tally (asterisk on F−card)
I=11 for cell and surface tallies, location of flagging cell numbers (CF card) in TDS array; −1 means no flagging cells but yes flagging surfaces
I=12 NE+1 = number of energy bins (E card) + 1
I=13 (NE+1)*(NT+1) where NT = number of time bins (T card)
I=14 NC*(NT+1)*(NE+1) where NC = number of cosine bins (C card)
I=15 NX*(NS+1)*NC*(NT+1)*(NE+1) where NX = number of user supplied bins (FU card) and NS = number of segment bins (FS card)
I=16 NT+1 = number of time bins (T card) + 1
I=17 location of segment bin divisors (VA card) in TDS array
I=18 multiplication factor for entire tally. = 1.0 or the first entry of FM card
I=19 location of user supplied bins (FU card) in TDS array
I=20 tally number, i.e., = 123 for F123 tally
I=21 number of fictitious cell with FS card data. For each FS card a fictitious cell, IA, is created which contains the FS card data in the cell arrays LCA(IA) and LJA(LCA(IA))
I=22 location of descriptive comment (FC card) in TDS array
I=23 a) for cell and surface tallies, location of flagging surface numbers (SF card) in TDS array;
b) for detector tallies (type 5), index minus 1 of detector cell contributions (PD card) in TDS array.

I=24 total number of time-energy arrays.
I=25 location of cosine bin multipliers (CM card)

Table D.4

Layout of TDS(I) Array
EQUIVALENCE(TDS,ITDS)

Note: Data in the first part of the TDS array are located by pointers in the IPTAL array. In this table the following notation is used:
In = IPTAL(n,ITAL)

Entry	Description
ITDS(I22)=n_{22}	number of FC card entries
TDS(I22+I), I=1,n_{22}	descriptive comment (FC card)
ITDS(I5)=n_5	number of cells, surfaces, or detectors (F card)
TDS(I5+I), I=1, n_5	problem cell or surface numbers (F card)
or TDS(I5+I), I=1, 5*n_5	quintuples describing detectors
ITDS(I11)=n_{11}	number of flagging cells (CF card)
TDS(I11+I), I=1, n_{11}	program cell names (CF card)
ITDS(I23)=n_{23}	number of flagging surfaces (SF card)
TDS(I23+I), I=1, n_{23}	program surface names (SF card)
TDS(I23+I), I=1, MXA	point detector contribution from program cell I (PD card); MXA = number of cells in problem
ITDS(I2)=n_2	number of energy bins (E card)
TDS(I2+I), I=1, n_2	upper bounds of energy bins (E card)
TDS(I7+I), I=1, n_2	energy bin multipliers (EM card)
ITDS(I3) = n_3	number of time bins (T card) = IPTAL(16,ITAL)-1
TDS(I3+I), I=1, n_3	upper bounds of time bins (T card)
TDS(I6+I), I=1, n_3	time bin multipliers (TM card)
ITDS(I4)=n_4	number of cosine bins (C card)
TDS(I4+I), I=1, n_4	upper bounds of cosine bins (C card)
TDS(I25+I), I=1, n_4	cosine bin multiplier (CM card)
ITDS(I17)=n_{17}	number of segment bin divisors (FS card)
TDS(I17+I), I=1, n_{17}	segment bin divisors (VA card)
ITDS(I19)=n_{19}	number of user supplied tally bins

APPENDIX D
Tally Structure

TDS(I19+I), I=1, n_{19}	entries on FU card
ITDS(I8)=n_8	number of reactions listed on FM card = number of entries - 2
ITDS(I8-1)=IA	name of fictitious or real cell having RHO(IA) = 1.0 and MAT(IA) = material number on FM card
ITDS(I8+I), I=1, n_8	reaction numbers listed on MT card.

After all tallies ITAL=1,NTAL are processed, this last portion of the TDS array is established:

ITDS(LOCCT(IA))=n	number of cell tallies for cell IA
ITDS(LOCCT(IA)+2*I-1), I=1, n	program number of tally, ITAL, which tallies cell IA
ITDS(LOCCT(IA)+2*I), I=1, n	location of word before tally in Block 1 for cell IA, tally ITAL
ITDS(LOCST(JA))=m	number of surface tallies for surface JA
ITDS(LOCST(JA)+2*I-1), I=1, m	program number of tally, ITAL, which tallies surface JA
ITDS(LOCST(JA)+2*I), I=1, m	location of word before tally in Tally Block 1 for surface JA, tally ITAL
ITDS(LOCDT(2,IDET)), IDET=1,NDET	Locator of word before tally in Tally BLOCK 1 for IDET

Subroutine ITALLY also processes data into some arrays not equivalenced to DAS. The LOCCT(MAXA) and LOCST(JMAX) arrays contain pointers for the last part of the TDS array (see Table D.4). The LOCDT(2,MXDT) array contains pointers for detector tallies. Finally, data from the FM card (tally multiplier information) are not loaded into the TDS array. Instead, a fictitious cell is defined. In the case of the FS card the surfaces defining tally segments are loaded into the LCA and LJA cell card arrays as if the fictitious cell were bounded by the surfaces listed on the FS card. In the case of the FM card, the material information is loaded into material arrays as if there were a fictitious cell containing that material. (If there is a real cell containing the material then it is used instead.)

C. Tally Arrays at Transport Time

During the transport portion (4c0 overlay) of MCNP, the tally arrays processed by subroutine ITALLY in the initiation (1c0 overlay) of MCNP are unchanged, but the contents of the TAL array vary as scores are made to each tally. As described in Table D.2, the TAL array is divided into three tally blocks. The first block, Tally Block 1, contains tally information for the current source particle history only. After each source particle history is ended, the data of Tally Block 1 are accumulated in Block 2 and the squares of the data in Tally Block 1 are accumulated in Tally Block 3. When it comes time to print the tallies, the desired currents, fluxes, etc. are simply the data in Tally Block 2 divided by the number of source particle histories. The desired error estimates are readily computed from the data in Tally Block 2 and Tally Block 3.

For example, let x_n be the L^{th} tally accumulated in Tally Block 1 for the n^{th} source particle history. TAL(LTAL+L) = x_n. After the source particle history is ended, x_n is accumulated in Tally Block 2 and

$$\text{TAL(LTAL+L+MXF)} = \sum_{i=1}^{n} x_i$$

The square of x_n is accumulated in Tally Block 3:

$$\text{TAL(LTAL+L+2*MXF)} = \sum_{i=1}^{n} x_i^2$$

The tally which is finally printed after the N^{th} particle history is TAL(LTAL+L+MXF) which is

$$\frac{1}{N} \sum_{i=1}^{N} x_i$$

and its variance is

APPENDIX D
Tally Structure

$$\left[\frac{\sum_{i=1}^{N} x_i^2}{\left(\sum_{i=1}^{N} x_i \right)^2} - \frac{1}{N} \right]^{1/2}$$

The location L within the tally block is
L=I1+(ID-1)*I15*NF+[(IX-1)*(NS+1)+IS-1]*I14
For the IDth cell/surface or detector on the F card,
the IXth user supplied bin (FU card)
the ISth segment bin (FS card)
the ICth cosine bin (C card)
the ITth time bin (T card) and the IEth energy bin (E card)

where In = IPTAL (n,ITAL)
NS = number of segment bins (0 if none).
NF = 1/2 = no/yes flagged or direct contribution. ITAL is the program number of problem tally number IPTAL (20,ITAL). For flagged or direct contribution tallies, L is the location of the regular tally and L+I15 is the location of the flagged or direct contribution.

Tallies are printed and stored in the TAL array according to the following hierachy which may be thought of as a nest of Fortran DO loops with energies being the inner loop.

Printing and Storage Hierarchy

tallies	heading
surfaces or cells or detectors	description
total and flagged or direct	description
user bins	description
segments	description
cosines	cosine line
times	time heading
energies	energy at left

D. Tally Variables in COMMON

Variables used in the MCNP tallying scheme are:

Constants set in PARAMETER statement

IPTALJ = 25 number of words for each tally in IPTAL array
MXDT = 20 total number of point/ring neutron/photon detectors
NTALMX = 41 total number of tallies +1
MP6 = 100*NTALMX words of DAS array set aside for raw or processed tally descriptions

Other COMMON Variables

AMFP = accumulated transmission factor for detector tallies.
= $\sum\rho_i\sigma_i x_i$; ρ_i = density of cell i, σ_i = total cross section of cell i, x_i = path length through cell i
DDET = distance from collision to current point or ring detector
IDET = current detector number
IPCLUN = 0/1 = no/yes photon heating is tallied in units of jerks/gram
IPSC = probability of scattering law number for point/ring detector
LPL = total number of words used for pointers, descriptions and locators of tallies in TDS array. This value is printed under "use of available storage" in MCNP output
MXAFM = MXA (number of problem cell's) + number of psuedo cells used for FM cards
MXAFS = MXA (number of problem cells) + number of pseudo cells used for FS cards
MXF = length of DAS array tally blocks = total number of tally bins
MXF3 = 3*MXF
NDET = total number of detector tallies
NTAL = total number of tallies
PSC = probability of scattering toward point/ring detector

Other COMMON Arrays

IFL(AMAX) = cell flagging array, > 0 if cell entered during current particle history, 0 otherwise
JFL(JMAX) = surface flagging array, > 0 if surface crossed during current particle history, = 0 otherwise
LOCCT(AMAX); LOCCT(I) = 0 if no tallies in cell I;
= location in TDS array of addresses for all tallies in cell I.

LOCDT(2,MXDT); LOCDT(1,I) = ITAL = tally number of detector I;
LOCDT(2,I) = location of word before tally in
Tally Block 1 of DAS array for detector I
LOCST(JMAX); LOCST(I) = 0 if no tallies on surface I;
= location in TDS array of address for
all tallies on surface I

E. Tally Subroutines

References to tallies are made throughout MCNP. The principal subroutines involved are:

IMCN: ITALLY processes raw tally data into IPTAL array and TDS array
TALLYH prints tally heading.

Raw tally data are read by RDPROB and RDLINE and copied into the RTP array with pointers set in IPNT. STUFF is used when an FM card is present. CHGNAM changes problem cell/surface names to program names.

XACT: EXPUNG is used when an FM card is used.

MCRUN:

TALLYD	point/ring detector routines for both neutrons and photons
CALCPS	calculates probability density function PSC for detector tallies
DDDET	calculates various parameters for detector tallies
TALLY	scores cell and surface tallies
TALLYX	user supplied tally routine
TALLYH	prints tally heading
WTMULT	adjusts tally if FM card used
GETXS	fetches cross sections for FM card use
BIN	selects appropriate energy/time/cosine tally bin
TALSHF	after each history shifts data from DAS array Tally Block 1 (Table D.1) to Tally Blocks 2 and 3
TALLYP	prints tallies
TRANSM	calculates particle transmission to detector
OMCFE	once-more-collided flux estimator

APPENDIX E

Cross-Section Formats

The MCNP neutron cross-section libraries are in ACE (A Compact ENDF) format. ACE format consists of a number of binary records as illustrated in Figure E.1.

Nuclide Cross-Section Records

The first record of each MCNP neutron cross section library is the nuclide directory. Following the nuclide directory are all the neutron cross-section data for all the nuclides listed in the directory. Each nuclide has three consecutive records. The PREF record contains the TRACE array - a short (40 words) array of identifiers, pointers, flags and Hollerith information. Then after a record mark comes the EST record which contains the reaction cross-section energy grid and the total cross-sections. After another record mark comes the CAPE record which contains the remaining blocks of data.

