LA-10363-MS

A Sample Problem for Variance Reduction in MCNP

Los Alamos
NATIONAL LABORATORY

Los Alamos National Laboratory is operated by the University of California for the United States Department of Energy under contract W-7405-ENG-36.

An Affirmative Action/Equal Opportunity Employer

LA-10363-MS

UC-32
Issued: October 1985

A Sample Problem for Variance Reduction in MCNP

Thomas E. Booth

Los Alamos Los Alamos National Laboratory
Los Alamos, New Mexico 87545

A SAMPLE PROBLEM FOR VARIANCE REDUCTION IN MCNP

by

Thomas E. Booth

ABSTRACT

The Los Alamos computer code Monte Carlo Neutron Photon (MCNP) has many useful variance reduction techniques to aid the Monte Carlo user. This report applies many of these techniques to a conceptually simple but computationally demanding neutron transport problem.

I. INTRODUCTION

This report is based on a series of four 50-min variance reduction talks ("MCNP Variance Reduction Techniques," video reels #12-15) given at the Magnetic Fusion Energy Conference on MCNP,* Los Alamos National Laboratory, October 1983. It is an overview of all variance reduction techniques in MCNP and not an in-depth consideration of any. In fact, the techniques are described only in the context of a single conceptually simple, but demanding, neutron transport problem, with only enough theory presented to describe the general flavor of the techniques. Detailed descriptions are in the MCNP manual.[1]

This report assumes a general familiarity with Monte Carlo transport vocabulary such as weight, roulette, score, bias, etc.

II. VARIANCE REDUCTION

Variance-reducing techniques in Monte Carlo calculations can often reduce the computer time required to obtain results of sufficient *precision.* Note that precision is only one requirement for a good Monte Carlo calculation. Even a zero variance Monte Carlo calculation cannot accurately predict natural behavior if other sources of error are not minimized. Factors affecting accuracy were outlined by Art Forster, Los Alamos (Fig. 1).**

This paper demonstrates how variance reduction techniques can increase the efficiency of a Monte Carlo calculation. Two user choices affect that efficiency, the choice of tally and of random walk sampling. The tally choice (of for example, collision vs track length estimators) amounts to trying to obtain the best results from the random walks sampled. The chosen random walk sampling amounts to preferentially sampling "important" particles at the expense of "unimportant" particles.

A. Figure of Merit

The measure of efficiency for MCNP calculations is the figure of merit (FOM) defined as

$$\mathrm{FOM} = \frac{1}{\sigma_{mr}^2\,T} ,$$

*Videotapes of the entire conference are available from Radiation Shielding Information Center, Oak Ridge National Laboratory, Oak Ridge, TN 37830. The reader wishing to run the sample problem here should refer to the appendix beginning on page 67 for input file details modified since the conference and after the writing of this report.

**Video reel #11, "Relative Errors, Figure of Merit" from MCNP Workshop, Los Alamos National Laboratory, October 4-7, 1983. Available from Radiation Shielding Information Center, Oak Ridge National Laboratory, Oak Ridge, TN 37830.

1. CODE FACTORS

PHYSICS AND MODELS
DATA UNCERTAINTIES
CROSS-SECTION REPRESENTATION
ERRORS IN THE CODING

2. PROBLEM-MODELING FACTORS

SOURCE MODEL AND DATA
GEOMETRICAL CONFIGURATION
MATERIAL COMPOSITION

3. USER FACTORS

USER-SUPPLIED SUBROUTINE ERRORS
INPUT ERRORS
VARIANCE REDUCTION ABUSE
CHECKING THE OUTPUT
UNDERSTANDING THE PHYSICAL MEASUREMENT

Fig. 1. Factors affecting accuracy.

where σ_{mr} = relative standard deviation of the mean and T = computer time for the calculation (in minutes). The FOM should be roughly constant for a well-sampled problem because σ_{mr}^2 is (on average) proportional to N^{-1} (N = number of histories) and T is (on average) proportional to N; therefore, the product remains approximately constant.

B. General Comments

Although all variance reduction schemes have some unique features, a few general comments are worthwhile. Consider the problem of decreasing

$$\sigma_{mr} = \frac{\sigma}{\sqrt{N}}/\mu \ ,$$

(where σ^2 = history variance, N = number of particles, and μ = mean) for fixed computer time T. To decrease σ_{mr}, we can try to decrease σ or increase N—that is, decrease the time per particle history—or both. Unfortunately, these two goals usually conflict because decreasing σ normally requires more time per history because better information is required and increasing N normally increases σ because there is less time per history to obtain information. However, the situation is not hopeless. It is often possible to decrease σ substantially without decreasing N too much or increase N substantially without increasing σ too much so that

$$\sigma_{mr} = \frac{\sigma}{\mu \sqrt{N}}$$

decreases.

Many techniques described here attempt to decrease σ_{mr} by either producing or destroying particles. Some techniques do both. In general, (1) techniques that produce tracks work by decreasing σ (we hope much faster than N decreases), and (2) techniques that destroy tracks work by increasing N (we hope much faster than σ increases).

III. THE PROBLEM

The problem is illustrated in Fig. 3, but before discussing its Monte Carlo aspects, I must point out that the problem is atypical and not real. I invented the sample problem so most of the MCNP variance reduction techniques could be applied. Usually, a real problem will not need so many techniques. Futhermore, without understanding and caution, "variance-reducing" techniques often increase the variance.

Figure 2 is the input file for an analog MCNP calculation and Fig. 3 is a slice through the geometry at z = 0.

```
SAMPLE PROBLEM FOR MFE TALKS
C       TALLIES FOR PARTICLES WITH E>.01MEV
    1 0  (1 -21):-2
 2      1 -2.03E0  -1  -3   2
 3      1 -2.03E0  -1  -4   3
 4      1 -2.03E0  -1  -5   4
 5      1 -2.03E0  -1  -6   5
 6      1 -2.03E0  -1  -7   6
 7      1 -2.03E0  -1  -8   7
 8      1 -2.03E0  -1  -9   8
 9      1 -2.03E0  -1  -10  9
10      1 -2.03E0  -1  -11  10
11      1 -2.03E0  -1  -12  11
12      1 -2.03E0  -1  -13  12
13      1 -2.03E0  -1  -14  13
14      1 -2.03E0  -1  -15  14
15      1 -2.03E0  -1  -16  15
16      1 -2.03E0  -1  -17  16
17      1 -2.03E0  -1  -18  17
18      1 -2.03E0  -1  -19  18
19      1 -2.03E0  -1  -20  19
20      0  -1  -21  20
21      1 -2.03E-2  -1   -22  21
   22 0 1 21 -22
   23 0 22

 1      CY   100
 2      PY   0
 3      PY   10
 4      PY   20
 5      PY   30
 6      PY   40
 7      PY   50
 8      PY   60
 9      PY   70
10      PY   80
11      PY   90
12      PY   100
13      PY   110
14      PY   120
15      PY   130
16      PY   140
17      PY   150
18      PY   160
19      PY   170
20      PY   180
21      PY   2000
22      PY   2010

MODE    0
C         THE FOLLOWING IS SCHAEFFER PORTLAND CONCRETE
M1      1001   -.010
        8016   -.529
        11023 -.016
        12000 -.002
        13027 -.034
        14000 -.337
        19000 -.013
        20000 -.044
        26000 -.014
        6012   -.001
SRC1    0 1.E-6 0 2 1.0
SI      2 2  14 14
SP      0 .5 .5 1
NPS     100000
IN      0 1 3R 15R 2R 0
F1      20
F4      21
F5      200 2005 0 0
PDO     0 19R 1 0 0<———ONLY CELL 21 CONTRIBUTES TO POINT DETECTOR TALLY
EO      .01 100
TO      100 1000  10000
CUTN    1.0E123 0.0  0 0
CTME    45
PRDMP   -5 -5
PRINT
```

Fig. 2. Input file for an analog MCNP calculation.

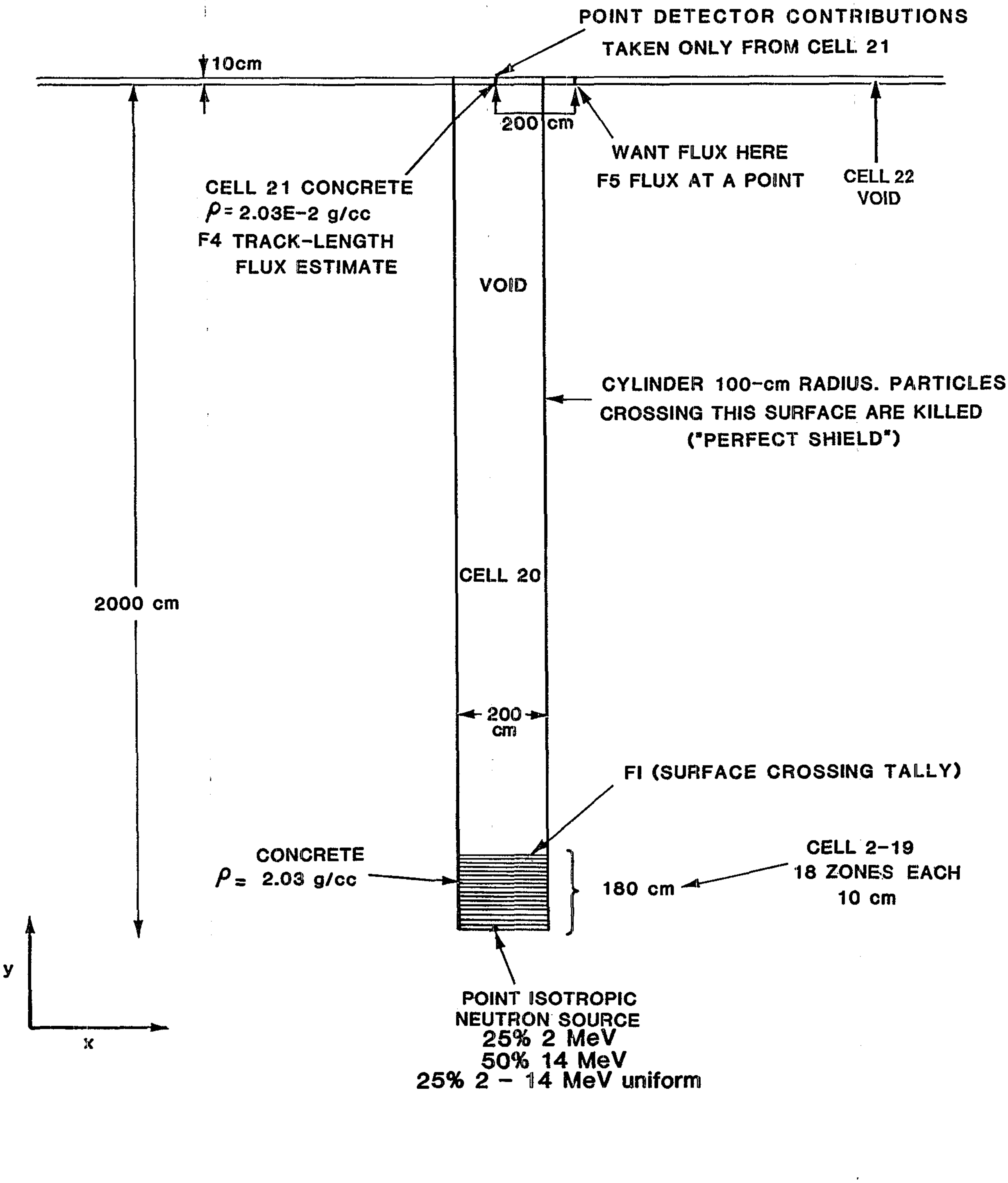


Fig. 3. The problem.

The primary tally is the point detector tally (F5) at the top of Fig. 3, 200 cm from the axis of the cylinder (y-axis). A point isotropic neutron source is just barely inside the first cell (cell 2) at the bottom of Fig. 3. The source energy distribution is 25% at 2 MeV, 50% at 14 MeV, and 25% uniformly distributed between 2 and 14 MeV. For this problem, the detectors will respond only to neutrons above 0.01 MeV.

A "perfect shield" immediately kills any neutrons leaving the cylinder (except from cell 21 to cell 22). Thus, to tally (F5), a neutron must

1. penetrate 180 cm of concrete (cells 2-19),
2. leave the concrete (cell 19) with a direction close enough to the cylinder axis that the neutron goes from the bottom of cell 20 (the cylindrical void) to the top and crosses into cell 21,
3. collide in cell 21 (because point detector contributions are made only from collision/source points), and
4. have energy above 0.01 MeV.

These events are unlikely because

1. 180 cm of 2.03-g/cm^3 concrete is difficult to penetrate,
2. there is only a small solid angle up the "pipe" (cell 20),
3. not many collisions will occur in 10 cm of 0.0203-g/cm^3 concrete, and
4. particles lose energy penetrating the concrete.

Before approaching these four problems, knowledge about the the point detector technique can be applied to keep from wasting time; only collisions in cell 21 can contribute to the point detector. Collisions in cells 2-19 cannot contribute through the perfect shield, that is, zero importance region. Thus, the MCNP input is set (PD0 card, Fig. 2) so that the point detector ignores collisions not in cell 21. If the point detector did not ignore collisions in cells 2-19, the following would happen at each collision.

1. The probablity density for scattering toward the point detector would be calculated.
2. A point detector pseudoparticle would be created and pointed toward the point detector.
3. The pseudoparticle would be tracked and exponentially attenuated through the concrete.
4. The pseudoparticle would eventually enter the perfect shield (cell 1) and be killed because a straight line from any point in cells 2-19 to the point detector would enter the perfect shield.

There is no point proceeding with these steps because the pseudoparticles from cells 2-19 are always killed; time is saved by ignoring point detector contributions from cells 2-19.

IV. ANALOG CALCULATION

Inspection of Fig. 4, which is derived from MCNP summary tables, shows that the analog calculation fails. Note that the tracks entering dwindle to zero as they try to penetrate the concrete (cells 2-19). This problem will be addressed in more detail later, but first note that the number weighted energy (NWE) is very low, especially in cells 12, 13, and 14. The NWE is simply the average energy, that is

$$\mathrm{NWE} = \frac{\int N(E)E\ dE}{\int N(E)\ dE} ,$$

where E = energy and N(E) = number density at energy E. This indicates that there are many neutrons below 0.01 MeV that the point detector will not respond to. There is no sense following particles too low in energy to contribute; therefore, MCNP kills neutrons when they fall below a user-supplied energy cutoff.

V. ENERGY AND TIME CUTOFFS

A. Energy Cutoff

The energy cutoff in MCNP is a single user-supplied problem-wide energy level. Particles are terminated when their energy falls below the energy cutoff. The energy cutoff terminates tracks and thus decreases the time per history. The energy cutoff should be used only when it is *known* that low-energy particles are either of zero importance or almost zero importance. A number of pitfalls exist.

1. Remember that low-energy particles can often produce high-energy particles (for example, fission or low-energy neutrons inducing high-energy photons). Thus, even if a detector is not sensitive to low-energy particles, the low-energy particles may be important to the tally.
2. The energy cutoff is the same throughout the entire problem. Often low-energy particles have zero importance in some regions and high importance in others.
3. The answer will be biased (low) if the energy cutoff is killing particles that might otherwise have contributed. Furthermore, as N$\rightarrow\infty$ the apparent error will go to zero and therefore mislead the unwary. Serious consideration should be given to two techniques (discussed later), energy roulette and space-energy weight window, that are always unbiased.

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	4783	3931	13949	3.5593E+00	2.4144E-03	4.7075E+00	1.0000E+00	5.8207E+00
3	3	2176	931	15057	3.8421E+00	4.5943E-04	1.9643E+00	1.0000E+00	3.9404E+00
4	4	1563	593	12510	3.1921E+00	2.0566E-04	1.2067E+00	1.0000E+00	3.3058E+00
5	5	939	362	7390	1.8857E+00	1.4450E-04	8.4454E-01	1.0000E+00	3.0062E+00
6	6	511	205	4213	1.0750E+00	9.3995E-05	5.6654E-01	1.0000E+00	2.7411E+00
7	7	287	115	2219	5.6622E-01	1.0022E-04	5.8205E-01	1.0000E+00	2.7733E+00
8	8	170	63	1587	4.0495E-01	6.4696E-05	4.4866E-01	1.0000E+00	2.5496E+00
9	9	87	40	961	2.4522E-01	6.2827E-05	4.6476E-01	1.0000E+00	2.6046E+00
10	10	44	16	304	7.7571E-02	1.0691E-04	5.1448E-01	1.0000E+00	3.0390E+00
11	11	31	10	230	5.8688E-02	6.2272E-05	2.4500E-01	1.0000E+00	2.4143E+00
12	12	31	7	330	8.4205E-02	2.2207E-05	1.1767E-01	1.0000E+00	2.1161E+00
13	13	18	6	218	5.5626E-02	1.9931E-06	6.6168E-04	1.0000E+00	1.9191E+00
14	14	4	2	17	4.3378E-03	3.7686E-06	7.9961E-04	1.0000E+00	2.0823E+00
15	15	0	0	0	0.	0.	0.	0.	0.
16	16	0	0	0	0.	0.	0.	0.	0.
17	17	0	0	0	0.	0.	0.	0.	0.
18	18	0	0	0	0.	0.	0.	0.	0.
19	19	0	0	0	0.	0.	0.	0.	0.
20	20	0	0	0	0.	0.	0.	0.	0.
21	21	0	0	0	0.	0.	0.	0.	0.
22	22	0	0	0	0.	0.	0.	0.	0.
TOTAL		10644	6281	58985	1.5051E+01				

ANALOG CALCULATION – NO VARIANCE REDUCTION TECHNIQUES

	TALLY 1			TALLY 4			TALLY 5		
NPS	MEAN	ERROR	FOM	MEAN	ERROR	FOM	MEAN	ERROR	FOM
1000	0.	0.0000	0	0.	0.0000	0	0.	0.0000	0
2000	0.	0.0000	0	0.	0.0000	0	0.	0.0000	0
3000	0.	0.0000	0	0.	0.0000	0	0.	0.0000	0
3919	0.	0.0000	0	0.	0.0000	0	0.	0.0000	0

```
**************************************************************************
DUMP NO.   2 ON FILE RUNTPE          NPS =     3919          CTM =     .61
```

Fig. 4. Analog calculation.

B. Time Cutoff

The time cutoff in MCNP is a single user-supplied, problem-wide time value. Particles are terminated when their time exceeds the time cutoff. The time cutoff terminates tracks and thus decreases the computer time per history. The time cutoff should only be used in time-dependent problems where the last time bin will be earlier than the cutoff.

The sample problem in this report is time-independent, so the time cutoff is not demonstrated here.

C. The Sample Problem with Energy Cutoff

Figure 5 gives the results of an MCNP calculation with a 0.01-MeV energy cutoff. Note that the number weighted energy is about 1000 times higher, so the energy cutoff has changed the energy spectrum as expected. Furthermore, note that about four times as many histories were run in the same time although the total number of collisions is approximately constant.

Despite more histories, fewer tracks enter deep into the concrete cylinder. This may seem a little counterintuitive until one remembers that the energy cutoff kills the typical particle that has had many collisions and is below the energy cutoff, that is, the typical particle deep in the concrete. This decrease in the tracks entering is not alarming because we know that only tracks with energy less than 0.01 MeV were killed and they cannot tally.

The trouble with the calculation is that the large amount of concrete is preventing neutron travel from the source to the tally region. The solution is to preferentially push particles up the cylinder. Four techniques in MCNP can be used for penetration,

1. geometry splitting/Russian roulette,
2. exponential transorm,
3. forced collisions,* and
4. weight window.

VI. GEOMETRY SPLITTING AND RUSSIAN ROULETTE

Geometry splitting/Russian roulette is one of the oldest, most widely used variance reduction techniques. As with most biasing techniques, the objective is to spend more time sampling important (spatial) cells and less time sampling unimportant cells. The technique (Fig. 6) is to

1. divide the geometry into cells;
2. assign importances (I_n) to these cells; and
3. when crossing from cell m to cell n, compute $v = I_n/I_m$. If
   a. $v = 1$, continue transport;
   b. $v < 1$, play Russian roulette,
   c. $v > 1$, split the particle into $v = I_n/I_m$ tracks.

A. Russian Roulette ($v < 1$)

If $v < 1$, the particle is entering a cell that we wish to sample less frequently, so the particle plays Russian roulette. That is,

1. with probability v, the particle survives and its weight is multiplied by v^{-1}, or
2. with probability $1 - v$ the particle is killed.

In general, Russian roulette increases the history variance but decreases the time per history, allowing more histories to be run.

B. Splitting ($v > 1$)

If $v > 1$, the particle is entering a more important region and is split into "v" subparticles. If v is an integer, this is easy to do; otherwise v must be sampled. Consider $n < v < n + 1$, then

Probability	Split Weight	
$p(n) = n + 1 - v$	$wt_s = wt/n$	sampled
$p(n+1) = v - n$	$wt_s = wt/(n + 1)$	splitting

The sampled splitting scheme above conserves the total weight crossing the splitting surface, but the split weight varies, depending on whether n or n + 1 particles are selected.

Actually, MCNP does not use the sampled splitting scheme. MCNP uses an expected value scheme:

Probability	Split Weight	
$p(n) = n + 1 - v$	$wt_s = wt/v$	expected value
$p(n + 1) = v - n$	$wt_s = wt/v$	splitting

The MCNP scheme does not conserve weight crossing a splitting surface at each occurrence. That is, if n particles are sampled, the total weight entering is

$$n \cdot \frac{wt}{v} = \frac{n}{v} \cdot wt < wt ,$$

but if n + 1 particles are sampled, the total weight entering is

$$(n + 1) \frac{wt}{v} = \frac{n + 1}{v} wt > wt .$$

However, the *expected* weight crossing the surface is wt:

$$p(n) \cdot n \cdot \frac{wt}{v} + p(n + 1) \cdot (n + 1) \cdot \frac{wt}{v} = wt .$$

*There will not be an example using forced collisions for penetration problems because it is awkward to do in MCNP. In fact, an alteration to the weight cutoff game is often necessary.

CELL PROGR	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	15416	14004	27380	1.9602E+00	2.2661E+00	6.0830E+00	1.0000E+00	6.9686E+00
3	3	4445	3098	15611	1.1176E+00	1.0718E+00	3.9688E+00	1.0000E+00	5.9464E+00
4	4	2197	1580	7830	5.6057E-01	8.8425E-01	3.4412E+00	1.0000E+00	5.6866E+00
5	5	973	716	3661	2.6210E-01	7.9762E-01	2.9569E+00	1.0000E+00	5.5260E+00
6	6	467	331	1726	1.2357E-01	7.2799E-01	2.6838E+00	1.0000E+00	5.4005E+00
7	7	233	171	765	5.4768E-02	8.0105E-01	2.6783E+00	1.0000E+00	5.5811E+00
8	8	110	85	420	3.0069E-02	7.4818E-01	2.4966E+00	1.0000E+00	5.5021E+00
9	9	56	43	186	1.3316E-02	9.0855E-01	2.6749E+00	1.0000E+00	5.6290E+00
10	10	40	24	155	1.1097E-02	5.8161E-01	1.7610E+00	1.0000E+00	4.9657E+00
11	11	20	15	78	5.5842E-03	5.3100E-01	1.6936E+00	1.0000E+00	4.4423E+00
12	12	8	7	42	3.0069E-03	4.0663E-01	1.5199E+00	1.0000E+00	4.6999E+00
13	13	3	2	8	5.7274E-04	4.8527E-02	3.3019E-01	1.0000E+00	3.1147E+00
14	14	0	0	0	0.	0.	0.	0.	0.
15	15	0	0	0	0.	0.	0.	0.	0.
16	16	0	0	0	0.	0.	0.	0.	0.
17	17	0	0	0	0.	0.	0.	0.	0.
18	18	0	0	0	0.	0.	0.	0.	0.
19	19	0	0	0	0.	0.	0.	0.	0.
20	20	0	0	0	0.	0.	0.	0.	0.
21	21	0	0	0	0.	0.	0.	0.	0.
22	22	0	0	0	0.	0.	0.	0.	0.
TOTAL		23968	20076	57862	4.1425E+00				

HAS INCREASED

COLLISIONS PER HISTORY HAS DECREASED

AVERAGE ENERGY HAS INCREASED

TOTAL NUMBER OF COLLISIONS PROCESSED ABOUT THE SAME

NPS	TALLY 1 MEAN	ERROR	FOM	TALLY 4 MEAN	ERROR	FOM	TALLY 5 MEAN	ERROR	FOM
1000	0.	0.0000	0	0.	0.0000	0	0.	0.0000	0
2000	0.	0.0000	0	0.	0.0000	0	0.	0.0000	0
3000	0.	0.0000	0	0.	0.0000	0	0.	0.0000	0
4000	0.	0.0000	0	0.	0.0000	0	0.	0.0000	0
5000	0.	0.0000	0	0.	0.0000	0	0.	0.0000	0
6000	0.	0.0000	0	0.	0.0000	0	0.	0.0000	0
7000	0.	0.0000	0	0.	0.0000	0	0.	0.0000	0
8000	0.	0.0000	0	0.	0.0000	0	0.	0.0000	0
9000	0.	0.0000	0	0.	0.0000	0	0.	0.0000	0
10000	0.	0.0000	0	0.	0.0000	0	0.	0.0000	0
11000	0.	0.0000	0	0.	0.0000	0	0.	0.0000	0
12000	0.	0.0000	0	0.	0.0000	0	0.	0.0000	0
13000	0.	0.0000	0	0.	0.0000	0	0.	0.0000	0
13968	0.	0.0000	0	0.	0.0000	0	0.	0.0000	0

****

DUMP NO. 2 ON FILE RUNTPF NPS = 13968 CTM = .60

NOTES:

ENERGY CUTOFF–0.01 MeV

1) N INCREASED FROM 3919 TO 13968
2) TRACKS STOP SOONER BECAUSE OF ENERGY CUTOFF
3) PARTICLES NOT GETTING TO TALLY REGIONS

Fig. 5. Energy cutoff of 0.01 MeV.