The data in the PREF, EST and CAPE records for each nuclide may be thought of as a single array, IN(I), I=1,END where END is less than or equal to 100,000. The PREF record, which comprises the first NPREF=40 words of the IN array, contains pointers for the remaining data in the IN array. In this Appendix data are described in terms of a single IN array because many ACE format processing codes use a single IN array and since the cross-section data libraries are laid out like a series of IN arrays for each nuclide. It should be emphasized that within the IN array the record marks between the PREF, EST, and CAPE records have been removed.

In MCNP the PREF, EST and CAPE records are loaded into different arrays: the PREF record is loaded into the TRACE array; the EST record is loaded into the XSN array; and the CAPE record is loaded into the XSN array after EST. All data in the EST and CAPE records which are not used, particularly data outside the energy range of interest, are deleted by routine EXPUNG. Then the PREF record pointers (which are now in the TRACE array) are changed to correctly locate data in the XSN array.

APPENDIX E
Neutrons

How to Locate ACE Format Data in the Cross-Section Libraries

To locate any ACE format data in the cross-section libraries, first use the nuclide directory (Table 1) to find the starting location of the IN array for the nuclide of interest. Then use the data in the PREF record (Table 2) to locate the specific data block of interest. These data blocks are described in Tables 4-16.

How to Locate ACE Format Data in MCNP

To locate ACE format data for the N^{th} nuclide in MCNP simply use the pointers in the TRACE array (Table 3). These pointers are of the form TRACE (ITR+I), I=1,NPREF where NPREF=40 and ITR=(N-1)*NPREF. The pointers locate different blocks of data in the XSN array. These blocks of data are described in Tables 4-16.

Units

ACE format requires that all energies be given in MeV; all cross sections are in barns ($10^{-24}cm^2$); heating numbers and Q-values are in MeV. Angles are described in terms of cosines.

Figure E.1
Diagram of ACE Format Binary Records
for a Typical MCNP Neutron
Cross-Section Library

record number	starting address	record description	
			record mark
1	1	Nuclide Directory See Table 1	
			record mark
2	DRX(2,1) = 202	Nuclide #1	(All records

		PREF RECORD (TRACE Array)	separated by a record mark)
			record mark
3	DRX(2,1)+NPREF+1	Nuclide #1 EST record	
			record mark
4	DRX(2,1)+NPREF+$NEST_1$+2	Nuclide #1 CAPE record	
			record mark
5	DRX(2,2)	Nuclide #2 PREF record (TRACE Array)	
			record mark
6	DRX(2,2)+NPREF+1	Nuclide #2 EST record	
			record mark
7	DRX(2,2)+NPREF+$NEST_2$+2	Nuclide #2 CAPE record	
			record mark
.	.	.	.
.	.	.	.
.	.	.	.
			record mark
3*N-1	DRX(2,N)	Nuclide #N PREF record (TRACE Array)	N < NDIR NDIR = 100
			record mark
3*N	DRX(2,N)+NPREF+1	Nuclide #N EST record	
			record mark
3*N+1	DRX(2,N)+NPREF+$NEST_N$+2	Nuclide #N CAPE record	

Note: a) NPREF = 40 in the present version of MCNP.

b) $NEST_l$ = the value of NEST for the l^{th} nuclide. NEST = 2*NES+1 where NES is the number of energies in the energy grid for the l^{th} nuclide. The value of NES for each nuclide is stored in the PREF record for each nuclide.

c) In the address of the EST record,

DRX(2,i)+NPREF+1, the +1 is added to account for the record mark between the PREF and EST records. Likewise, the +2 in the CAPE record address, DRX(2,i)+NPREF+NEST$_i$+2, is added to account for the record marks after the PREF and EST records.

Table 1
Nuclide Directory
Length of record = 2*NDIR = 200

Location	Parameter	Description
1	DRX(1,1)	ZAID number of first nuclide in library
2	DRX(2,1)	absolute location of first nuclide in library file (beginning of IN array)
3	DRX(1,2)	ZAID number of second nuclide
4	DRX(2,2)	location of second nuclide
.	.	.
.	.	.
.	.	.
2*N-1	DRX(1,N)	ZAID number of last (N^{th}) nuclide
2*N	DRX(2,N)	location of N^{th} nuclide
2*N+1	DRX(1,N+1)	0.0
2*N+2	DRX(2,N+1)	0.0
.	.	.
.	.	.
.	.	.
2*NDIR-2	DRX(2,NDIR-1)	0.0
2*NDIR-1	DRX(1,NDIR)	name of this library file version
2*NDIR	DRX(2,NDIR)	date when this library file was created

(The ZAID number is the nuclide identifier name.
See Chapter 2 under Materials Specification for details.)

Table 2
PREF Record for a Typical Nuclide in an ACE
Format Cross-Section Library

The PREF record is the first NPREF=40 words of the IN array. This record contains pointers for the rest of the data for this nuclide as it would be found in the single continuous IN array.

Location	Parameter	Description
IN(1)	ZAID	nuclide identifier number
2	AWR	atomic weight ratio
3	NTR	total number of reactions including MT > 100 but not including elastic
4	NES	number of energies
5	NR	number of reactions having secondary neutrons (not elastic)
6	ESZ	loc. of energy table ESZ=NPREF+1=41
7	MTR	loc. of ENDF MT's
8	TYR	loc. of reaction types
9	LSIG	loc. of table of cross section locators
10	SIG	loc. of cross sections
11	LAND	loc. of table of cosine distribution locators
12	AND	loc. of cosine tables
13	LDLW	loc. of table of law locators
14	DLW	loc. of law data
15	GPD	loc. of gamma production data
16	LQR	loc. of table of reaction Q's
17	MGPT	type of data set: ≠-2, neutron cross sections; =-2, S(α,β) library in which case this PREF record has an entirely different meaning.
18	NU	loc. of fission ν data
19	FIS	loc. of total fission cross section
20	END	loc. of last word of this nuclide
21		
22		
23		
24	description of neutron cross sections (BCD Hollerith)	
25	description of neutron cross sections (BCD Hollerith)	
26	description of neutron cross sections (BCD Hollerith)	

27	description of neutron cross sections (BCD Hollerith)
28	description of neutron cross sections (BCD Hollerith)
29	description of neutron cross sections (BCD Hollerith)
30	
31	ENDF MAT number (BCD Hollerith)
32	description of gamma production data (BCD Hollerith)
33	description of gamma production data (BCD Hollerith)
34	description of gamma production data (BCD Hollerith)
35	description of gamma production data (BCD Hollerith)
36	description of gamma production data (BCD Hollerith)
37	description of gamma production data (BCD Hollerith)
38	description of gamma production data (BCD Hollerith)
39	description of gamma production data (BCD Hollerith)
40	date last processed (A10)

Note: GPD=0 no gamma production data
NU=0 no fission
FIS=0 no total fission cross section; partial fission reaction cross sections used instead

Table 3
TRACE Array (PREF records) for Nuclides
Stored in MCNP

This array contains locators for the rest of the data for all nuclides in a problem. This array is NPREF*MAXE words long: ((TRACE(I,N),I=1,NPREF),N=1,MAXE); NPREF=40, MAXE = maximum number of nuclides allowed. For Nuclide N, the TRACE array is:

I	TRACE(I,N)	Description
1	ZAID	nuclide identifier number
2	AWR	atomic weight ratio
3	NTR	total number of reactions excluding elastic
4	NES	number of energies
5	NR	number of nonelastic reactions having secondary neutrons NR ≤ NTR
6	ESZ	loc. of energy table
7	MTR	loc. of ENDF MT's

8	TYR	loc. of reaction types
9	LSIG	loc. of table of cross section locators
10	SIG	loc. of cross sections
11	LAND	loc. of table of cosine distribution locators
12	AND	loc. of cosine tables
13	LDLW	loc. of table of law locators
14	DLW	loc. of law data
15	GPD	loc. of gamma production data. 0 if none
16	LQR	loc. of reaction Q's
17	MGPT	type of neutron data set: ≥0, continuous energy; -1, discrete reaction; -2, S(α,β)
18	NU	loc. of fission ν data. 0 if no fission
19	FIS	loc. of total fission cross section. 0 if no fission
20	END	loc. of last word of this data set
21	-	not used
22	-	not used
23	-	not used
24	-	S(α,β) quantity
25	σ_t	total cross section at energy ERG without thermal effects (as taken from ACE format input data)
26	ERG	energy prior to collision, LAB
27	σ_t	total cross section at energy ERG with thermal effects included
28	IERG	i such that $ES_i \leq ERG < ES_{i+1}$ (ES_i = energy table energy. See ESZ block)
29	RINT	interpolation fraction: $(ERG-ES_i)/(ES_{i+1}-ES_i)$
30	EHD	variable in thermal procedure
31	σ_a	absorption cross section at energy ERG with thermal effects included
32	σ_{el}	elastic cross section at energy ERG with thermal effects included
33	σ_f	total fission cross section at energy ERG without thermal effects
34	-	S(α,β) quantity
35	-	not used
36	-	not used
37	LTNN	thermal collision parameter
38	-	S(α,β) quantity
39	$\bar{\nu}$	average number of neutrons per fission at energy ERG

40 -- presently unused

The above locations are all relative to XSN(1) unless otherwise noted.

Table 4
ESZ BLOCK
Total Length = 5*NES+1
Data in IN(I) for libraries and in XSN(I) for MCNP

Location I	Parameter	Description
IN(ESZ)	NES	number of energy grid points
IN(ESZ+1)	ES(I),I=1,NES	energy table
IN(ESZ+NES+1)	σ_t(I),I=1,NES	total cross section
IN(ESZ+2*NES+1)	σ_a(I),I=1,NES	total absorption cross section
IN(ESZ+3*NES+1)	σ_{el}(I),I=1,NES	elastic cross section
IN(ESZ+4*NES+1)	H_{ave}(I),I=1,NES	average heating numbers

Table 5
NU Block - Fission $\bar{\nu}$ data
XNU=IN(NU) for libraries, = XSN(NU) for MCNP

There are three possibilities for the NU Block:

1. XNU=0 no NU Block
2. XNU>0 either prompt $\bar{\nu}$ or total $\bar{\nu}$ is given. The NU array begins at location IN(KNU) or XSN(KNU) where KNU=NU
3. XNU<0 both prompt $\bar{\nu}$ and total $\bar{\nu}$ are given. The prompt $\bar{\nu}$ NU Array begins at IN(KNU) or XSN(KNU) where KNU=NU+1; the total $\bar{\nu}$ NU Array begins at IN(KNU) or XSN(KNU) where KNU=NU+ABS(XNU)+1.