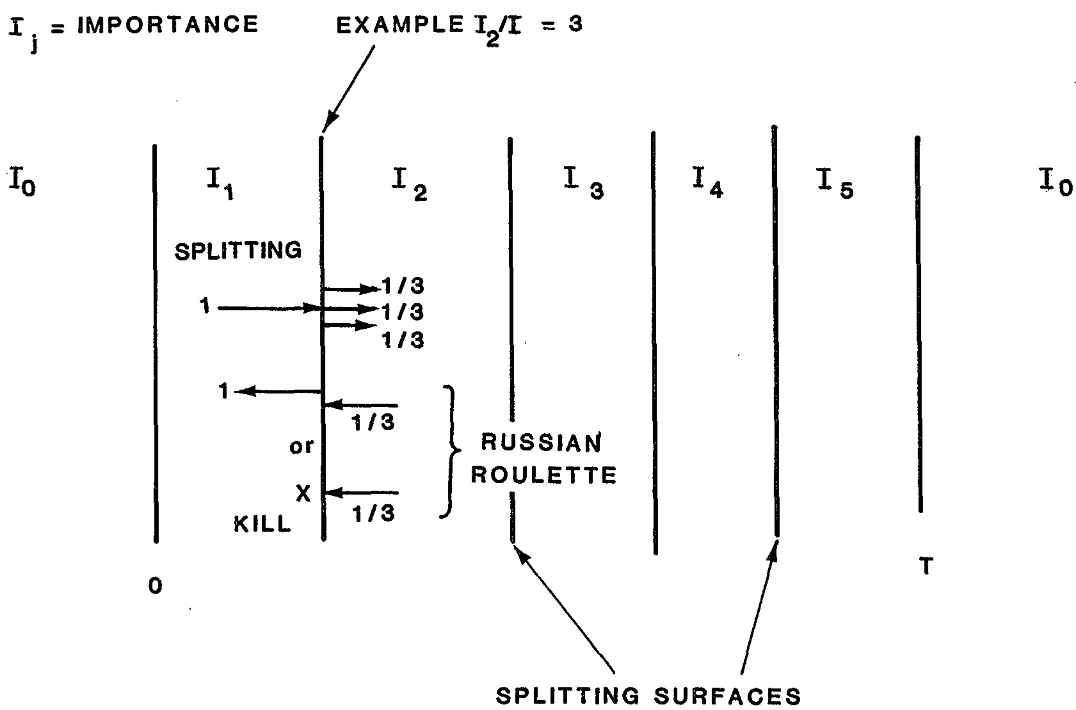


Fig. 6. Geometry splitting/Russian roulette technique.

The MCNP scheme has the advantage that all particles crossing the surface will have weight wt/v. Furthermore, if

1. geometry splitting/Russian roulette is the only nonanalog technique used and
2. all source particles start in a cell of importance I_s with weight w_s, then all particles in cell j will have weight

$$w_s \cdot \frac{I_s}{I_j}$$

regardless of the random walk taken to cell j.

MCNP's geometry splitting/Russian roulette introduces no variance in particle weight within a cell. The variation in the *number* of tracks scoring rather than a variation in particle weight determines the history variance. Empirically, it has been shown that large variations in particle weights affect tallies deleteriously. Booth[2] has shown theoretically that expected value splitting is superior to sampled splitting in high-variance situations.

C. Comments on Geometry Splitting/Russian Roulette

One other small facet deserves mention. MCNP never splits into a void although Russian roulette may be played entering a void. Splitting into a void accomplishes nothing except extra tracking because all the split particles must be tracked across the void and they all make it to the next surface. The split should be done according to the importance ratio of the last nonvoid cell departed and the first nonvoid cell entered (integer splitting into a void wastes time, but it does not increase the history variance). In contrast, noninteger splitting into a void may increase the history variance and waste time.

Finally, splitting generally decreases the history variance but increases the time per history.

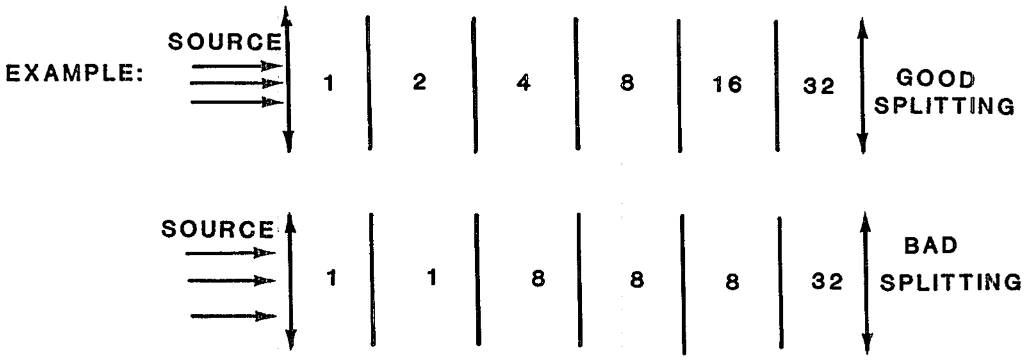
Note three more items:

1. Geometry splitting/Russian roulette works well only in problems without extreme angular dependence. In the extreme case, splitting/Russian roulette can be useless if no particles ever enter an important cell where the particles can be split.
2. Geometry splitting/Russian roulette will preserve weight variations. The technique is "dumb" in the sense that it never looks at the particle weight before deciding appropriate action. An example is geometry splitting/Russian roulette used with source biasing.
3. Geometry splitting/Russian roulette are turned on or off *together*.

D. Cautions

Although splitting/Russian roulette is among the oldest, easiest to use, and most effective techniques in MCNP, it can be abused. Two common abuses are:

1. compensating for previous poor sampling by a very large importance ratio and doing the splitting "all at once."



2. using splitting/Russian roulette with other techniques (for example, exponential transform) without forethought to possible interference effects.

E. The Sample Problem with Geometry Splitting/Russian Roulette

Returning to the problem, recall

	Cell		Tracks
	Progr	Probl	Entering
Source Cell	2	2	15416
	3	3	4445
	4	4	2197
	5	5	973
	6	6	467
	7	7	233
	8	8	110
	9	9	56
	10	10	40
	11	11	20
	12	12	8
	13	13	3
	14	14	0
	15	15	0
	16	16	0
	17	17	0
	18	18	0
	19	19	0
	20	20	0
	21	21	0
	22	22	0

Note that except for the source cell, the tracks entering are decreasing by about a factor of 2 in each subsequent cell. Furthermore, because half the particles from cell 2 (the source cell) immediately exit the geometry from the isotropic source, the rough factor of 2 even holds for the source cell. Thus as a first rough guess, try importance ratios of 2:1 through the concrete; that is, factor of 2 splitting.

Figure 7 indicates that this splitting is much better than no splitting. Not only did particles finally penetrate the concrete (see Tally 1) but the "tracks entering" column is roughly constant within a factor of 2. Slightly more splitting in cells 9-19 might improve the "tracks entering" just a little bit more. The splitting ratios were refined to be 2 in cells 2-8 and 2.15 in cells 9-19 in the next calculation.

Figure 8 summarizes the refined splitting. Immediately evident is that the FOM (Tally 1) unexpectedly decreased from 27 (Fig. 7) to 23, so at first glance, the refined splitting appears worse. However, note that the refined splitting had the desired effect; the "tracks entering" numbers are flatter. Thus I think the refined splitting is better despite the lower FOM.

What justifies being so cavalier about FOMs? Remember that the FOM is only an estimate of the calculational efficiency. At relative-error estimates near 25%, these FOMs are not meaningful enough to take the 27-to-23 FOM difference seriously. Furthermore, the FOM is only one of the *many* available pieces of summary information. At 25% error levels, it is much more important that the refined splitting appears to be sampling the geometry better.

F. Discussion of Results

The effect of refined splitting in this sample problem illustrates an important point about most variance reduction techniques; most of the improvement can usually be gained on the first try. Either one of these splitting/Russian roulette runs is several orders of magnitude better than the run without splitting. In fact, this

NOTE: MUCH BETTER

CELL PROGR	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	2329	2229	4011	1.8938E+00	2.2793E+00	6.0872E+00	1.0000E+00	7.0075E+00
3	3	1279	1119	4424	1.0444E+00	1.1595E+00	4.2228E+00	1.0000E+00	6.0880E+00
4	4	1323	1140	4803	5.6693E-01	9.6248E-01	3.5562E+00	1.0000E+00	5.8928E+00
5	5	1321	1131	4719	2.7851E-01	8.8914E-01	3.1819E+00	1.0000E+00	5.7399E+00
6	6	1326	1156	4687	1.3831E-01	8.3666E-01	2.9173E+00	1.0000E+00	5.6134E+00
7	7	1353	1154	5038	7.4333E-02	7.1902E-01	2.5852E+00	1.0000E+00	5.3490E+00
8	8	1358	1177	4881	3.6008E-02	7.5760E-01	2.6075E+00	1.0000E+00	5.4513E+00
9	9	1261	1081	4777	1.7621E-02	6.7145E-01	2.2906E+00	1.0000E+00	5.2870E+00
10	10	1182	1013	4632	8.5428E-03	6.2347E-01	2.1041E+00	1.0000E+00	5.2385E+00
11	11	1089	931	4118	3.7974E-03	5.7569E-01	1.9068E+00	1.0000E+00	5.1391E+00
12	12	998	853	3755	1.7313E-03	5.5764E-01	1.7904E+00	1.0000E+00	5.0335E+00
13	13	823	697	3127	7.2089E-04	5.8678E-01	1.8015E+00	1.0000E+00	5.1663E+00
14	14	792	678	3220	3.7117E-04	5.5526E-01	1.7102E+00	1.0000E+00	5.0461E+00
15	15	734	623	2798	1.6126E-04	5.5983E-01	1.7822E+00	1.0000E+00	5.1226E+00
16	16	664	568	2489	7.1726E-05	5.6580E-01	1.8696E+00	1.0000E+00	5.1708E+00
17	17	525	453	1807	2.6036E-05	6.1889E-01	1.9954E+00	1.0000E+00	5.2508E+00
18	18	514	441	1826	1.3155E-05	5.7871E-01	1.8610E+00	1.0000E+00	5.1414E+00
19	19	406	375	1323	4.7657E-06	7.4292E-01	2.1918E+00	1.0000E+00	5.5246E+00
20	20	163	163	0	0.	1.5219E+00	3.3568E+00	1.0000E+00	1.0000+123
21	21	0	0	0	0.	0.	0.	0.	0.
22	22	0	0	0	0.	0.	0.	0.	0.
TOTAL		19440	16982	66435	4.0653E+00				

LOW BECAUSE NEXT TO A VOID

VOID CELL

F4 TALLY IN THIS TOP CELL IS ZERO BECAUSE NO PARTICLES EVER ENTERED THE CELL

NEXT RUN : INCREASE SPLITTING AT CELL 9 TO 2.15 UNTIL CELL 19

		TALLY 1			TALLY 4			TALLY 5		
	NPS	MEAN	ERROR	FOM	MEAN	ERROR	FOM	MEAN	ERROR	FOM
	1000	8.08716E-07	.3220	31	0.	0.0000	0	0.	0.0000	0
WE GOT A TALLY!	2000	5.95093E-07	.2532	27	0.	0.0000	0	0.	0.0000	0
	2118	5.87154E-07	.2445	27	0.	0.0000	0	0.	0.0000	0

```
*************************************************************************
DUMP NO.  2 ON FILE RUNTPG           NPS =    2118          CTM =     .60
```

PARTICLES STARTED HAVE DECREASED FROM 13968

Fig. 7. Factor of 2 splitting from cells 2 to 19.

FACTOR OF 2 SPLITTING CELLS 2 - 8 FACTOR OF 2.15 SPLITTING 9 -19

FACTOR OF 2

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	1673	1598	2818	1.8539E+00	2.3501E+00	6.2140E+00	1.0000E+00	7.0985E+00
3	3	936	816	3320	1.0921E+00	1.1862E+00	4.1852E+00	1.0000E+00	6.1472E+00
4	4	1017	867	3605	5.9293E-01	9.2757E-01	3.4306E+00	1.0000E+00	5.8783E+00
5	5	1024	869	3705	3.0469E-01	8.2254E-01	2.9743E+00	1.0000E+00	5.6386E+00
6	6	1000	882	3818	1.5699E-01	7.7874E-01	2.7444E+00	1.0000E+00	5.4952E+00
7	7	971	845	3588	7.3766E-02	6.6696E-01	2.4201E+00	1.0000E+00	5.3016E+00
8	8	910	797	3115	3.2021E-02	6.7071E-01	2.3358E+00	1.0000E+00	5.3777E+00
9	9	959	838	3462	1.6553E-02	6.2399E-01	2.2691E+00	1.0000E+00	5.3498E+00
10	10	1021	881	3615	8.0391E-03	7.2649E-01	2.2829E+00	1.0000E+00	5.4503E+00
11	11	1110	954	3948	4.0836E-03	6.8330E-01	2.1960E+00	1.0000E+00	5.3577E+00
12	12	1182	1034	4499	2.1644E-03	6.5035E-01	2.0670E+00	1.0000E+00	5.2891E+00
13	13	1147	1006	4278	9.5725E-04	6.0991E-01	2.0765E+00	1.0000E+00	5.2858E+00
14	14	1066	957	3955	4.1162E-04	6.2387E-01	2.0088E+00	1.0000E+00	5.1735E+00
15	15	1026	914	3870	1.8733E-04	6.1733E-01	2.0714E+00	1.0000E+00	5.2052E+00
16	16	1011	893	3704	8.3395E-05	6.1399E-01	1.9799E+00	1.0000E+00	5.1022E+00
17	17	886	772	3409	3.5699E-05	5.5395E-01	1.8365E+00	1.0000E+00	4.9429E+00
18	18	801	710	3099	1.5094E-05	4.8961E-01	1.6415E+00	1.0000E+00	4.8516E+00
19	19	666	617	2448	5.5458E-06	5.4063E-01	1.7694E+00	1.0000E+00	5.0166E+00
20	20	222	222	0	0.	1.0130E+00	2.5229E+00	1.0000E+00	1.0000+123
21	21	1	1	0	0.	9.5608E-02	9.5608E-02	1.0000E+00	2.9257E+02
22	22	0	0	0	0.	0.	0.	0.	0.
TOTAL		18629	16473	64256	4.1390E+00				

FACTOR 2.15

SLIGHTLY FLATTER

TALLY FLUCTUATION CHARTS

	TALLY 1			TALLY 4			TALLY 5		
NPS	MEAN	ERROR	FOM	MEAN	ERROR	FOM	MEAN	ERROR	FOM
1000	5.37182E-07	.3002	26	0.	0.0000	0	0.	0.0000	0
1520	5.02929E-07	.2693	23	7.21401E-14	.9997	1	0.	0.0000	0

```
**************************************************************************************
DUMP NO.  2 ON FILE RUNTPE          NPS =     1520          CTM =     .58
```

1 HISTORY SCORED

THESE FOM NOT TERRIBLY MEANINGFUL
AT THESE ERROR LEVELS.
CONCLUSION: LIKE THESE SPLITTING FACTORS
BETTER BECAUSE OF TRACKS ENTERING
COLUMN, DESPITE "LOWER" FOM

DECREASED FROM 2118
BECAUSE OF HIGHER SPLITTING

Fig. 8. Refined splitting.

problem is so bad without splitting that it is hard to guess how much splitting/Russian roulette has improved the efficiency. Contrast this improvement to the (questionable) FOM difference of 27 (Fig. 7) to 23 (Fig. 8) between the factor of 2 splitting and the refined splitting. Usually one can do better with a variance reduction technique on the second try than on the first, but usually by not more than a factor of 2.

Quickly reaching diminishing returns is characteristic of a competent user and a good variance reduction technique. Competent users can quickly learn good importances because there is a very broad near-optimal range. Because the optimum is broad, the statistics often mask which importance set is best when they are all in the vicinity of the optimum.

Now that a reasonably flat track distribution has been obtained, perhaps it is time to explain why one expects this to be near optimal. There are some plausible arguments, but the real reason is empirical; it has been observed in many similar problems (that is, essentially one-dimensional bulk penetration problems) that a flat track distribution is near optimal. The radius of the concrete cylinder is large enough (100 cm) that the cylinder appears much like a slab; very few particles cross its cylindrical surface at a given depth (y-coordinate) compared to the particle population at that depth. Indeed, if the radius were infinite, the cylinder would be a slab and no particles would cross its cylindrical surface.

A plausible argument for flat track distribution can be made by considering an extremely thick slab and possible track distributions for two cases. For too little splitting, the track population will decrease roughly exponentially with increasing depth and no particles will ever penetrate the slab. For too much splitting, the importance ratios are too large; the track population will increase roughly exponentially and a particle history will never terminate. In both cases, albeit for different reasons, there are never any tallies. If neither an exponentially decreasing population nor an exponentially increasing population is advisable, the only choice is a flat distribution.

Of course, there are really many more choices than exponentially decreasing, flat, or exponentially increasing populations, but track populations usually behave in one of these ways because the importance ratios from one cell to the next are normally chosen (at least for a first guess) equal. The reason is that one cell in the interior is essentially equivalent to the next cell, so there is little basis to choose a different importance ratio from one cell to the next. However, the cells are not quite equivalent because they are different depths from the source, so the average energy (and mean free path) decreases with increasing depth. This is probably why it was necessary to increase the importance ratio from 2 to 2.15 in the deep parts of the sample problem. Note, however, that this is a small correction.

Returning to Fig. 8, note that the energy and mean free path decrease with increasing depth, as expected. Not also that the higher splitting has decreased the particles per minute.

VII. ENERGY SPLITTING/ROULETTE

Energy splitting/Russian roulette is very similar to geometry splitting/Russian roulette except energy splitting/roulette is done in the energy domain rather than in the spatial domain. Note two differences.

1. Unlike geometry splitting/roulette, the energy splitting/roulette uses actual splitting ratios as supplied in the input file rather than obtaining the ratios from importances.
2. It is possible to play energy splitting/roulette only on energy decreases if desired.

There are two cautions.

1. The weight cutoff game takes no account of what has occurred with energy splitting/roulette.
2. Energy splitting/roulette is played throughout the entire problem. Consider using a space-energy weight window if there is a substantial space variation in what energies are important.

One can expect an improvement in speed using energy roulette by recalling that the problem ran a factor of 4 faster with an energy cutoff of 0.01 MeV than without an energy cutoff. Low-energy particles get progressively less important as their energy drops, so it might help to play Russian roulette at several different energies as the energy drops. In the following run, a 50% survival game was played at 5 MeV, 1 MeV, 0.3 MeV, 0.1 MeV, and 0.03 MeV. The energies and the 50% survival probability were only guesses.

The energy roulette (splitting does not happen here because there is no upscatter) results are shown in Fig. 9. Note that there were substantially (~50%) more tracks entering, approximately the same number of collisions, and three times as many particles run. The FOM looks better, but the mean (Tally 1) has increased from 5.0E-7 (Fig. 8) to 8.4E-7. This deserves note and caution, but not panic, because the error is 18%, so poor estimates in both tally and error can be expected. Despite the previous statement, the energy roulette looks successful in improving tallies 1 and 4.

VIII. IMPLICIT CAPTURE AND WEIGHT CUTOFF

A. Implicit Capture

Implicit capture, survival biasing, and absorption by weight reduction are synonymous. Implicit capture is a

AT 5 1 0.3 0.1 0.03MeV

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	4925	4840	5669	1.9902E+00	2.3276E+00	6.1205E+00	1.2907E+00	7.0018E+00
3	3	2071	1921	4678	1.1643E+00	1.0958E+00	3.9717E+00	1.6928E+00	5.9897E+00
4	4	1837	1698	4111	5.8076E-01	8.0935E-01	3.3359E+00	1.9502E+00	5.6657E+00
5	5	1567	1438	3601	2.5505E-01	9.3732E-01	3.1243E+00	1.9957E+00	5.6926E+00
6	6	1362	1270	3176	1.3056E-01	7.8732E-01	2.8812E+00	2.1442E+00	5.5827E+00
7	7	1260	1166	2822	5.3775E-02	8.7579E-01	2.8324E+00	2.1363E+00	5.7139E+00
8	8	1179	1099	2638	2.5744E-02	7.9706E-01	2.6317E+00	2.2040E+00	5.6698E+00
9	9	1208	1134	2558	1.2297E-02	8.3991E-01	2.7341E+00	2.1718E+00	5.7046E+00
10	10	1290	1183	2941	7.3733E-03	6.7628E-01	2.3517E+00	2.4449E+00	5.3481E+00
11	11	1391	1276	3403	4.2709E-03	6.2320E-01	2.1638E+00	2.6094E+00	5.1918E+00
12	12	1327	1225	2960	1.6659E-03	6.6384E-01	2.1940E+00	2.5545E+00	5.3025E+00
13	13	1309	1200	3201	8.0190E-04	7.1465E-01	2.2723E+00	2.4564E+00	5.4214E+00
14	14	1295	1192	2862	3.0306E-04	9.3180E-01	2.4979E+00	2.2807E+00	5.7146E+00
15	15	1321	1218	3155	1.7364E-04	6.2616E-01	2.0338E+00	2.6129E+00	5.2835E+00
16	16	1372	1265	3097	8.2605E-05	5.4080E-01	1.9274E+00	2.7255E+00	5.1923E+00
17	17	1353	1268	2968	3.6516E-05	5.8219E-01	2.0133E+00	2.6598E+00	5.2528E+00
18	18	1361	1282	3143	1.7358E-05	5.5134E-01	1.9932E+00	2.6756E+00	5.2760E+00
19	19	1283	1240	2708	6.5219E-06	6.4948E-01	2.0873E+00	2.5420E+00	5.5096E+00
20	20	570	570	0	0.	1.0062E+00	2.6046E+00	4.3766E+00	1.0000+123
21	21	6	6	0	0.	7.2412E-01	1.1786E+00	3.0008E+00	4.2805E+02
22	22	0	0	0	0.	0.	0.	0.	0.
TOTAL		29287	27491	59691	4.2274E+00				

SUBSTANTIALLY MORE TRACKS

SLIGHTLY FEWER COLLISIONS

NPS	TALLY 1 MEAN	ERROR	FOM	TALLY 4 MEAN	ERROR	FOM	TALLY 5 MEAN	ERROR	FOM
1000	7.09853E-07	.3305	66	0.	0.0000	0	0.	0.0000	0
2000	8.88887E-07	.2890	42	2.00483E-13	.9997	3	0.	0.0000	0
3000	9.61129E-07	.2133	52	3.00442E-13	.6381	5	0.	0.0000	0
4000	9.46209E-07	.1857	53	2.25331E-13	.6381	4	0.	0.0000	0
4699	8.37540E-07	.1796	50	1.91812E-13	.6382	4	0.	0.0000	0

```
*****************************************************************************
DUMP NO.  2 ON FILE RUNTPF          NPS =    4699          CTM =    .61
```

INCREASED FROM 1520 WITH THE INTRODUCTION OF ENERGY ROULETTE

FOM LOOKS BETTER, BUT INCREASE IN MEAN FROM 5.0 TO 8.4 DESERVES NOTE AND CAUTION. ERROR IS 18%, SO EXPECT POOR ESTIMATES BOTH IN TALLY AND IN ERROR.