The NU Array has two forms if it exists:
[Location = IN(IJK) for libraries, XSN(IJK) for MCNP]

a) Polynomial function form of NU Array:

IJK	Parameter	Description
KNU	LNU=1	polynomial function flag
KNU+1	NC	number of coefficients
KNU+2	C(I),I=1,NC	coefficients

$$\bar{\nu}(E) = \sum_{I=1}^{NC} C(I)*E**(I-1) \qquad E \text{ in MeV}$$

b) Tabular data form of NU array

IJK	Parameter	Description
KNU	LNU=2	tabular data flag
KNU+1	NR	number of interpolation regions
KNU+2	NBT(I),I=1,NR	ENDF interpolation parameters
KNU+2+NR	INT(I),I=1,NR	If NR=0, NBT and INT are omitted and linear-linear interpolation is used.
KNU+2+2*NR	NE	number of energies
KNU+3+2*NR	E(I),I=1,NE	tabular energy points
KNU+3+2*NR+NE	$\bar{\nu}$(I),I=1,NE	corresponding values of $\bar{\nu}$

Note: For type 19 fission (FIS=0) the NU block follows the

ESZ block. For type 18 fission (FIS>0) the NU block is buried in the DLW block. NU and FIS are pointers in the PREF record (Tables 2 and 3)

Table 6
MTR Block - ENDF MT Numbers
Length of block = NTR
Location of data: IN(IJK) in cross-section libraries
XSN(IJK) in MCNP

IJK	Parameter	Description
MTR	MT_1	first ENDF reaction available
MTR+1	MT_2	second ENDF reaction available
.	.	.
.	.	.
.	.	.
MTR+NTR-1	MT_{NTR}	last ENDF reaction available

Note: 1. MT_1, MT_2, ... are standard ENDF MT numbers, i.e., MT=16=(n,2n); MT=17=(n,3n); etc.
2. MTR, NTR are defined in the PREF record, Tables 2 and 3

Table 7
LQR Block - Reaction Q-values
Length of block = NTR
Location of data: IN (IJK) in cross-section libraries
XSN(IJK) in MCNP

IJK	Parameter	Description
LQR	Q_1	Q-value of reaction MT_1
LQR+1	Q_2	Q-value of reaction MT_2
.	.	.
.	.	.
.	.	.
LQR+NTR-1	Q_{NTR}	Q-value of reaction MT_{NTR}

Note: LQR and NTR are defined in the PREF record, Tables 2 and 3; MT_1 is defined in Table 6.

Table 8
TYR Block - Number of secondary neutrons released per collision for each reaction type

Length of block: NTR
Location of data: IN(IJK) for cross-section libraries
XSN(IJK) for MCNP

IJK	Parameter	Description
TYR	TY_1	neutron release for reaction MT_1
TYR+1	TY_2	neutron release for reaction MT_2
.	.	.
.	.	.
.	.	.
TYR+NTR-1	TY_{NTR}	neutron release for reaction MT_{NTR}

Note: the possible values of TY_i are ±1, ±2, ±3,...., 18, 19, 0. The sign indicates the system for scattering: negative = CM system; positive = LAB system.
Thus if TY_i = -3, then 3 neutrons are released in the CM system for reaction MT_i.
TY_i = 18 indicates type 18 fission. The number of secondary neutrons released is determined from the total fission cross section, the MCNP fission subroutine, and the fission $\bar{\nu}$ data.
TY_i = 19 indicates type 19 fission. The number of secondary neutrons released is determined from the individual fission reaction cross sections (ENDF reactions MT=19, 20, 21), individual secondary energy laws, and the fission $\bar{\nu}$ data.
TY_i=0 indicates absorption (ENDF reactions MT>100); no neutrons are released.
TYR and NTR are defined in the PREF record, Tables 2 and 3
MT_i is defined in the MTR block, Table 6

APPENDIX E
Neutrons

Table 9
LSIG Block - Reaction cross-section locators
for use in SIG Block (Table 10)

Length of block = NTR
Location of data: IN (IJK) for cross section libraries
XSN(IJK) for MCNP

IJK	Parameter	Description
LSIG	$LOCA_1$	location of reaction MT_1
LSIG+1	$LOCA_2$	location of reaction MT_2
.	.	.
.	.	.
.	.	.
LSIG+NTR-1	$LOCA_{NTR}$	location of reaction MT_{NTR}

Note: All locators are relative to SIG. See the SIG Block. (Table 10). $LOCA_1$ = 1 always.
LSIG and NTR are defined in the PREF record, Tables 2 and 3.
MT_i is defined in the MTR Block, Table 6

Table 10
Reaction Cross Sections

Location of data: IN(IJK) for cross-section libraries
XSN(IJK) for MCNP

IJK	Parameter	ENDF Reaction	XSEC energy	parameter description
SIG	IE_1	MT_1	—	energy grid index
SIG+1	NE_1	MT_1	—	number of consecutive entries
SIG+2	σ_1	MT_1	$ES(IE_1)$	reaction cross section
.	.	.	.	.
.	.	.	.	.
.	.	.	.	.
$SIG+1+NE_1$	σ_{NE_1}	MT_1	$ES(IE_1+NE_1-1)$	reaction cross section
$SIG+LOCA_2-1$	IE_2	MT_2	—	energy grid index
$SIG+LOCA_2$	NE_2	MT_2	—	number of consecutive entries

SIG+LOCA$_2$+1	σ_1	MT$_2$	ES(IE$_2$)	reaction cross section
.	.	.	.	. . .
.	.	.	.	. . .
.	.	.	.	. . .
SIG+LOCA$_2$+NE$_2$	σ_{NE_2}	MT$_2$	ES(IE$_2$+NE$_2$-1)	reaction cross section
		...etc...		
SIG+LOCA$_{NTR}$-1	IE$_{NTR}$	MT$_{NTR}$	--	energy grid index
SIG+LOCA$_{NTR}$	NE$_{NTR}$	MT$_{NTR}$	--	number of consecutive entries
SIG+LOCA$_{NTR}$+1	σ_1	MT$_{NTR}$	ES(IE$_{NTR}$)	reaction cross section
.	.	.	.	. . .
.	.	.	.	. . .
.	.	.	.	. . .
SIG+LOCA$_{NTR}$ +NE$_{NTR}$	$\sigma_{NE_{NTR}}$	MT$_{NTR}$	ES(IE$_{NTR}$+NE$_{NTR}$-1)	reaction cross section

Note: The values of LOCA$_i$ are given in the LSIG block (Table 9). The energy grid, ES(I), I=1,NES, is given in the ESZ block (Table 4). The reaction numbers, MT$_i$, are given in the MTR block (Table 6). The values of SIG, NES, and NTR are given in the PREF record (Tables 2 and 3). The energy grid index IE$_i$ corresponds to the first energy in the grid at which a cross section is given.

Table 11
LAND Block: Cosine Distribution locators for use in AND Block (Table 12)

Length of block: NR+1
Location of data: IN(IJK) for cross-section libraries
XSN(IJK) for MCNP

IJK	Parameter	Description
LAND	LOCB$_1$=1	location of cosine distribuion for elastic reaction
LAND+1	LOCB$_2$	location of cosine distribution for reaction MT$_1$
.	.	.
.	.	.
.	.	.
LAND+NR	LOCB$_{NR+1}$	location of cosine distribution for reaction MT$_{NR}$

Note: all locators are relative to AND. If $LOCB_i$=0, there is no cosine distribution given for this reaction and isotropic scattering is assumed in either the LAB or CM system. Choice of LAB or CM system depends upon values for this reaction in the TYR block (Table 8). LAND, AND and NR are defined in the PREF record, Tables 2 and 3. MT_i is defined in the MTR block, Table 6.

Table 12
AND Block - Cosine Distribution Tables

Location of data: IN(IJK) for cross-section libraries
XSN(IJK) for MCNP

IJK	Description
AND+$LOCB_1$-1	AND Array for elastic collision
AND+$LOCB_2$-1	AND Array for reaction MT_1
.	
.	
.	
AND+$LOCB_{NR+1}$-1	AND Array for reaction MT_{NR}

Note:
1. $LOCB_i$ is defined in the LAND block, Table 11;
2. Whereas $LOCB_i$=1, the first word of the AND block, IN(AND) or XSN(AND), is not used;
3. If $LOCB_i$=0, then no AND array is given and scattering is assumed to be isotropic in either the LAB or CM system. Choice of LAB or CM system depends on the TYR block (Table 8).

The i^{th} AND Array has the form

IJK	Parameter	Description
AND+$LOCB_i$-1	NE	number of energies at which angular distributions are tabulated.
AND+$LOCB_i$	E(J),J=1,NE	energy grid for table
AND+$LOCB_i$+NE	LC(J),J=1,NE	location of tables associated with energy grid point E(J).

		Note that if LC(J)=0 then no table is given for energy E(J) and scattering is isotropic in the coordinate system indicated by the TYR Block.
AND+LC_1-1	P(1,K),K=1,33	32 equi-probable cosine bins for scattering at energy E_1.
.	.	.
.	.	.
.	.	.
AND+LC_J-1	P(J,K),K=1,33	Bins for scattering at energy E_J.
.	.	Note that P(J,1)=-1 and P(J,33)=1.
.	.	.
.	.	.
AND+LC_{NE}-1	P(NE,K),K=1,33	bins for scattering at energy E_{NE}

Note: MT_i is defined in the MTR block, Table 6.
AND and NR are defined in the PREF record, Tables 2 and 3.

Table 13
LDLW Block - Law Data locators for use in DLW Block (Table 14)

Length of block = NR
Location of data: IN(IJK) for cross-section libraries
XSN(IJK) for MCNP

IJK	Parameter	Description
LDLW	$LOCC_1$	law data locator for reaction MT_1
LDLW+1	$LOCC_2$	law data locator for reaction MT_2
.	.	.
.	.	.
.	.	.
LDLW+NR-1	$LOCC_{NR}$	law data locator for reaction MT_{NR}

Note: MT_i is defined in the MTR block, Table 6. LDLW and NR are defined in the PREF record, Tables 2 and 3.

Table 14
DLW Block - Law Data for Each Reaction

Location of data: IN(IJK) for cross-section libraries
XSN(IJK) for MCNP

IJK	Parameter	Description
DLW+LOCC$_I$-1	LNW$_I$	location of next law. If LNW=0, then law LAW$_I$, is used regardless of other circumstances.
DLW+LOCC$_I$	LAW$_I$	name of this law
DLW+LOCC$_I$+1	IDAT$_I$	location of data for this law relative to DLW
DLW+LOCC$_I$+2	NR	number of interpolation regions to define law applicability regime
DLW+LOCC$_I$+3	NBT(I),I=NR	ENDF interpolation parameters.
DLW+LOCC$_I$+3+NR	INT(I),I=1,NR	If NR=0, NBT and INT are omitted and linear-linear interpolation is used.
DLW+LOCC$_I$+3+2*NR	NE	number of energies
DLW+LOCC$_I$+4+2*NR	E(I),I=NE	tabular energy points
DLW+LOCC$_I$+4 +2*NR+NE	P(I),I=1,NE	probability of law validity. If the particle energy, E, is E<E(1) then P(E)=P(1) If E>E(NE) then P(E)=P(NE). If more than 1 law is given, then law LAW$_I$ is used only if ξ<P(E) where ξ is a random number between 0 and 1.
DLW+IDAT$_I$-1	LDAT(I),I=1,L	law data array for law LAW$_I$. The length, L, of the law data array, LDAT, is determined from parameters within LDAT. The various law data arrays, LDAT for each law, LAW$_I$, are given in Table 15
DLW+LNW$_1$-1	LNW$_2$	location of next law
DLW+LNW$_1$	LAW$_2$	name of this law
DLW+LNW$_1$+1	IDAT$_2$	location of data for this law

. . .
. . .

etc.

Note: The locator, $LOCC_1$ is defined in the LDLW Block, Table 13. All locators are relative to DLW. DLW is defined in the PREF record, Tables 2 and 3.

Table 15
Description of Law Data of Various Laws

The location of the LDAT array is defined in the DLW Block, Table 14

a. LAW_1=1 ENDF Law 1 Tabular Energies Out

Location	Parameter	Description
LDAT(1)	NR	interpotation scheme
LDAT(2)	NBT(I),I=1,NR	between tables of E_{out}
LDAT(2+NR)	INT(I),I=1,NR	If NR=0 or if INT(I) ≠1= histogram, linear-linear interpolation is used
LDAT(2+2*NR)	NE	number of incident energies tabulated
LDAT(3+2*NR)	E_{in}(I),I=1,NE	list of incident energies for which E_{out} is tabulated
LDAT(3+2*NR+NE)	NET	number of outgoing energies in each E_{out} table
LDAT(4+2*NR+NE)	E_{out_1}(I),I=1,NET	E_{out} tables are NET endpoints of NET-1 equally likely energy intervals. Linear-linear interpolation is used between intervals.
	.	
	.	
	.	
	$E_{out_{NE}}$(I),I=1,NET	

b. LAW_1=3 ENDF Law 3 Level Scattering

E_{out} = LDAT(2)*(E-LDAT(1))

For CM system,

$$LDAT(1) = \left(\frac{A+1}{A}\right)|Q| \qquad LDAT(2) = \left(\frac{A}{A+1}\right)^2$$

For LAB system,

$$LDAT(1) = \frac{A(A+1)|Q|}{A^2+1} \qquad LDAT(2) = \frac{A^2+1}{(A+1)^2}$$

where E = incident particle energy
A = atomic weight ratio
Q = Q-value

c. LAW_i=5 ENDF Law 5 General Evaporation Spectrum
$E_{out} = X(\xi)*T(E)$ where $X(\xi)$ is a randomly sampled table of X's, and E is the incident energy

Location	Parameter	Description
LDAT(1)	NR	interpolation scheme between T's
LDAT(2)	NBT(I),I=1,NR	
LDAT(2+NR)	INT(I),I=1,NR	
LDAT(2+2*NR)	NE	number of incident energies tabulated
LDAT(3+2*NR)	E(I),I=1,NE	incident energy table
LDAT(3+2*NR+NE)	T(I),I=1,NE	tabulated function of incident energies
LDAT(3+2*NR+2*NE)	NET	number of X's tabulated
LDAT(4+2*NR+2*NE)	X(I),I=1,NET	tabulated probabilistic function

d. LAW_i=7 ENDF Law 7 Simple Maxwell Fission Spectrum

$$f(E \rightarrow E_{out}) = C*SQRT(E_{out})\ \exp(-E_{out}/T(E))$$

with restriction $0 \le E_{out} \le E-U$

$$C = T^{-3/2}\left[\frac{\sqrt{\pi}}{2}\ \mathrm{erf}\left(\sqrt{(E-U)/T}\right) - \sqrt{(E-U)/T}\ e^{-(E-U)/T}\right]^{-1}$$

The format of the LDAT array is given below with the description of LAW 9.

e. LAW_i=9 ENDF Law 9 Evaporation Spectrum

$$f(E\rightarrow E_{out}) = C*E_{out}\exp(-E_{out}/T(E))$$

with restriction $0 \le E_{out} \le E-U$

$$C = T^{-2}\left[1-e^{-(E-U)/T}\left(1+(E-U)/T\right)\right]^{-1}$$

Location	Parameter	Description
LDAT(1)	NR	interpolation scheme between T's
LDAT(2)	NBT(I),I=1,NR	
LDAT(2+NR)	INT(I),I=1,NR	
LDAT(2+2*NR)	NE	number of incident energies tabulated
LDAT(3+2*NR)	E(I),I=1,NE	incident energy table
LDAT(3+2*NR+NE)	T(I),I=1,NE	tabulated T's
LDAT(3+2*NR+2*NE)	U	restriction energy

f. LAW_i=10 ENDF Law 10 Watt Spectrum

$$f(E\rightarrow E_{out}) = C_o \exp(-E_{out}/a) \sinh(b*E_{out})^{1/2}$$

This is sampled by the rejection scheme in LA-5061-MS (R11 pg. 45):

let $k = 1+ab/8$

$L = a(k+(k^2-1)^{1/2})$

$M = k+(k^2-1)^{1/2}-1$

let $X = -\ln \xi_1$ ξ_1, ξ_2 = random numbers [0,1]

$Y = -\ln \xi_2$

accept X if $(Y-M(X+1))^2 \le bLX$

then $E_{out} = L\ X$

LDAT(1)=b; LDAT(2)=M; LDAT(3)=L

g. LAW_i=11 ENDF Law 11 Energy Dependent Watt Spectrum

$$f(E \rightarrow E_{out}) = C_o \exp[-E_{out}/a(E_{in})] \sinh[b(E_{in}) \cdot E_{out}]^{1/2}$$

with restriction $0 \leq E_{out} < E-U$

This is also sampled by the rejection scheme in LA-5061-MS (R11, pg. 45) as is done in Law 10.

Location	Parameter	Description
LDAT(1)	NR	
LDAT(2)	NBT(I),I=1,NR	Interpolation scheme between a's
LDAT(2+NR)	INT(I),I=1,NR	
LDAT(2+2*NR)	NE	Number of incident energies tabulated for $a(E_{in})$ table
LDAT(3+2*NR)	E(I),I=1,NE	Incident energy table
LDAT(3+2*NR+NE)	a(I),I=1,NE	Tabulated a's
let L=3+2•(NR+NE)		
LDAT(L+1)	NR	
LDAT(L+2)	NBT(I),I=1,NR	Interpolation scheme between b's
LDAT(L+2+NR)	INT(I),I=1,NR	
LDAT(L+2+2*NR)	NE	Number of incident energies tabulated for $b(E_{in})$ table
LDAT(L+3+2*NR)	E(I),I=1,NE	Incident energy table
LDAT(L+3+2*NR+NE)	b(I),I=1,NE	Tabulated b's
LDAT(L+3+2*NR+2*NE)	U	Rejection energy

h. LAW_i=18 Fission using MCNP fission subroutine. The LDAT Array is simply the NU Block (Table 5)

i. LAW_i=21 UK Law 1 Tabular Energies Out
(This law is the same as LAW 1 except a probability distribution is used rather than an equi-probable mesh.)

Location	Parameter	Description
LDAT(1)	NR	Interpolation parameters which are not used by MCNP
LDAT(2)	NDT(I),I=1,NR	
LDAT(2+NR)	INT(I),I=1,NR	

LDAT(2+2*NR)	NE	number of incident energies tabulated
LDAT(3+2*NR)	$E_{in}(I), I=1,NE$	list of incident energies for which E_{out} is tabulated
LDAT(3+2*NR+NE)	LOCD(I), I=1,NE	locators of E_{out} tables

Location IJK*	Parameter	Description
DLW+LOCD$_1$-1	NF_1	if $E_{in_1} \le E < E_{in_{I+1}}$ and ξ is a random number [0,1] then if $\sum_{k=1}^{k=K} P_{I,k} < \xi \le \sum_{k=1}^{k=K+1} P_{I,k}$ $E_{out} = E_{out_{I,K}}$
DLW+LOCD$_1$	$P_1(K), K=1,NF_1$	
DLW+LOCD$_1$+NF$_1$	$E_{out_1}(k), K=1,NF_1$	
DLW+LOCD$_2$-1	NF_2	
DLW+LOCD$_2$	$P_2(K), K=1,NF_2$	
DLW+LOCD$_2$+NF$_2$	$E_{out_2}(K), K=1,NF_2$	
.	.	
.	.	
.	.	
DLW+LOCD$_{NE}$-1	NF_{NE}	
DLW+LOCD$_{NE}$	$P_{NE}(K), K=1,NF_{NE}$	
DLW+LOCD$_{NE}$+NF$_2$	$E_{out_{NE}}(K), K=1,NF_{NE}$	

*Location = IN(IJK) for cross-section libraries
= XSN(IJK) for MCNP

j. LAW_i = 22 UK Law 2 Tabular linear functions of incident energy out

Location	Parameter	Description
LDAT(1)	NR	Interpolation parameters which are not used by MCNP
LDAT(2)	NBT(I), I=1,NR	
LDAT(2+NR)	INT(I), I=1,NR	
LDAT(2+2*NR)	NE	number of incident energies tabulated
LDAT(3+2*NR)	$E_{in}(I), I=1,NE$	list of incident energies for E_{out} tables
LDAT(3+2*NR+NE)	LOCE(I), I=1,NE	locators of E_{out} tables

APPENDIX E

Neutrons

Location IJK*	Parameter	Description
DLW+LOCE$_I$-1	NF$_I$	
DLW+LOCE$_I$	P$_I$(K),K=1,NF$_I$	
DLW+LOCE$_I$+NF$_I$	T$_I$(K),K=1,NF$_I$	
DLW+LOCE$_I$+2*NF$_I$	C$_I$(K),K=1,NF$_I$	if $E_{in_I} \le E < E_{in_{I+1}}$ and ξ is a random number [0,1] then if
DLW+LOCE$_2$-1	NF$_2$	
DLW+LOCE$_2$	P$_2$(K),K=1,NF$_2$	
DLW+LOCE$_2$+NF$_2$	T$_2$(K),K=1,NF$_2$	
DLW+LOCE$_2$+2*NF$_2$	C$_2$(K),K=1,NF$_2$	$\sum_{k=1}^{k=K} P_{I,k} < \xi \le \sum_{k=1}^{k=K+1} P_{I,k}$
.	.	
.	.	$E_{out} = C_{I,K}*(E-T_{I,K})$
.	.	
DLW+LOCE$_2$-1	NF$_{NE}$	
DLW+LOCE$_{NE}$	P$_{NE}$(K),K=1,NF$_{NE}$	
DLW+LOCE$_{NE}$+NF$_{NE}$	T$_{NE}$(K),K=1,NF$_{NE}$	
DLW+LOCE$_{NE}$+2*NF$_{NE}$	C$_{NE}$(K),K=1,NF$_{NE}$	

*Location = IN(IJK) for cross-section libraries
= XSN(IJK) for MCNP

k. LAW$_I$ = 23 UK Law 5

$E_{out} = T_I*E^{1/2}$ where T_I is the same as E_{out} in LAW$_I$=1 except that there is no interpolation between tables.

l. LAW$_I$=24 UK Law 6

$E_{out} = T_I*E$ where T_I is the same as E_{out} in LAW$_I$=1 except that there is no interpolation between tables.

Table 16
Gamma Production Block
Length of block = NES+600

Location of data = IN(IJK) for cross-section libraries
= XSN(IJK) for MCNP

IJK	Parameter	Description
GPD	$\sigma_\gamma(I), I=1,NES$	Gamma production cross section
GPD+NES	EG(1,K),K=1,20	20 equally likely outgoing photon energies for incident neutron energy EN(1) ≤ E < EN(2)
GPD+NES+20	EG(2,K),K=1,20	20 equiprobable outgoing photon energies for incident neutron energy EN(2) ≤ E < EN(3)
.	.	.
.	.	.
.	.	.
GPD+NES+580	EG(30,K),K=1,20	20 equiprobable outgoing photon energies for incident neutron energy E ≥ EN(30)

Note: the discrete incident neutron energy array in MeV is EN(J),J=1,30: 1.39E-10, 1.52E-7, 4.14E-7, 1.13E-6, 3.06E-6, 8.32E-6, 2.26E-5, 6.14E-5, 1.67E-4, 4.54E-4, 1.235E-3, 3.35E-3, 9.23E-3, 2.48E-2, 6.76E-2, .184, .303, .500, .823, 1.353, 1.738, 2.232, 2.865, 3.68, 6.07, 7.79, 10., 12., 13.5, 15.