Fig. 9. Using energy roulette
(50% survival at 5 1 0.3 0.1 0.03 MeV)

variance reduction technique applied in MCNP *after* the collision nuclide has been selected. Let

σ_{ti} = total microscopic cross section for nuclide i and
σ_{ai} = microscopic absorption cross section for nuclide i.

When implicit capture is used rather than sampling for absorption with probability σ_{ai}/σ_{ti}, the particle always survives the collision and is followed with new weight

$$wt \cdot \left(1 - \frac{\sigma_{ai}}{\sigma_{ti}}\right).$$

Two advantages of implicit capture are
1. a particle that has finally, against considerable odds, reached the tally region is not absorbed just before a tally is made, and
2. the history variance, in general, decreases when the surviving weight (that is, 0 or wt) is not sampled, but an expected surviving weight is used instead (but see weight cutoff discussion, Sec. VIII.B).

Two disadvantages are
1. implicit capture introduces fluctuation in particle weight and
2. increases the time per history (but see weight cutoff discussion, Sec. VIII.B).

Note that
1. Implicit capture is the default in MCNP (except for note 4).
2. Implicit capture is always turned on for neutrons unless the weight cutoff game is turned off.
3. Explicit (analog) capture is not allowed for the photon simple physics treatment (high energy).
4. Analog capture is allowed only in detailed photon physics.

B. Weight Cutoff

In weight cutoff, Russian roulette is played if a particle's weight drops below a user-specified weight cutoff. The particle is either killed or its weight is increased to a user-specified level. The weight cutoff was originally envisioned for use with geometry splitting/Russian roulette and implicit capture. Because of this,
1. the weight cutoffs in cell j depend not only on WC1 and WC2 (see Fig. 2) on the CUTN and CUTP cards, but also on the cell importances. This dependence is intended to adjust the weight cutoff values to make sense with geometry splitting/Russian roulette.
2. Implicit capture is always turned on (except in detailed photon physics) whenever a nonzero WC1 is specified.

The weight cutoffs WC1 and WC2 are illustrated in Fig. 10. If a particle's weight falls below $R_j \cdot WC2$, a weight cutoff game is played; with probability $wt/(WC1 \cdot R_j)$ the particle survives with new weight $WC1 \cdot R_j$; otherwise the particle is killed.

As mentioned earlier, the weight cutoff game was originally envisioned for use with geometry splitting and implicit capture. Consider what can happen without a weight cutoff. Suppose a particle is in the interior of a very large medium and there are no time nor energy cutoffs. The particle will go from collision to collision, losing a fraction of its weight at each collision. Without a weight cutoff, the particle's weight would eventually be too small to be representable in the computer, at which time an error would occur. If there are other loss mechanisms (for example, escape, time cutoff, or energy cutoff), the particle's weight will not decrease indefinitely, but the particle may take an unduly long time to terminate.

Weight cutoff's dependence on the importance ratio can be easily understood if one remembers that the weight cutoff game was originally designed to solve the low-weight problem sometimes produced by implicit capture. In a high-importance region, the weights are low *by design*, so it makes no sense to play the same weight cutoff game in high- and low-importance regions. In fact, as mentioned in a previous section, if splitting is the only nonanalog technique used, all particles in a given cell have the same weight, so no weight cutoff game would make sense. That is, if the particle weight is too small in a cell, the cell importance simply needs to be decreased. The weight cutoff is meant to indicate when a particle's weight is too low to be worth transporting.

In addition to the weight cutoff's dependence on cell importance, the weight cutoffs are automatically made relative to the minimum source weight if the source is a standard MCNP source and the weight cutoffs (WC1, WC2) are prefixed by a negative sign.

1. Cautions

a. Many techniques in MCNP cause weight change; the weight cutoff was really designed with geometry splitting and implicit capture in mind. Care should be taken in the use of other techniques.

b. In most cases, if you specify a weight cutoff, you automatically get implicit capture.

2. Notes

a. Weight cutoff games are unlike time and energy cutoffs. In time and energy cutoffs, the random walk is *always* terminated when the threshold is crossed. Potential bias may result if the particle's importance was not zero. A weight cutoff (weight roulette would be a better name) does *not* bias the game because the weight is increased for those particles that survive.

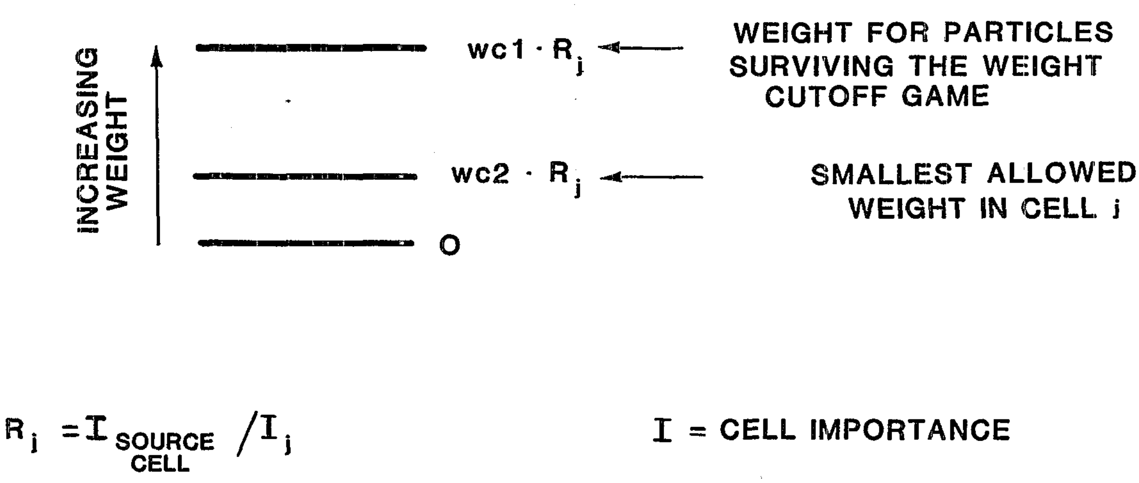


Fig. 10. Weight cutoff mechanics.

b. By default, the weight cutoff game is turned off in a weight window cell.

C. Weight Cutoff and Implicit Capture Applied to the Sample Problem

Figure 11 shows the result of adding weight cutoff and implicit capture techniques in addition to the
1. energy cutoff,
2. refined geometry splitting/Russian roulette, and
3. energy roulette techniques.

Comparing Fig. 11 to Fig. 9, one can see that implicit capture and weight cutoff did apparently reduce the tally 1 error for the same number of particles. However, the number of particles run was down by a factor of 2, resulting in a net decrease in the FOM. In general, if a nonanalog technique does not show a clear improvement, do not use it; thus for the next run, the implicit capture and weight cutoff will be turned off.

Tally 1 seems reasonably well optimized by
1. geometry splitting and roulette,
2. energy cutoff,
3. energy roulette (and splitting), and
4. analog capture.

Tally 4 is bad because very few tracks exit the concrete cylinder (cell 19) in the small solid angle subtended by cell 21. Tally 5 is even worse, in fact nonexistent, because of the few particles that do reach cell 21, none collide, so there are no point detector contributions.

Consider improving the worst tally (tally 5) first. Note from the summary charts that the free path in cell 21 is ~1000 cm and the cell is 10 cm thick. Only a tiny fraction of the particles entering cell 21 will collide in an analog fashion. The forced collision technique in MCNP solves this problem by *requiring* each track entering a cell to collide.

IX. FORCED COLLISIONS

Forced collision is normally used to sample collisions in optically thin (fractional mean free path) cells where not enough collisions are being sampled. A track entering a forced collision cell is split into two tracks: uncollided and collided. That is, MCNP calculates the expected weight traversing the cell and assigns that weight to the uncollided track, and MCNP calculates the expected weight colliding in the cell and assigns that weight to the collided track (Fig. 12). The uncollided track is put on the cell boundary (the point intersected by the cell boundary and the track direction), and the collided track's collision site is sampled in the usual way except that the collision site must now be sampled from a conditional probability, the condition being that a collision occurs at a distance $0 < x < \ell$.

A. Comments

1. Although the forced collision technique is normally used to obtain collisions in optically thin cells, it can also be used in optically thick cells to get the uncollided transmission.
2. The weight cutoff game is normally turned off in a forced collision cell (see MCNP Manual for exceptional cases[1]).
3. The forced collision technique decreases the history variance, but the time per history increases.
4. More than one collision can be forced in a cell.
5. ℓ of Fig. 12 is always the distance from the point at which the track is split into its collided and uncollided parts to the boundary. In Fig. 12, the split is done upon entrance to the cell, but the split can occur at an interior point as well (splits at interior

CELL PROGR	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	2237	2189	2898	2.1476E+00	1.8374E+00	5.6742E+00	1.2206E+00	6.7028E+00
3	3	1186	1097	2897	1.1189E+00	1.2427E+00	4.0314E+00	1.3540E+00	5.8642E+00
4	4	1325	1208	3166	6.4234E-01	8.5774E-01	3.1891E+00	1.4630E+00	5.5270E+00
5	5	1491	1358	3368	3.0964E-01	9.1416E-01	3.1710E+00	1.3746E+00	5.7341E+00
6	6	1415	1310	3201	1.3708E-01	7.9773E-01	2.9849E+00	1.3433E+00	5.6437E+00
7	7	1398	1295	3302	7.8890E-02	5.9432E-01	2.5009E+00	1.4377E+00	5.2856E+00
8	8	1418	1313	3188	3.1016E-02	7.7036E-01	2.7295E+00	1.2648E+00	5.5940E+00
9	9	1445	1320	3317	1.6468E-02	6.4889E-01	2.4403E+00	1.2894E+00	5.3027E+00
10	10	1464	1356	3522	7.0522E-03	8.0370E-01	2.5044E+00	1.1811E+00	5.4445E+00
11	11	1405	1305	3208	2.9794E-03	6.6734E-01	2.4093E+00	1.1671E+00	5.3589E+00
12	12	1450	1344	3129	1.4564E-03	5.5143E-01	2.1809E+00	1.2476E+00	5.1541E+00
13	13	1492	1366	3362	6.9754E-04	7.1905E-01	2.2514E+00	1.1605E+00	5.3834E+00
14	14	1565	1453	3683	3.3628E-04	6.4563E-01	2.1268E+00	1.1339E+00	5.1845E+00
15	15	1577	1465	3625	1.5533E-04	5.9601E-01	2.0087E+00	1.1594E+00	5.1098E+00
16	16	1611	1484	3696	7.1971E-05	5.9638E-01	2.0357E+00	1.1115E+00	5.1709E+00
17	17	1632	1514	3831	3.2251E-05	6.1970E-01	2.0445E+00	1.0851E+00	5.3119E+00
18	18	1601	1489	3358	1.1371E-05	7.3876E-01	2.2109E+00	9.9903E-01	5.5864E+00
19	19	1484	1438	3236	5.0066E-06	7.1488E-01	2.3301E+00	9.6091E-01	5.6517E+00
20	20	652	652	0	0.	1.7044E+00	3.4031E+00	1.4500E+00	1.0000+123
21	21	4	4	0	0.	1.7145E+00	1.7156E+00	7.7972E-01	5.8049E+02
22	22	0	0	0	0.	0.	0.	0.	0.
TOTAL		27852	25960	59987	4.4947E+00				

NUMBER OF COLLISIONS PROCESSED THE SAME, BUT ABOUT HALF THE NUMBER OF PARTICLES RUN

APPARENTLY THE WEIGHT CUTOFF AND IMPLICIT CAPTURE DID REDUCE THE HISTORY VARIANCE, BUT LOST SINCE THE TIME PER HISTORY INCREASED TOO MUCH

REDUCED RELATIVE ERROR FOR SAME PARTICLES

TALLY FLUCTUATION CHARTS

	TALLY 1			TALLY 4			TALLY 5		
NPS	MEAN	ERROR	FOM	MEAN	ERROR	FOM	MEAN	ERROR	FOM
1000	4.45147E-07	.2634	49	7.21958E-14	.9995	3	0.	0.0000	0
2000	5.47418E-07	.2031	42	5.86343E-14	.7254	3	0.	0.0000	0
2099	5.61666E-07	.1933	37	5.58688E-14	.7254	2	0.	0.0000	0

****

DUMP NO. 2 ON FILE RUNTPG NPS = 2099 CTM = .61

APPEARS WORSE THAN FOM = 50 LAST TIME; STATISTICS STILL NOT VERY RELIABLE

NEXT TIME REMOVE WEIGHT CUTOFF AND IMPLICIT CAPTURE

Fig. 11. Weight cutoff (.5 .25) and implicit capture.