GPD and NES are defined in the PREF record, Tables 2 and 3.

APPENDIX E
Photons

Photon Cross Sections

Merging of Cross Sections:

PXSEC is a short merging code that generated the public file of photon cross sections called MCPLIB that is used by MCNP. It is in the LIX file RGSPPR on CFS under /X6CODE/PHOTON.

An earlier code of X-6 called MCP (see LA-5157-MS) had photon cross sections for Z=1 to Z=94 at energies up to 15 MeV. The current ENDF file of photon cross sections goes up to 100 MeV.

PXSEC does the following:

1) checks the ENDF file to verify that the partial cross sections above 15 MeV add to the total tabulated cross section.
2) Merges the cross sections of the MCP file, up to 15 MeV, with the ENDF file above 15 MeV.
3) Creates a file set directory similar to the neutron directory, and adds it to the merged file.
4) Calculates a heating number at each of the energy mesh points for each element.
5) Prints the cross sections in the format of MCP. The energies and cross sections are printed instead of the natural logs of these numbers, which is what is stored in MCPLIB.

Heating Numbers

The heating numbers are calculated from the average energy deposited from incoherent, photoelectric plus fluorescence, and pair production processes. The pair production average energy deposited is deterministic.

The sampling for the average energy deposited from incoherent and fluorescent scattering is done using the same code as MCNP, except that the random number selection is done by quadrature. That is, the interval (0,1) is divided into 1000 parts to yield 1000 values of the random number, equally spaced in the interval. The energy deposited is calculated at each of these points, and the average determined. For incoherent scattering, the random number is used to determined the outgoing energy E′ from the incoming energy E, in the Klein-Nishina treatment.

The expected energy loss per photon collision has been calculated for MCG gamma cross sections. These cross sections have no coherent data, no incoherent form factors, and no fluorescent treatment. A comparison has been made of MCG versus MCP heating numbers at the MCP energy mesh points. This was for Z=94. In general, the agreement was good except at E=.15 MeV, where the emergent fluorescent photon had an energy comparable to the incoming energy.

Definition of Files Used in PXSEC

Files (All on CFS under /X6CODE/PHOTON):

BETA34 ← input file from LCMCPLB . These are the old MCP cross sections from CROS that have been run through CRSCNV.

BETA2 ← RGSPXS input file from ENDF file of photon cross sections. RGSPXS came from the ENDF file DLC7EHP available from Group T-2. The file DLC7EHP is in CROS format. It was run through CRSCNV, CROSPL, and UPDATE with options 8,d in order to remove the identifiers.

BETA4 → BCD output file. This file is saved as RGSPHXS so that microfiche listings of the cross sections can easily be made.

BETA35 → NMCPLIB-binary cross section output of the merged files. This file is saved as RGSMCP, and is also used as the public file MCPLIB.

Format of Library MCPLIB

Directory:

This is identical in format to the neutron cross section directory. The 200 word directory lists the Z-number of each element, followed by its beginning decimal address. For example, the first four words of the directory are 1., 202, 2., 589. Element 1, hydrogen, has its cross sections starting at location 202. Its length is 589-202, or 387 words.

Pointer Words:

Each element has a nine word pointer array P(I), where

P(1) = the Z-number of the element.

P(2) = the number of energies.
P(3) = the location of the incoherent data.
P(4) = the location of the coherent data.
P(5) = the location of the fluorescent data.
P(6) = the length of the fluorescent data.
P(7) = the location of the heating numbers.
P(8),P(9) not used at present.

Cross Sections:

Following the nine word pointer block are the partial cross sections and data:

A) - the natural log of the energies.
B) - the natural log of the incoherent cross sections.
C) - the natural log of the coherent cross sections.
D) - the natural log of the photoelectric cross sections.
E) - the natural log of the pair production cross sections.
F) - the incoherent form factors.
G) - the integrated coherent form factors.
H) - the coherent form factors.
I) - the fluorescent data (if any).
J) - the heating numbers.

The lengths of blocks A) through E) and J) are given by P(2).
The length of block F) is 21.
The lengths of blocks G) and H) are 55 each.
The length of Block I) is P(6).

Notice that the total cross section is not stored in the file.

Hydrogen has 41 energies and no fluorescent data.
Its total length is 9+6*41+21+2*55+1(eor)=387 words.

Comparison of MCPLIB and ENDF-102 data for Z=26.

Total Cross Section:

There is good agreement in the total cross section below 15 MeV. This is not surprising since the data base (Howerton) is the same. There are a few more points in the ENDF file. This file lists only the total and photoelectric cross section at a resonance energy. Consequently, the

coherent and incoherent cross sections at this energy would have to be interpolated and added to the file in order to put it into the MCPLIB format.

Incoherent Form Factors:

The form factors of MCPLIB are a subset of the ENDF. It would require only a minor change in the code MCNP to use the ENDF data.

Coherent Form Factors:

MCNP uses the coherent form factors, (actually their squares), directly in the point detector calculation. The code uses a normed integral of these form factors for the scattering. See LA-5157-MS, page 8, for the definition of this integral, $A(Z,V^2)$. The code that originally calculated these factors is missing. A small LTSS code has been written which reads the MCPLIB file for the regular coherent form factors and re-calculates SCOHFF by a trapezoidal rule. This code has as input in a data statement a set of V(I), I=1,55, provided by Cashwell and Everett (Cromer/Hanson coherent scattering V's). This small code could easily be modified to read the ENDF file for the coherent form factors, and make the same calculation.

Some comments on the use of ENDF-102 data entirely:

This file has no data for the six elements Z=84, 85, 87, 88, 89, and 93. Consequently, the file MCPLIB has data for these elements only to 15 MeV, which is the old MCP data. These data are essentially the Storm/Israel data from the nuclear data tables 7, 565-681 (1970).

The ENDF file appears to have no fluorescent data. The report LA-5240-MS[17] gives the data and the fluorescent treatment that is in MCNP.

If ENDF heating numbers were used, they would need to be calculated at each of the energy mesh points of MCPLIB. The quadrature method used in PXSEC to calculate heating is consistent with the code for the scattering treatment in MCNP. However, the code could change. In particular, the latest method of sampling the Klein-Nishina probability distribution, (LA-7188-MS)[18], is not used in MCNP.

APPENDIX F

Cross-Section Libraries

This appendix is divided into three sections. The first section lists some of the more frequently used ENDF/B reaction types that can be used with the FMn input card. Secondly, a list is given of all neutron cross-section evaluations in the MCNP libraries that do not have an upper energy of 20 MeV or a lower energy of 10^{-11} MeV. Thirdly, a list of all evaluations in the neutron cross-section libraries is given.

I. *ENDF/B REACTION TYPES*

The following list includes some of the more useful reactions for use with the FMn input card, but it is not the complete ENDF/B list. The complete list can be found in the ENDF/B manual.

R	Microscopic Cross-Section Description
1	Total
2	Elastic
3	Total nonelastic = total - elastic
4	Total inelastic (n,n') = sum of R = 51 to 91
16	(n,2n)
17	(n,3n)
18	Total fission (n,fx) = sum of R = 19,20,21,38
19	(n,f)
20	(n,n'f)
21	(n,2nf)
22	(n,n')α
38	(n,3nf)
51	(n,n') to 1^{st} excited state
52	(n,n') to 2^{nd} excited state
.	.
.	.
.	.
90	(n,n') to 40^{th} excited state

91	(n,n′) to continuum
102	(n,γ)
103	(n,p)
104	(n,d)
105	(n,t)
106	(n,^{3}He)
107	(n,α)

Notes:

(1) The R number for tritium production varies from nuclide to nuclide and evaluation to evaluation. For ^{6}Li and ^{7}Li

^{6}Li(n,α)t	R = 107	ENDF/B evaluations
^{7}Li(n,n′)αt	R = 91	ENDF/B evaluations
^{6}Li(n,t)α	R = 105	ENDL 1976 evaluation
^{7}Li(n,n′t)	R = 33	ENDL 1973 evaluation
^{7}Li(n,n′α)t	R = 22	AWRE evaluation

(2) The nomenclature between MCNP and ENDF/B is inconsistent in that MCNP refers to the number of the reaction type as R whereas ENDF/B uses MT. They are one and the same, however. The problem arises since MCNP has an MT input card used for the S(α,β) thermal treatment.

APPENDIX F

II. NEUTRON NONSTANDARD ENERGY LIMITS

Following is a list of 98 neutron cross-section evaluations that do not fall within the energy range of either 20 MeV or 10^{-11} MeV. If neutrons are transported above or below the energy range of a particular cross-section evaluation, they are transported with the cross sections at the maximum or minimum energy, as the case may be.

ZAID	EMIN	EMAX	ZAID	EMIN	EMAX
1001.01	1.0000e-10	2.0000e+01	7014.02	1.0000e-10	2.0000e+01
1001.30	1.0000e-10	2.0000e+01	7014.30	1.0000e-10	2.0000e+01
1002.01	1.0000e-10	2.0000e+01	8016.01	1.0000e-10	2.0000e+01
1002.02	1.0000e-10	2.0500e+01	8016.02	1.0000e-10	2.0000e+01
1003.01	1.0000e-10	2.0000e+01	8016.30	1.0000e-10	2.0000e+01
1003.30	1.0000e-10	2.0000e+01	9019.01	1.0000e-10	2.0000e+01
2000.01	1.0000e-11	1.5000e+01	11023.01	1.0000e-10	2.0000e+01
2003.01	1.0000e-10	2.0000e+01	12000.01	1.0000e-10	2.0000e+01
2003.02	1.0000e-10	1.5000e+01	12000.02	1.0000e-11	1.8000e+01
2004.02	2.5000e-08	1.4600e+01	13027.01	1.0000e-10	2.0000e+01
2004.30	1.0000e-10	2.0000e+01	13027.02	6.0000e-10	1.5000e+01
3006.01	1.0000e-09	1.5000e+01	14000.01	1.0000e-10	2.0000e+01
3006.30	1.0000e-10	2.0000e+01	15031.01	1.0000e-10	2.0000e+01
3007.01	1.0000e-10	2.0000e+01	16032.01	1.0000e-10	2.0000e+01
3007.02	1.0000e-09	1.5000e+01	17000.01	1.0000e-10	2.0000e+01
3007.03	1.0000e-11	1.5000e+01	18000.01	1.0000e-10	2.0000e+01
4009.01	1.0000e-10	2.0000e+01	19000.01	1.0000e-10	2.0000e+01
5000.01	1.0000e-10	2.0000e+01	20000.01	1.0000e-10	2.0000e+01
5010.01	1.0000e-10	2.0000e+01	22000.01	1.0000e-10	2.0000e+01
5010.02	1.0000e-11	1.5000e+01	22000.02	1.0000e-09	1.8000e+01
5011.01	1.0000e-10	2.0000e+01	24000.01	1.0000e-10	1.5000e+01
5011.02	1.0000e-11	1.5000e+01	25055.01	1.0000e-10	2.0000e+01
6012.01	1.0000e-10	1.5000e+01	26000.01	1.0000e-10	2.0000e+01
6012.02	1.0000e-11	1.5000e+01	26000.30	1.0000e-10	2.0000e+01
6012.04	2.5300e-08	2.2260e+01	28000.01	1.0000e-10	1.5000e+01
6012.30	1.0000e-10	2.0000e+01	28058.01	1.0000e-10	2.0000e+01
7014.01	1.0000e-10	2.0000e+01	29000.01	1.0000e-10	2.0000e+01

29000.02	1.0000e-10	1.5000e+01	90232.01	1.0000e-10	2.0000e+01
31000.01	1.0000e-10	2.0000e+01	90232.02	1.0000e-10	1.5000e+01
40000.01	1.0000e-10	1.7000e+01	90232.10	1.0000e-10	2.0000e+01
40000.02	1.0000e-10	2.0000e+01	92233.01	1.0000e-10	2.0000e+01
41093.01	1.0000e-10	2.0000e+01	92233.10	1.0000e-10	2.0000e+01
41093.30	1.0000e-10	2.0000e+01	92234.01	1.0000e-10	2.0000e+01
42000.01	1.0000e-10	2.0000e+01	92234.10	1.0000e-10	2.0000e+01
48000.01	1.0000e-10	2.0000e+01	92235.01	1.0000e-10	2.0000e+01
50000.01	1.0000e-10	2.0000e+01	92235.30	1.0000e-10	2.0000e+01
50999.02	1.0000e-10	2.0000e+01	92236.01	1.0000e-10	2.0000e+01
56138.01	1.0000e-10	2.0000e+01	92237.01	1.0000e-10	2.0000e+01
63000.01	1.0000e-10	2.0000e+01	92238.01	1.0000e-10	2.0000e+01
64000.01	1.0000e-10	2.0000e+01	92238.30	1.0000e-10	2.0000e+01
67165.01	1.0000e-10	2.0000e+01	92239.01	1.0000e-10	2.0000e+01
73181.01	2.5000e-08	1.4600e+01	92240.01	1.0000e-10	2.0000e+01
73181.02	1.0000e-10	2.0000e+01	94238.01	1.0000e-10	2.0000e+01
74000.01	1.0000e-10	2.0000e+01	94239.01	1.0000e-10	2.0000e+01
78000.01	1.0000e-10	2.0000e+01	94239.99	2.5000e-08	1.4600e+01
79197.01	1.0000e-10	2.0000e+01	94240.01	1.0000e-10	2.0000e+01
79197.02	2.5000e-08	1.4600e+01	94241.01	1.0000e-10	2.0000e+01
79197.10	1.0000e-10	2.0000e+01	95242.01	1.0000e-10	2.0000e+01
82000.01	1.0000e-10	2.0000e+01	95242.01	1.0000e-10	2.0000e+01
82000.02	1.0000e-10	1.5000e+01			

APPENDIX F

III. MCNP CROSS-SECTION LIBRARIES

Discussion of the Monte Carlo cross-section libraries is given in two parts: (1) description of the available libraries, and (2) available visual information to aid the user in cross-section selection. A list of the library contents then follows.

A. LIBRARY DESCRIPTION

Continuous energy or pointwise neutron data for use in Monte Carlo calculations are available in five separate libraries. Each nuclide evaluation in the libraries has a unique identifier number in each of the libraries. Photon cross-section data are available in only one library, MCPLIB.

The Recommended Monte Carlo Cross Section (RMCCS) library contains cross sections for isotopes and elements which are, in our opinion, the best (and usually the most current) evaluations of nuclear data that have been processed. The RMCCS library has one, and only one, evaluation of each nuclide listed. With sufficient core storage the RMCCS library can handle most requirements that a user may need. On the other hand the Alternate Monte Carlo Cross Section (AMCCS) library contains different evaluations of many of the isotopes on the RMCCS library. AMCCS also contains cross-section information at different temperatures, and some evaluations have no photon production data. Cross-section needs for most problems can be satisfied using the RMCCS and AMCCS libraries. For the user who requires special or archival cross-section information two other libraries are available: XMCCS and UMCCS. Nuclides in these libraries may have more than one evaluation and should be used cautiously.

A fifth library, the Discrete Reaction Monte Carlo Cross Section (DRMCCS) library repeats much of the RMCCS and AMCCS data and some of the XMCCS and UMCCS data in "multigroup" form. The 240-group treatment applies only to neutron reaction cross sections; secondary angular and energy distributions are identical to those in the continuous energy RMCCS, AMCCS, XMCCS and UMCCS libraries. DRMCCS cross sections are accessed whenever a DRXS card is used (see page 156). A great advantage of using the discrete cross sections is that computer cross-section storage requirements can be reduced (by a factor of two on the average) so that a timesharing environment is enhanced.

Each nuclide in the above libraries is identified by a unique nuclide

identifier number called the ZAID number. See page 154 for details concerning the ZAID number. If a nuclide's ZAID number is listed on the DRXS card, then the cross sections are read off the DRMCCS library rather than off one of the continuous energy libraries. An index at the end of this Appendix gives the cross-section evaluations and libraries for each ZAID number. The list indicates which nuclides of the RMCCS, AMCCS, XMCCS, and UMCCS libraries have discrete reaction counterparts on the DRMCCS library file.

Finally, the user may use any other library file of his own simply by using the XS input card and/or by using the XS option on the MCNP execution line.

B. VISUAL INFORMATION

To assist the user in cross-section selection two sources of information describing the cross sections in the Monte Carlo libraries are available. First, there are microfiche cards available which display the Hollerith and BCD information for all isotopes in the Monte Carlo libraries. Second, there is an index of the cross sections which are on each of the RMCCS, AMCCS, DRMCCS, XMCCS, or UMCCS libraries. This index is at the end of this Appendix and lists the nuclide identification number (ZAID) and its library location. Also on the index is pertinent Hollerith information in addition to an important parameter SIZE. SIZE is the total, unexpunged, size of the cross-section set and warns the user of the LCM storage requirements. In this list *DRXS* indicates discrete reaction cross sections are available; *GPXS*, gamma production cross sections are available; and Ψ, total fission $\bar{\nu}$ not available.

Graphic information which compares the master evaluations (such as ENDF/B, ENDL, and AWRE) and the ACE representation of the evaluation is available on 35mm film. Other plotting packages are being developed to help the MCNP user in cross-section selection.