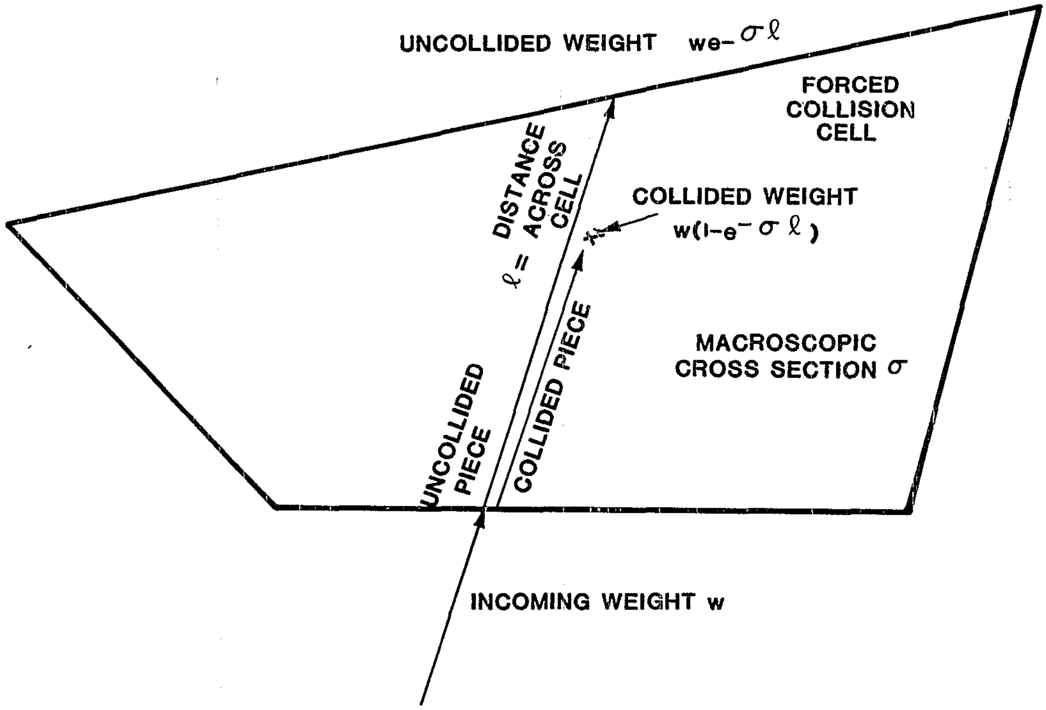


Fig. 12. Forced collision procedure.

points normally occur when more than one collision per entering track is forced).

B. Caution

Because weight cutoffs are turned off in forced collision cells, the number of tracks can get exceedingly large if there are several adjacent forced collision cells.

C. Forced Collisions Applied to the Sample Problem

Recall that the point detector tally (tally 5) was nonexistent because there were no collisions in cell 21. Figure 13 shows the effects of forcing one collision in cell 21 in addition to energy cutoff, refined geometry splitting/Russian roulette, and energy roulette. Note that 44 tracks entered cell 21 and there were 44 collisions in cell 21. Also note that the point detector tally is now obtaining contributions. Thus, the forced collision has really helped the point detector tally. The trouble now is not the lack of collisions from tracks that enter cell 21, but rather the small number of particles that enter cell 21. Angle biasing in some form is required to preferentially scatter particles into cell 21.

X. DXTRAN

The DXTRAN technique and source angle biasing are currently the only angle-biasing techniques in MCNP. Unlike source angle biasing, DXTRAN biases the scattering directions as well as the source direction.

Before explaining the DXTRAN theory, I will first loosely describe what occurs. A typical problem in which DXTRAN might be employed is much like the sample problem; a small region (for example, cell 21) is being inadequately sampled because particles almost never scatter toward the small region. To ameliorate this situation, the user can specify a DXTRAN sphere (in the input file) that encloses the small region. Upon particle collision (or exiting the source) outside the sphere, the DXTRAN technique creates a special "DXTRAN particle" and deterministically scatters it toward the DXTRAN sphere and deterministically transports it, without collision, to the surface of the DXTRAN sphere (Fig. 14). The collision itself is otherwise treated normally, producing a non-DXTRAN particle that is sampled in the normal way, with no reduction in weight. However, the non-DXTRAN particle is killed if it tries to enter the DXTRAN sphere.

The subtlety about DXTRAN is how the extra weight created for the DXTRAN particles is balanced by the

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	33158	32585	38721	1.9463E+00	2.2794E+00	6.0714E+00	1.2965E+00	6.9546E+00
3	3	14914	13932	34186	1.1305E+00	1.0507E+00	3.8230E+00	1.7265E+00	5.8555E+00
4	4	13784	12817	31887	5.7432E-01	8.8826E-01	3.2690E+00	1.9341E+00	5.6357E+00
5	5	13485	12457	30177	2.7660E-01	7.7716E-01	2.9746E+00	2.0974E+00	5.5166E+00
6	6	12627	11649	28028	1.3317E-01	7.8522E-01	2.8797E+00	2.1597E+00	5.5396E+00
7	7	11729	10786	26188	6.3336E-02	7.8444E-01	2.7773E+00	2.2148E+00	5.5669E+00
8	8	10735	9904	24261	3.0388E-02	7.5598E-01	2.6051E+00	2.2929E+00	5.5113E+00
9	9	10630	9847	24056	1.4810E-02	6.7951E-01	2.4260E+00	2.3985E+00	5.4027E+00
10	10	10475	9692	24365	7.0740E-03	6.7178E-01	2.3186E+00	2.4523E+00	5.3440E+00
11	11	10265	9545	23448	3.1479E-03	6.7630E-01	2.3071E+00	2.4446E+00	5.3741E+00
12	12	10218	9461	23357	1.4920E-03	6.6839E-01	2.2525E+00	2.4837E+00	5.3770E+00
13	13	10203	9435	23575	7.0432E-04	6.6659E-01	2.2037E+00	2.4998E+00	5.3401E+00
14	14	10157	9377	23262	3.3582E-04	6.2614E-01	2.1241E+00	2.5585E+00	5.2455E+00
15	15	10134	9398	24042	1.5945E-04	6.1916E-01	2.0813E+00	2.5892E+00	5.1981E+00
16	16	10070	9313	23380	7.1203E-05	6.0021E-01	2.0213E+00	2.6266E+00	5.1843E+00
17	17	9993	9291	23245	3.2652E-05	5.9809E-01	2.0129E+00	2.6205E+00	5.1975E+00
18	18	9661	8978	22146	1.3640E-05	6.3806E-01	2.0663E+00	2.5577E+00	5.2681E+00
19	19	8727	8431	19918	5.3706E-06	6.9204E-01	2.1459E+00	2.4912E+00	5.4016E+00
20	20	3669	3665	0	0.	1.5197E+00	3.4888E+00	3.6344E+00	1.0000+123
21	21	44	88	44	2.8836E-11	5.3640E+00	6.1170E+00	5.8094E-01	8.4249E+02
22	22	7	7	0	0.	4.8993E+00	5.0533E+00	2.1420E-02	1.0000+123
TOTAL		224685	210658	468286	4.1825E+00				

ONE COLLISION PER TRACK

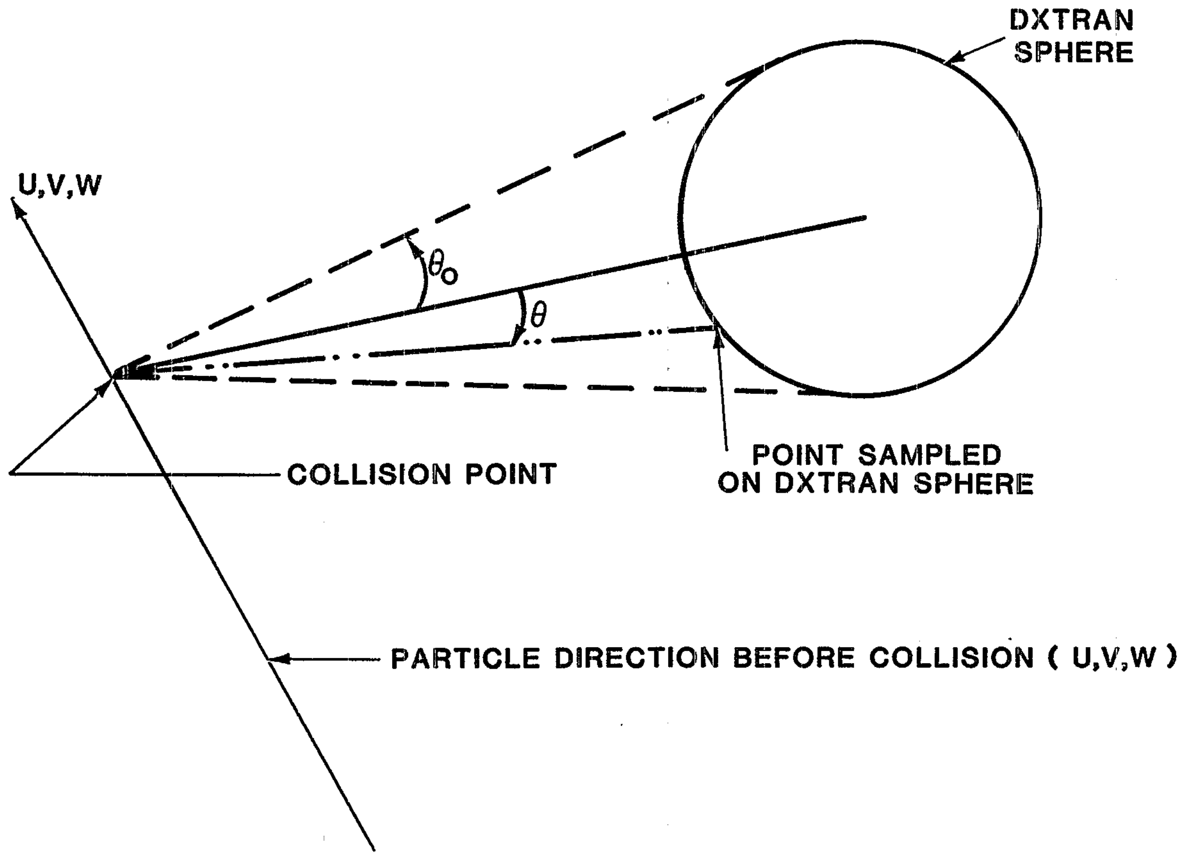
NOW GET DETECTOR TALLY FROM COLLISIONS

NPS	TALLY 1 MEAN	ERROR	FOM	TALLY 4 MEAN	ERROR	FOM	TALLY 5 MEAN	ERROR	FOM
2000	4.43762E-07	.2926	38	3.93854E-14	.9997	3	1.09306E-17	.9997	3
4000	5.00005E-07	.2048	39	1.96927E-14	.9999	1	5.46528E-18	.9999	1
6000	4.89498E-07	.1735	38	1.31285E-14	.9999	1	3.64352E-18	.9999	1
8000	5.45431E-07	.1560	35	2.95408E-14	.7453	1	2.73264E-18	.9999	0
10000	5.75778E-07	.1318	38	2.36326E-14	.7453	1	2.18611E-18	.9999	0
12000	5.76643E-07	.1146	42	2.64523E-14	.6109	1	5.74344E-18	.7528	0
14000	5.67196E-07	.1050	43	3.40704E-14	.4703	2	9.73618E-18	.5345	1
16000	5.64437E-07	.0982	43	5.44775E-14	.3741	3	1.63891E-17	.4208	2
18000	5.44298E-07	.0930	44	4.84244E-14	.3741	2	1.45681E-17	.4208	2
20000	5.34245E-07	.0895	43	4.35820E-14	.3741	2	1.31113E-17	.4208	1
22000	5.45122E-07	.0845	44	5.04040E-14	.3346	2	1.61184E-17	.3816	2
24000	5.78085E-07	.0791	45	6.92172E-14	.3261	2	2.47790E-17	.3653	2
26000	5.59956E-07	.0770	44	7.60020E-14	.2963	3	2.66047E-17	.3296	2
28000	5.51656E-07	.0739	44	7.05733E-14	.2963	2	2.47043E-17	.3296	2
30000	5.52045E-07	.0706	46	7.11231E-14	.2842	2	2.42602E-17	.3172	2
31617	5.58607E-07	.0682	45	7.52914E-14	.2664	2	2.61322E-17	.2919	2

```
**********************************************************************
DUMP NO.  2 ON FILE RUNTPF            NPS =    31617           CTM =     4.61
```

PART / MIN = 6858

Fig. 13. Weight cutoff and implicit capture removed; forced collision used in cell 21.