APPENDIX F

X-6 Monte Carlo neutron cross-section library 6Feb78

ZAID	Library	Mat	Size
z = 1	**********		
1001.01	xmccs h-1 lll-howerton 1/73	501	3250
	gpxs = from xsec 12/14/72.		
1001.02	xmccs h-1 lasl dec 1970	5990	2011
	gpxs = from xsec 12/14/72.		
1001.03	xmccs h-1 endf/b-iii (t301)	1148	2459
	gpxs = from xsec 12/14/72.		
1001.04	rmccs h-1 endf/b4 t404 rev.1 11/75	1269	2459
	drxs gpxs = endf/b		2760
1001.30	amccs h-1 endl 7101 t=0k emin=1.0e-4	7101	1444
	gpxs = endf/b		
1002.01	xmccs h-2 lll-howerton 1/73	502	2122
	gpxs = from xsec 12/14/72.		
1002.02	rmccs h-2 uk-lasl dec 1967	2254	3007
	drxs gpxs = from xsec 12/14/72.		
1002.03	xmccs h-2 endf/b-iii (t301)	1120	1963
	gpxs = from xsec 12/14/72.		
1002.04	amccs d-2 endf/b4 t402 rev.1 15oct75	1120	2144
	drxs gpxs = endf/b		2780
1003.01	xmccs h-3 lll-howerton 1/73	503	1111
	gpxs = none		
1003.02	umccs h-3 uk-lasl dec 1967	2252	1483
	drxs gpxs = none		2336
1003.03	rmccs h-3 endf/b-iv t=300k 2/21/78	1169	2114
	gpxs = none		

```
  1003.30  amccs h-3  endl 7103 t=0k emin=1.0e-4      7103   1026
           drxs  gpxs = none

z =  2  ***********************************************************

  2000.01  rmccs he endf/b-iii (t301) 20feb74          1088   1705
           drxs  gpxs = none                                 1976

  2003.01  xmccs he-3 lll-howerton 1/73                 504   1320
                gpxs = none

  2003.02  xmccs he-3       uk  jan 1971               7220    818
                gpxs = none

  2003.03  rmccs he-3 endf/b-iii (t301) 20feb74        1146   1517
           drxs  gpxs = none                                 1916

  2004.01  umccs he-4 lll-howerton 1/73                 505   1217
           drxs  gpxs = none                                 2141

  2004.02  xmccs he-4       lasl 1965                    22   1162
                gpxs = none

  2004.03  rmccs he-4 endf/b-iv t=300k 2/21/78          1270   2407
           drxs  gpxs = none

  2004.30  amccs he4  endl 7105 t=0k emin=1.0e-4        7105   1267
                gpxs = none

z =  3  ***********************************************************

  3006.01  xmccs li-6       awre april 1965            2214   3443
                gpxs = endf/b iii  march 1973

  3006.02  xmccs li-6 endf/b-iii (t301)                1115   3997
                gpxs = endf/b iii  march 1973

  3006.04  xmccs  li-6 endf/b-iv                       1271   4615
                gpxs = endf/b
```

```
  3006.10  rmccs li-6 lasl sublib mat 101 t=0 4/76     101    8294
           drxs  gpxs = endf/b                                 6566

  3006.30  xmccs li-6 endl 7106 0k 9/76 emin=1.0-4    7106    2940
                gpxs = endf/b

  3007.01  xmccs li-7 lll-howerton 1/73                 507    2787
                gpxs = endf/b iii  march 1973

  3007.02  xmccs li-7       awre april 1965            2215    3097
                gpxs = endf/b iii  march 1973

  3007.03  xmccs li-7 endf/b-iii (t301)                1116    3562
                gpxs = endf/b iii  march 1973

  3007.05  rmccs li-7 endf/b4 t404 rev.1 10/75         1272    3751
           drxs  gpxs = endf/b                                 3647

z =  4 ************************************************************

  4009.01  xmccs be-9 lll-howerton 1/73                 509    3843
                gpxs = from xsec 12/14/72.

  4009.02  umccs be-9 endf/b-iii (t301) 20feb74        1154    3321
           drxs  gpxs = from xsec 12/14/72.                    4093

  4009.03  rmccs be-9 lasl sublibrary t=300k 2/16/78    104    7883
           drxs  gpxs = lasl sublibrary

z =  5 ************************************************************

  5000.01  rmccs b lll-howerton 1/73                    510    5058
           drxs  gpxs =.198*gpxs for b-10, 1/73 seamon          3975

  5010.01  xmccs b-10 lll-howerton 1/73                 511    5359
                gpxs = june 6, 1973 seamon

  5010.02  amccs b-10 endf/b-iii (t301) 20feb74        1155    2589
                gpxs = june 6, 1973 seamon
```

```
  5010.03  rmccs  b-10 endf/b-iv                        1273   9241
           drxs   gpxs = endf/b                                 5522

  5011.01  xmccs b-11 lll-howerton 1/73                  565   1516
                 gpxs = none

  5011.02  rmccs b-11 endf/b-iii (t301) 20feb74         1160   5134
           drxs   gpxs = none                                   2390

z =  6  ***************************************************************

  6012.01  xmccs c-12       uk  jan 1969                3006   3098
                 gpxs = from xsec 12/14/72.

  6012.02  xmccs c-12 endf/b-iii (t301)                  1165   6942
                 gpxs = from xsec 12/14/72.

  6012.03  xmccs  c-12 endf/b-iv                         1274   7567
                 gpxs = endf/b

  6012.04  xmccs c-12 webster-11+mat 1165 elas ang       6965   2430
                 gpxs = none

  6012.10  rmccs c-12 lasl sublib mat 102 4/76            102   8309
           drxs   gpxs = endf/b                                 6068

  6012.30  xmccs c-12 endl 7112 0k 9/76 emin=1.0-4       7112   2815
                 gpxs = endf/b

z =  7  ***************************************************************

  7014.01  xmccs n-14 lll-howerton 1/73                   513   9586
                 gpxs = mat=4133 mod4. p.young 7/73

  7014.02  xmccs n-14       lasl-lrl  aug 1970           5513   9211
                 gpxs = mat=4133 mod4. p.young 7/73

  7014.04  rmccs  n-14 endf/b-iv                         1275  21553
           drxs   gpxs = endf/b                                 9698
```

```
  7014.30  amccs n-14 endl 7113 0k 9/76 emin=1.0-4    7113    5148
                gpxs = endf/b

z =  8  ********************************************************

  8016.01  xmccs o-16 lll-howerton 1/73                514    5540
                gpxs = mat=4134 mod2. p.young 7/73

  8016.02  xmccs o-16       lrl  april 1971           5514    4791
                gpxs = mat=4134 mod2. p.young 7/73

  8016.04  rmccs  o-16 endf/b-iv                      1276   21823
           drxs  gpxs = endf/b                                10332

  8016.30  amccs o-16 endl 7114 0k 9/76 emin=1.0-4    7114    4565
           drxs  gpxs = endf/b                                4607

z =  9  ********************************************************

  9019.01  amccs f-19 lll-howerton 1/73                515    3101
           drxs  gpxs= endf/b4 mat=1277 shlaph t411 2/75      4062

  9019.02  xmccs f-19 endf/b4 mat=1277 t411 2/75       1277   26334
                gpxs = endf/b4 mat=1277 shlaph t411 2/75

  9019.03  rmccs  f-19 endf/b-iv                      1277   24464
           drxs  gpxs = endf/b                                8772

z = 11  ********************************************************

 11023.01  rmccs na-23 lll-howerton 1/73               516    6816
           drxs  gpxs =endf/b-iii (mat 1156)  28june72       4940

z = 12  ********************************************************

 12000.01  xmccs mg lll-howerton 1/73                  517    5137
                gpxs = none

 12000.02  rmccs mg endf/b-iii (t302) 20feb74         1014    3771
           drxs  gpxs = none                                  2742
```

```
z = 13  ************************************************************

 13027.01  amccs al-27 lll-howerton 1/73                  518   3915
                gpxs = from xsec 12/14/72.

 13027.02  xmccs al-27      awre  april 1965             2035   5532
                gpxs = from xsec 12/14/72.

 13027.03  xmccs al-27 endf/b-iii (t302)                  1135  18038
                gpxs = from xsec 12/14/72.

 13027.04  rmccs  al-27 endf/b-iv                         1193  32517
           drxs  gpxs = endf/b                                  9546

z = 14  ************************************************************

 14000.01  amccs si lll-howerton 1/73                     519  12371
           drxs  gpxs = from xsec 12/14/72.                     4477

 14000.02  rmccs si endf/b-iii (t302) 20feb74            1151  21632
           drxs  gpxs = from xsec 12/14/72.                     6819

z = 15  ************************************************************

 15031.01  rmccs p-31 lll-howerton 1/73                  520   2842
           drxs  gpxs = none                                    3238

z = 16  ************************************************************

 16032.01  rmccs s-32 lll-howerton 1/73                  521   3252
           drxs  gpxs = none                                    3057

z = 17  ************************************************************

 17000.01  amccs cl lll-howerton 1/73                    522   9854
                gpxs= endf/b-iii (mat 1149) 6/17/72

 17000.02  rmccs cl endf/b-iii (t302) 20feb74           1149  38371
           drxs  gpxs = endf/b-iii(mat 1149) 6/17/72             8061
```

```
z = 18  ************************************************************

 18000.01  rmccs argon lll-howerton 1/73                523   2029
           drxs  gpxs = none                                      2743

z = 19  ************************************************************

 19000.01  rmccs k lll-howerton 1/73                    524   7436
           drxs  gpxs = endf/b-iii (mat 1150)   15june72        3976

z = 20  ************************************************************

 20000.01  amccs ca lll-howerton 1/73                   525   9623
           drxs  gpxs = endf/b-iii (mat 1152)   16june72        4303

 20000.10  rmccs ca endf/b4 1165 0k 9/76 emin=1.-5      1195  24085
           drxs  gpxs = endf/b                                    9022

z = 22  ************************************************************

 22000.01  amccs ti lll-howerton 1/73                   526   9259
           drxs  gpxs = none                                      2926

 22000.02  xmccs ti        uk  april 1965              2190   3884
                 gpxs = none

 22000.11  rmccs ti endf/b4 1286 t=300.0k 5/77         1286  10644
           drxs  gpxs = endf/b                                    3743

z = 23  ************************************************************

 23000.30  rmccs v endf/b4 1196 0k 9/76 emin=1.0-5      1196   6456
           drxs  gpxs = endf/b                                    4449

z = 24  ************************************************************

 24000.01  amccs cr        uk  may 1966                2045   3667
           drxs  gpxs = none                                      3422

 24000.11  rmccs cr endf/b4 t=300.0 24feb76             1191  38240
```

```
          drxs  gpxs = endf/b                                    11613

 24000.12 xmccs cr endf/b4 t=900.0 24feb76              1191   51663
          gpxs = endf/b

z = 25  ************************************************************

 25055.01 rmccs mn-55 lll-howerton 1/73                  527    3586
          drxs  gpxs = none                                      2967

z = 26  ************************************************************

 26000.01 xmccs fe lll-howerton 1/73                     528    4102
          gpxs = from xsec 12/14/72.

 26000.03 xmccs fe   dna  mat=4180mod2,  feb 1974        4180   62886
          gpxs = from xsec 12/14/72.

 26000.11 rmccs fe endf/b4 t=300.0 24feb76               1192   54104
          drxs  gpxs = endf/b                                    8698

 26000.12 xmccs fe endf/b4 t=900.0 24feb76               1192   57638
          gpxs = endf/b

 26000.30 amccs fe endl 7132 0k 9/76 emin=1.0-4          7132   23179
          drxs  gpxs = endf/b                                    3925

z = 28  ************************************************************

 28000.01 amccs ni        uk  oct 1965                  2046    5714
          drxs  gpxs = none                                      3244

 28000.11 rmccs ni endf/b4 t=300.0 24feb76               1190   35192
          drxs  gpxs = endf/b                                    5504

 28000.12 xmccs ni endf/b4 t=900.0 24feb76               1190   40842
          gpxs = endf/b

 28058.01 rmccs ni-58 lll-howerton 1/73                  529    5347
          drxs  gpxs = none                                      3375
```

```
z = 29  ************************************************************

 29000.01  amccs cu lll-howerton 1/73                    530   3629
           drxs  gpxs = from xsec 12/14/72.                      4273

 29000.02  xmccs cu        uk 1967                      2249   6476
                 gpxs = from xsec 12/14/72.

 29000.10  rmccs cu endf/b-iv 1295 0k 9/76              1295  14703
           drxs  gpxs = endf/b                                 8456

z = 31  ************************************************************

 31000.01  rmccs ga lll-howerton 1/73                    531   3730
           drxs  gpxs = none                                   2775

z = 40  ************************************************************

 40000.01  amccs zr        uk  nov 1965                 2009   3837
           drxs  gpxs = none                                   1877

 40000.02  rmccs zr endl mat7141 howerton-lll 3/75       7141  10312
           drxs  gpxs = endl 3/75 mat7141 w/laphan0 5/75        4157

z = 41  ************************************************************

 41093.01  amccs nb-93 lll-howerton 1/73                 532   5880
           drxs  gpxs = none                                   3138

 41093.30  rmccs nb-93 endl 7143 0k 9/76 emin=1.-4       7143  29725
           drxs  gpxs = endf/b                                 5751

z = 42  ************************************************************

 42000.01  rmccs mo lll-howerton 1/73                    533   5714
           drxs  gpxs = t-2 phlag program  april 1973          3796

z = 48  ************************************************************

 48000.01  rmccs cd lll-howerton 1/73                    534   7690
```

```
          drxs  gpxs = none                                        2839

z = 50  ********************************************************************

 50000.01  rmccs sn 111-howerton 1/73                      535    2332
          drxs  gpxs = none                                        2993

 50999.02  rmccs fission products crude endl 1/73          558    1647
          drxs  gpxs = none                                        2708

z = 56  ********************************************************************

 56138.01  rmccs ba-138 111-howerton 1/73                  536    2606
          drxs  gpxs = none                                        3069

z = 63  ********************************************************************

 63000.01  rmccs eu 111-howerton 1/73                      537    3133
          drxs  gpxs = none                                        2885

z = 64  ********************************************************************

 64000.01  rmccs gd 111-howerton 1/73                      538    3206
          drxs  gpxs = none                                        2946

z = 67  ********************************************************************

 67165.01  rmccs ho-165 111-howerton 1/73                  539    3626
          drxs  gpxs = none                                        3157

z = 73  ********************************************************************

 73181.01  amccs ta-181    lrl  oct 1966                  8731    2431
          drxs  gpxs = none                                        2968

 73181.02  rmccs ta-181 111-howerton 1/73                  540   18114
          drxs  gpxs = none                                        4240

z = 74  ********************************************************************
```

```
 74000.01  rmccs w lll-howerton 1/73                    541   3237
           drxs  gpxs = from mats 328,329,330,331 11sep73   3896

 74182.10  rmccs w-182 endf/b4 t=0.0 24feb76           1128  33247
           drxs  gpxs = endf/b                                5766

 74183.10  rmccs w-183 endf/b4 t=0.0 24feb76           1129  27816
           drxs  gpxs = endf/b                                6971

 74184.10  rmccs w-184 endf/b4 t=0.0 24feb76           1130  27996
           drxs  gpxs = endf/b                                5985

 74186.10  rmccs w-186 endf/b4 t=0.0 24feb76           1131  30916
           drxs  gpxs = endf/b                                6054

z = 78  ************************************************************

 78000.01  rmccs pt lll-howerton 1/73                    566  10313
           drxs  gpxs = none                                  3049

z = 79  ************************************************************

 79197.01  xmccs au-197 lll-howerton 1/73                542   3561
                 gpxs = none

 79197.02  xmccs au-197    uk  oct 1965                 2222   2660
                 gpxs = none

 79197.10  rmccs au-197 endl 7163 t=0. 4/76             7163  19844
           drxs  gpxs = endf/b                                4519

z = 82  ************************************************************

 82000.01  amccs pb lll-howerton 1/73                    543   3148
           drxs  gpxs = from xsec 12/14/72.                   3902

 82000.02  xmccs pb        awre  april 1965             2026   2192
                 gpxs = from xsec 12/14/72.

 82000.10  rmccs pb endf/b-iv 0k 9/76                   1288  21052
```

```
          drxs  gpxs = endf/b                                    11376

z = 90  ********************************************************************

 90232.01  xmccs th-232 lll-howerton 1/73 Ψ              544   4239
           gpxs = none

 90232.02  xmccs th-232 awre april 1965 Ψ               2022   2433
           gpxs = none

 90232.10  rmccs th-232 endl 7165 13apr76 t=0.0 Ψ       7165  31235
           drxs  gpxs = endf/b                                  4237

z = 92  ********************************************************************

 92233.01  xmccs u-233 lll-howerton 1/73 Ψ               545   4591
           gpxs = none

 92233.10  rmccs u-233 endl 7166 13apr76 t=0.0 Ψ        7166   7945
           drxs  gpxs = endf/b                                  4393

 92234.01  xmccs u-234 lll-howerton 1/73 Ψ               546   2750
           gpxs = none

 92234.10  rmccs u-234 endl 7167 13apr76 t=0.0 Ψ        7167   3008
           drxs  gpxs = endf/b                                  4581

 92235.01  amccs u-235 lll-howerton 1/73 Ψ               547  12770
           drxs  gpxs = mat=7052 lll 10/73 t=0. 6/75            4985

 92235.04  xmccs u-235 endf/b4 t407 t=3.0e+3 2/75       1261  29516
           gpxs = u-235 t=3.0e+3 9jun75

 92235.05  xmccs u-235 endf/b4 t407 t=3.0e+4 2/75       1261  18573
           gpxs = u-235 t=3.0e+4 9jun75

 92235.06  xmccs u-235 endf/b4 t407 t=6.0e+5 2/75       1261  12560
           gpxs = u-235 t=6.0e+5 9jun75

 92235.07  xmccs u-235 endf/b4 t407 t=1.2e+7 2/75       1261  11268
```

```
                gpxs = u-235 t=1.2e+7 9jun75

92235.08  xmccs u-235 endf/b4 t407 t=0.0e+0 2/75    1261  42923
                gpxs = u-235 t=0.0 9jun75

92235.09  xmccs u-235 endf/b4 t407 t=3.0e+2k 2/75   1261  41638
                gpxs = u-235 t=3.0e+2 9jun75

92235.10  amccs u-235 endf/b4 407 t=0. 1/76         1261  42716
          drxs  gpxs = endf/b                             7236

92235.11  amccs u-235 endf/b4 407 t=300. 1/76       1261  41332
          drxs  gpxs = endf/b                             7236

92235.15  amccs u-235 endf/b4 407 t=12.e+6 1/76     1261  11267
          drxs  gpxs = endf/b                             7236

92235.18  amccs u-235 endf/b4 t=3000.0 24feb76      1261  29254
          drxs  gpxs = endf/b                             7236

92235.19  rmccs u-235 endf/b4 t=3*e+4 24feb76       1261  18385
          drxs  gpxs = endf/b                             7236

92235.20  amccs u-235 endf/b4 t=6*e+5 24feb76       1261  12497
          drxs  gpxs = endf/b                             7236

92235.30  amccs u-235 endl 0k 14sep76 Ψ             7168  17940
          drxs  gpxs = endf/b                             4931

92236.01  rmccs u-236 lll-howerton 1/73 Ψ            548   3153
          drxs  gpxs = none                               3908

92237.01  rmccs u-237 lll-howerton 1/73 Ψ            549   2478
          drxs  gpxs = none                               3604

92238.01  amccs u-238 lll-howerton 1/73 Ψ            550   4703
          drxs  gpxs = endf/b (stewart,hunter) 12/72      5115

92238.04  xmccs u-238 endf/b4 t409 t=3.0e+4k 2/75    1262  32918
                gpxs = u-238 t=3.0e+4 9jun75
```

```
92238.05  xmccs u-238 endf/b4 t409 t=6.0e+5 2/75     1262  18803
                gpxs = u-238 t=6.0e+5 9jun75

92238.06  xmccs u-238 endf/b4 t409 t=1.2e+7 2/75     1262  10399
                gpxs = u-238 t=1.2e+7 9jun75

92238.12  umccs u-238 endf/b4 300k 8/77 10%thin      1262  50412
                gpxs = endf/b

92238.13  rmccs u-238 endf/b4 t409 t=3.e+4 1/78      1262  32895
          drxs  gpxs = endf/b                                6799

92238.15  amccs u-238 endf/b4 t409 t=12.e+6 1/76     1262  10351
          drxs  gpxs = endf/b                                6864

92238.20  amccs u-238 endf/b4 t=6e+5 24feb76         1262  18721
          drxs  gpxs = endf/b                                6858

92238.30  amccs u-238 endl 0k 14sep76 Ψ              7171  15110
          drxs  gpxs = endf/b                                6639

92239.01  rmccs u-239 lll-howerton 1/73 Ψ             551   3227
          drxs  gpxs = none                                  3767

92240.01  rmccs u-240 lll-howerton 1/73 Ψ             552   2777
          drxs  gpxs = none                                  3833

z = 94  ****************************************************************

94238.01  rmccs pu-238 lll-howerton 1/73 Ψ            553   2588
          drxs  gpxs = none                                  3495

94239.01  amccs pu-239 lll-howerton 1/73 Ψ            554   6077
          drxs  gpxs = endf mat 304 stewart,hunter 12/72     5772

94239.02  xmccs pu-239 endf/b4 t407 t=3.0e+3 2/75     1264  40464
                gpxs = pu-239 t=3.0e+3 9jun75

94239.03  xmccs pu-239 endf/b4 t407 t=3.0e+4 2/75     1264  25460
                gpxs = pu-239 t=3.0e+4 9jun75
```

94239.04	xmccs	pu-239 endf/b4 t407 t=6.0e+5 2/75	1264	13633
		gpxs = pu-239 t=6.0e+5 9jun75		
94239.05	xmccs	pu-239 endf/b4 t407 t=1.2e+7 2/75	1264	11349
		gpxs = pu-239 t=1.2e+7 9jun75		
94239.06	xmccs	pu-239 endf/b4 t407 t=0.0e+0 2/75	1264	41167
		gpxs = pu-239 t=0.0 9jun75		
94239.07	xmccs	pu-239 endf/b4 t407 t=3.0e+2 2/75	1264	34659
		gpxs = pu-239 t=3.0e+2 9jun75		
94239.15	amccs	pu-239 endf/b4 407 t=12.e+6 1/76	1264	11297
	drxs	gpxs = endf/b		7880
94239.16	amccs	pu-239 endf/b4 t=0 24feb76	1264	41153
	drxs	gpxs = endf/b		7880
94239.17	amccs	pu-239 endf/b4 t=300 24feb76	1264	34631
	drxs	gpxs = endf/b		7880
94239.18	amccs	pu-239 endf/b4 t=3000 24feb76	1264	40421
	drxs	gpxs = endf/b		7880
94239.19	rmccs	pu-239 endf/b4 t=3*e+4 24feb76	1264	25417
	drxs	gpxs = endf/b		7880
94239.20	amccs	pu-239 endf/b4 600000k 24feb76	1264	13590
	drxs	gpxs = endf/b		7880
94239.99	xmccs	pu-239 lrl 1/65 Ψ	942	3833
		gpxs =endf mat 304 stewart,hunter 12/72		
94240.01	amccs	pu-240 lll-howerton 1/73 Ψ	555	3653
	drxs	df mat 305 stewart hunter 12/72		4640
94240.12	rmccs	pu-240 endf/b-iv 900k 14sep76	1265	41821
	drxs	gpxs = endf/b		5911
94241.01	rmccs	pu-241 lll-howerton 1/73 Ψ	556	3443

```
           drxs  gpxs = none                                        3955

z = 95  ***************************************************************

 95242.01  rmccs am-242 lll-howerton 1/73 Ψ              557    5777
           drxs  gpxs = none                                        3641

z = 99  ***************************************************************

 99003.01  umccs bm-3 boredum test data                   1    1039
                 gpxs = none


Note*************************************************************

ZAID  = X-6 cross-section library identification number
Mat   = evaluators Material identification no.
Size  = words10 in cross-section set

length of rmccs =   894976
length of amccs =   487424
length of xmccs =   797866
length of umccs =    57691
Ψ = total ν̄ not available for nuclide
```

REFERENCES

1. D. Garber, C. Dunford, and S. Pearlstein, "Data Formats and Procedures for the Evaluated Nuclear Data File, ENDF," BNL-NCS-50496 (1975).

2. R. J. Howerton et al., "An Integrated System for Production of Neutronics and Photonics Calculational Constants," UCRL-50400 Series (May 1978).

3. E. Storm and H. I. Israel, "Photon Cross Sections from 0.001 to 100 Mev for Elements 1 through 100," LA-3753 (1967).

4. L. L. Carter and E. D. Cashwell, "Particle-Transport Simulation with the Monte Carlo Method," *ERDA Critical Review Series*, TID-26607 (1975).

5. "Fermi Invention Rediscovered at LASL," *The Atom*, Los Alamos Scientific Laboratory (October 1966).

6. E. D. Cashwell and C. J. Everett, *A Practical Manual on the Monte Carlo Method for Random Walk Problems*, Pergamon Press, Inc., New York (1959). [Also LA-2120, (1957)].

7. R. R. Johnston, "A General Monte Carlo Neutronics Code," LAMS-2856 (1963).

8. E. D. Cashwell, J. R. Neergaard, W. M. Taylor, and G. D. Turner, "MCN: A Neutron Monte Carlo Code," LA-4751 (1972).

9. E. D. Cashwell, J. R. Neergaard, C. J. Everett, R. G. Schrandt, W. M. Taylor, and G. D. Turner, "Monte Carlo Photon Codes: MCG and MCP," LA-5157-MS (1973).

10. E. D. Cashwell and C. J. Everett, "Intersection of a Ray with a Surface of Third or Fourth Degree," LA-4299 (1969).

11. H. J. Kalli and E. D. Cashwell, "Evaluation of Three Monte Carlo Estimation Schemes for Flux at a Point," LA-6865-MS (1977).

12. J. E. Stewart, "A General Point-on-a-Ring Detector," *Transactions of the American Nuclear Society*, 28, 643 (June 1978).

13. R. A. Forster, "Ring Detector and Angle Biasing," Internal LASL Memo TD-6-8-79 (July 1979).

14. F. H. Clark, "The Exponential Transform as an Importance-Sampling Device, A Review," ORNL-RSIC-14 (January 1966).

15. P. K. Sarkar and M. A. Prasad, "Prediction of Statistical Error and Optimization of Biased Monte Carlo Transport Calculations," *Nuclear Science and Engineering*, 70, 243-261 (1979).

16. J. S. Hendricks, "Calculation of Cell Volumes and Surface Areas in MCNP," LA-8113-MS (soon to be issued).

17. C. J. Everett and E. D. Cashwell, "MCP Code Fluorescence-Routine Revision," LA-5240-MS (1973).

18. C. J. Everett and E. D. Cashwell, "A New Method of Sampling the Klein-Nishina Probability Distribution for All Incident Photon Energies Above 1 keV (A Revised Complete Account)," LA-7188-MS (1978).

☆ U.S. Government Printing Office: 1979 — 677—115/224