1. A point on the DXTRAN sphere is sampled.
2. A particle is scattered towards the selected point.
3. The particle's weight is exponentially decreased by the optical path and adjusted for bias in the scattering angle.
4. The original particle is sampled in the normal way (with no reduction in weight).
5. If the original particle tries to enter the DXTRAN sphere, it is terminated.

Fig. 14. DXTRAN concept.

weight killed as non-DXTRAN particles cross the DXTRAN sphere. The non-DXTRAN particle is followed without any weight correction, so if the DXTRAN technique is to be unbiased, the extra weight put on the DXTRAN sphere by DXTRAN particles must somehow (on average) balance the weight of non-DXTRAN particles killed on the sphere.

A. DXTRAN Viewpoint #1

One can view DXTRAN as a splitting process (much like the forced collision technique) wherein each particle is split upon departing a collision (or source point) into two distinct pieces:

1. the weight that does not enter the DXTRAN sphere on the next flight either because the particle is not pointed toward the DXTRAN sphere or because the particle collides before reaching the DXTRAN sphere, and
2. the weight that enters the DXTRAN sphere on the next flight.

Let w_o be the weight of the particle before exiting the collision, let p_1 be the analog probability that the particle does not enter the DXTRAN sphere on its next flight, and let p_2 be the analog probability that the particle does enter the DXTRAN sphere on its next flight. The particle must undergo one of these mutually exclusive events, thus $p_1 + p_2 = 1$. The expected weight not entering the DXTRAN sphere is $w_1 = w_o p_1$, and the expected weight entering the DXTRAN sphere is $w_2 = w_o p_2$. Think of DXTRAN as deterministically splitting the original particle with weight w_o into two particles, a non-DXTRAN (particle 1) particle of weight w_1 and a DXTRAN (particle 2) particle of weight w_2. Unfortunately, things are not quite that simple.

Recall that the non-DXTRAN particle is follwed with unreduced weight w_o rather than weight $w_1 = w_o p_1$. The reason for this apparent discrepancy is that the non-DXTRAN particle (#1) plays a Russian roulette game. Particle 1's weight is increased from w_1 to w_o by playing a Russian roulette game with survival probability $p_1 = w_1/w_o$. The reason for playing this Russian roulette game is simply that p_1 is not known, so assigning weight $w_1 = p_1 w_o$ to particle 1 is impossible. However, it is possible to play the Russian roulette game without explicitly knowing p_1. It is not magic, just slightly subtle.

The Russian roulette game is played by sampling particle 1 normally and keeping it only if it does not enter (on its next flight) the DXTRAN sphere; that is, particle 1 survives (by definition of p_1) with probability p_1. Similarly, the Russian roulette game is lost if particle 1 enters (on its next flight) the DXTRAN sphere; that is, particle 1 loses the roulette with probability p_2. Now I restate this idea. With probability p_1, particle 1 has weight w_o and does not enter the DXTRAN sphere and with probability p_2, the particle enters the DXTRAN sphere and is killed. Thus, the expected weight not entering the DXTRAN sphere is $w_o p_1 + 0 \cdot p_2 = w_1$, as desired.

So far, this discussion has concentrated on the non-DXTRAN particle and ignored exactly what happens to the DXTRAN particle. The sampling of the DXTRAN particle will be discussed after a second viewpoint on the non-DXTRAN particle.

B. DXTRAN Viewpoint #2

If you have understood the first viewpoint, you need not read this viewpoint. On the other hand, if the first viewpoint was not clear, perhaps this second one will be.

This second way of viewing DXTRAN does not see it as a splitting process but as an accounting process where weight is both created and destroyed on the surface of the DXTRAN sphere. In this view, DXTRAN estimates the weight that should go to the DXTRAN sphere upon collision and creates this weight on the sphere as DXTRAN particles. If the non-DXTRAN particle does not enter the sphere, its next flight will proceed exactly as it would have without DXTRAN, producing the same tally contributions and so forth. However, if the non-DXTRAN particle's next flight attempts to enter the sphere, the particle must be killed or there would be (on average) twice as much weight crossing the DXTRAN sphere as there should be, the weight crossing the sphere having already been accounted for by the DXTRAN particle.

C. The DXTRAN Particle

Although the DXTRAN particle does not confuse people nearly as much as the non-DXTRAN particle, the DXTRAN particle is nonetheless subtle.

The problem is how to sample the DXTRAN particle's location on the DXTRAN sphere. One cannot afford to calculate a cumulative distribition function to select the scattering direction θ indicated in Fig. 14. [The azimuthal angle is sampled uniformly in $(0,2\pi)$]. This would essentially involve integrating the scattering probability density at each collision. Instead of sampling the true probability density, one samples an arbitrary density and adjusts the weight appropriately.

As indicated above, a point on the DXTRAN sphere can be selected from any density function because the weight of the DXTRAN particle is modified by

$$\frac{\text{true density to select point } p_s}{\text{density sampled to select } p_s}$$

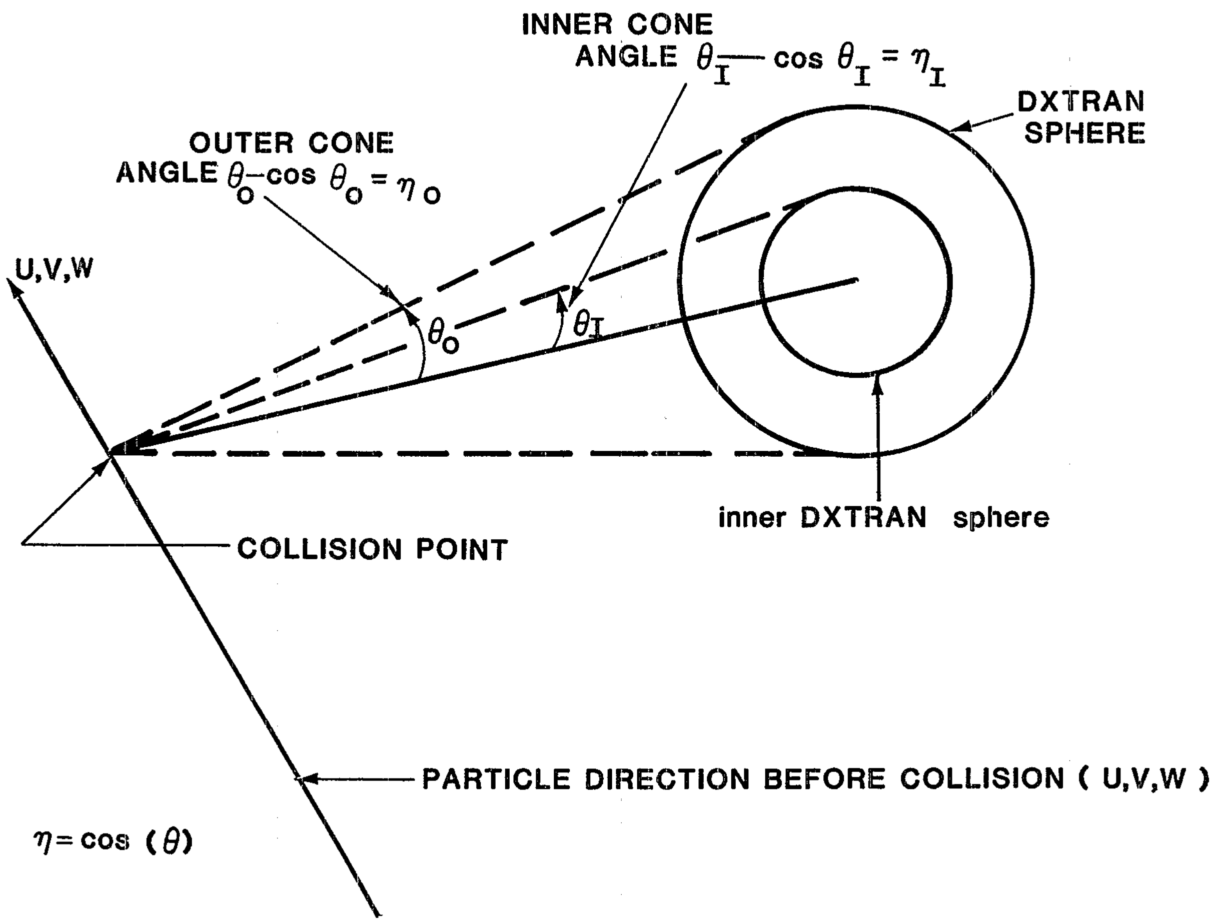


Fig. 15. Sampling the DXTRAN particle.

This is easy to do because the true scattering density function is immediately available even if its integral is not. MCNP arbitrarily uses the two-step density described below. In fact, the inner DXTRAN sphere has *only* to do with this arbitrary density and is not essential to the DXTRAN concept.

MCNP samples the inner cone uniformly in $(\eta_I, 1)$, and the outer cone uniformly in (η_o, η_I) (Fig. 15). However, the inner cone is sampled with five times the probability density that the outer is sampled. That is to say the inner cone is taken to be five times as important as the outer cone. Further mathematical details are given in the MCNP manual[1] and will not be discussed here.

After the scattering angle has been chosen, the DXTRAN particle is deterministically transported to the DXTRAN sphere without collision and with weight attenuated by the exponential of the optical path.

D. Inside the DXTRAN Sphere

So far, only collisions outside the DXTRAN sphere have been discussed. At collisions inside the DXTRAN sphere, the DXTRAN game is not played* because first, the particle is already in the desired region and second, it is impossible to define the angular cone of Fig. 14.

E. Terminology - Real Particle, Pseudoparticle

In X-6 documentation, at least through the April 1981 MCNP Manual,[1] the DXTRAN particle is called a

*If there are several DXTRAN spheres and the collision occurs in sphere i, then DXTRAN will be played for all spheres except sphere i.

pseudoparticle and the non-DXTRAN particle is called the original or real particle. The terms "real particle" and "pseudoparticle" are potentially misleading. Both particles are equally real; both execute random walks, both carry nonzero weight, and both contribute to tallies. The only stage at which the DXTRAN particle should be considered "psuedo" or "not real" is during creation. A DXTRAN particle is created on the DXTRAN sphere, but creation involves determining what weight the DXTRAN particle should have upon creation. Part of this weight determination requires calculating the optical path between the collision site and the DXTRAN sphere. MCNP determines the optical path by tracking a pseudoparticle from the collision site to the DXTRAN sphere. This pseudoparticle is deterministically tracked to the DXTRAN sphere simply to determine the optical path; no distance to collision is sampled, no tallies are made, and no records of the pseudoparticle's passage are kept (for example, tracks entering). In contrast, once the DXTRAN particle is created at the sphere's surface, the particle is no longer a pseudoparticle; the particle has real weight, executes random walks, and contributes to tallies.

F. Comments

1. DXTRAN spheres have their own weight cutoffs.
2. The DD card (by default) stops extremely low-weighted tracks by roulette. See the manual[1] for how this is accomplished.
3. Strongly consider producing DXTRAN particles only on some fraction of the number of collisions, as allowed by the DXCPN card.

G. CAVEATS

1. DXTRAN should be used carefully in optically thick problems. Do not rely on DXTRAN to do penetration.
2. If the source is user-supplied, some provision (SRCDX, page 263 of the MCNP manual[1]) must be made for obtaining the source contribution to particles on the DXTRAN sphere.
3. Extreme care must be taken when more than one DXTRAN sphere is in a problem. Cross-talk between spheres can result in extremely low weights and an explosion in particle tracks.
4. A different set of weight cutoffs is used inside the DXTRAN sphere.

H. DXTRAN Applied to the Sample Problem

Recall that there was a problem getting enough particles to scatter in the direction of cell 21. To solve this problem, a DXTRAN sphere was specified just large enough to surround cell 21 (Fig. 16). If a larger DXTRAN sphere were used, some DXTRAN particles would miss cell 21 and this would be less efficient. If a smaller DXTRAN sphere were used, it would be possible for a non-DXTRAN particle to enter cell 21, resulting in an undesirable large weight fluctuation in cell 21. Note also that the inner and outer DXTRAN spheres are coincident. This choice was made because specifying different spheres would introduce a five-to-one weight variation even though all particles entering cell 21 are about equally important.

I. Discussion

Note from Fig. 17 that DXTRAN did have the desired effect; the tracks entering cell 21 have increased dramatically and the FOMs for tallies 4 and 5 have increased by a factor of 7. However, note that the particles-per-minute number has decreased by a factor of 4; this is reflected in a factor of 4 decrease in tally 1's FOM. It would be wonderful if DXTRAN did not slow the problem down so much. Fortunately in some cases, a little thinking and judicious use (described below) of the DXCPN card can alleviate this speed problem.

Recall the caveat about using DXTRAN carefully in optically thick problems, in particular, not to rely on DXTRAN to do the penetration. Geometry splitting has done well at penetration, so DXTRAN is needed mostly for the angle bias, as is desirable. However, at every collision, regardless of how many mean free paths the

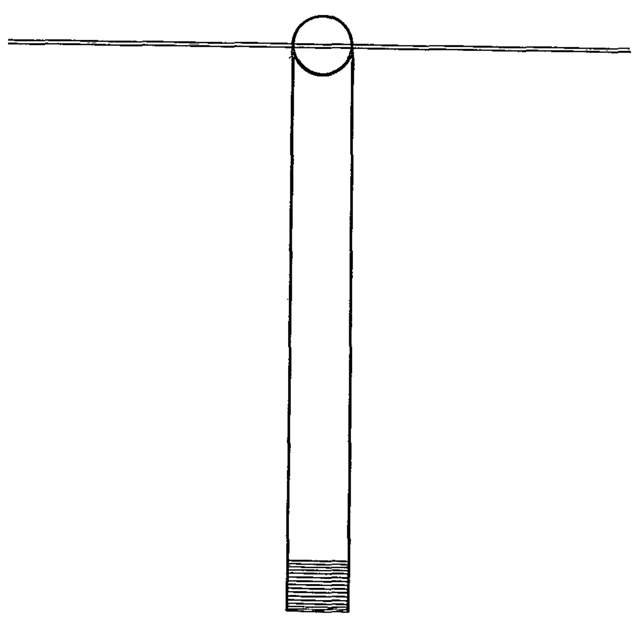


Fig. 16. DXTRAN sphere. The inner and outer spheres are identical because specifying different spheres would just create weight fluctuation.

collision is from cell 21, a DXTRAN particle is produced. DXTRAN particles that are many free paths from the DXTRAN sphere will have their weights exponentially decreased by the optical path so that their weights are negligible by the time they are put on the DXTRAN sphere. MCNP automatically (unless turned off on the DD card) plays Russian roulette on the DXTRAN particles as their weight falls exponentially, because of transport, below some fraction of the average weight (on the DXTRAN sphere). This provides the user some protection against spending a lot of time following DXTRAN particles of inconsequential weight. However, there is a better solution for the sample problem.

Although the DD card will play roulette on DXTRAN particles as they are transported through media to the DXTRAN sphere, it still takes time to produce and follow the DXTRAN particles until they can be rouletted. It is much better not to produce so many DXTRAN particles in the first place. MCNP allows the user (on the DXCPN card) to specify, by cell, what fraction of the collisions will result in DXTRAN particles. Everything is treated the same except that if p is the probability of creating a DXTRAN particle, then when a DXTRAN particle is created, its weight is multiplied by p^{-1}, thus making the game unbiased. The destruction game is unaffected; regardless of whether the sampling produced a DXTRAN particle, the non-DXTRAN particle is killed if it tries to enter the DXTRAN sphere.

As usual, this new capability requires even more input parameters; that is, the entries on the DXCPN card. Before despairing unduly, note that the entries on the DXCPN card are not highly critical, and the user has already gained a lot of useful information in the geometry-splitting optimization.

Table I shows the DXCPN probabilities that I chose for the sample problem. Note three things from this table.

1. Near the top of the concrete cylinder (cells 18 and 19) every collision creates a DXTRAN particle ($p = 1$).
2. As the cells get progressively farther (cells 12-17) from the DXTRAN sphere, p gets progressively smaller by roughly a factor of 2, chosen because the importance from cell to cell decreases by factors of about 2.
3. Not much thought was spent selecting p's for cells 2-11 because these cells contribute almost no weight to the DXTRAN sphere. Thus within reason, almost any values can be selected if they are small enough that not much time is spent following DXTRAN particles in cells 2-11. Note that even $p = 0.001$ will not totally preclude creating DXTRAN particles because there are 2000-3000 collisions in each of cells 2-5, where $p = 0.001$.

Before examining what happened when the DXCPN card was used, I would like to digress and use item 3 above as a specific example of a general principle. When biasing against random walks of presumed low importance, always make sure that at least a few of these random walks are followed so that if the presumption is wrong, the statistics will so indicate by bouncing around. As an example, I fully believe that $p = 10^{-6}$ would be appropriate in cell 2, but I chose $p = 10^{-3}$. Had I chosen $p = 10^{-6}$, probably no DXTRAN particles would be produced from collisions in cell 2. Thus if these DXTRAN particles turn out to be a lot more important than anticipated, the tally may be missing a substantial contribution with no statistical indication that something is amiss. By choosing $p = 0.001$ in cells 2-5, I cause the MCNP to produce approximately ten DXTRAN particles by the 10,000 or so collisions in cells 2-5 (see Fig. 17). Following 10 DXTRAN particles is a very small time price to pay to be sure that they are not important. If the problem were to be run long enough that there would be 10^7 collisions in cell 2, then I would not hesitate to use $p = 10^{-6}$ because *some* DXTRAN particles would be produced.

J. Results of Using DXTRAN with the DXCPN card

The result of adding the DXCPN card is shown in Fig. 18. Note that all FOMs improved by better than a factor of 2. The histories per minute increased from 1560 to 4395 when the DXCPN card was added, but 4395 is still slower than the 6858 without DXTRAN. The FOM for tally 1, although almost three times as good as that without the DXCPN card, is nonetheless still less than the no DXTRAN FOM of 45. This is an example of the general rule:

Increasing sampling in one region in general is at the expense of another region.

In the sample problem, we have decided to increase the sampling of cell 21 at the expense of cells 2-19. Overall, however, DXTRAN has clearly improved the calculation.

XI. TALLY CHOICE, POINT DETECTOR VERSUS RING DETECTOR

Recall from the introductory section on variance reduction that the FOM is affected by the tally choice as well as by the random walk sampling. So far, I have tinkered only with the random walk sampling; now, suppose I tinker with the tally.

Consider tally 5, the point detector tally. Note that the sample problem is symmetric about the y-axis, so a ring

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	2357	2301	2909	2.0318E+00	2.3810E+00	5.9965E+00	1.2916E+00	6.9395E+00
3	3	1159	1075	2685	1.2673E+00	8.7258E-01	3.6570E+00	1.7528E+00	5.6912E+00
4	4	1005	944	2340	6.0928E-01	8.1250E-01	3.2031E+00	2.0094E+00	5.4488E+00
5	5	957	901	2247	2.5856E-01	1.1096E+00	3.2913E+00	1.9192E+00	5.7046E+00
6	6	872	808	1912	1.0868E-01	1.3123E+00	3.3536E+00	1.8915E+00	6.0199E+00
7	7	793	742	1718	5.3250E-02	9.6440E-01	2.8593E+00	2.1229E+00	5.5466E+00
8	8	846	783	1899	3.3400E-02	6.8582E-01	2.5136E+00	2.3368E+00	5.3099E+00
9	9	857	796	2073	1.7817E-02	6.3392E-01	2.3447E+00	2.4164E+00	5.2447E+00
10	10	896	828	2026	7.9255E-03	7.0803E-01	2.3925E+00	2.3320E+00	5.4905E+00
11	11	941	858	2122	3.9925E-03	6.9361E-01	2.1789E+00	2.3943E+00	5.3566E+00
12	12	980	894	2287	1.9016E-03	7.1915E-01	2.1137E+00	2.4458E+00	5.2989E+00
13	13	983	895	2319	1.0406E-03	5.0262E-01	1.7716E+00	2.7835E+00	4.9197E+00
14	14	974	907	2445	5.6999E-04	4.7699E-01	1.5858E+00	2.9712E+00	4.7692E+00
15	15	906	844	2156	1.9251E-04	7.1176E-01	1.8944E+00	2.5154E+00	5.3166E+00
16	16	892	826	2223	8.8589E-05	7.2278E-01	1.8604E+00	2.5116E+00	5.1586E+00
17	17	914	840	2202	4.1190E-05	6.0178E-01	1.8622E+00	2.5492E+00	5.1533E+00
18	18	897	828	2091	1.7260E-05	6.4129E-01	1.8927E+00	2.5250E+00	5.2535E+00
19	19	755	731	1823	6.8759E-06	5.8583E-01	1.8406E+00	2.5705E+00	5.2381E+00
20	20	2172	9716	0	0.	7.6847E-01	2.5674E+00	3.0960E-01	1.0000+123
21	21	8813	17634	9015	6.7811E-11	1.7049E+00	3.6797E+00	3.1082E-04	7.3565E+02
22	22	703	703	0	0.	9.0071E-01	2.6050E+00	1.5934E-05	1.0000+123
TOTAL		29672	44854	48492	4.3959E+00				

INCREASED FROM 44
WITHOUT DXTRAN

	TALLY 1			TALLY 4			TALLY 5		
NPS	MEAN	ERROR	FOM	MEAN	ERROR	FOM	MEAN	ERROR	FOM
1000	7.31774E-07	.3642	10	1.11808E-13	.3077	14	6.89810E-17	.3427	11
2000	7.18177E-07	.2485	12	1.14993E-13	.2149	17	7.12499E-17	.2307	14
2231	7.34682E-07	.2293	12	1.23719E-13	.2053	15	7.62305E-17	.2177	14

```
*******************************************************************************
DUMP NO.   2 ON FILE RUNTPG            NPS =    2231         CTM =     1.43
```

INCREASED BY
FACTOR OF 7

DECREASED BY
FACTOR OF 4

PART / MIN = 1560
WITHOUT DXTRAN = 6858

CONCLUSION: DXTRAN TECHNIQUE SUCCESSFUL FOR TALLY 4 AND TALLY 5 BUT TOO SLOW.

Fig. 17. DXTRAN sphere at about cell 21.

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	11938	11744	13796	1.9875E+00	1.9553E+00	5.8909E+00	1.3213E+00	6.8365E+00
3	3	5292	4918	11952	1.0727E+00	1.1897E+00	4.0414E+00	1.6629E+00	5.9855E+00
4	4	5010	4651	11384	5.7591E-01	8.6965E-01	3.3229E+00	1.9476E+00	5.6454E+00
5	5	4827	4469	10847	2.7700E-01	7.9325E-01	3.0387E+00	2.0971E+00	5.5575E+00
6	6	4414	4073	9805	1.2674E-01	7.8981E-01	2.9908E+00	2.1426E+00	5.6477E+00
7	7	4177	3816	9507	6.2377E-02	7.9364E-01	2.8062E+00	2.2181E+00	5.5340E+00
8	8	4114	3780	9500	3.2883E-02	7.5226E-01	2.5813E+00	2.3164E+00	5.4448E+00
9	9	4114	3803	9158	1.5241E-02	7.2149E-01	2.5273E+00	2.3520E+00	5.4441E+00
10	10	4105	3833	9297	7.4165E-03	7.0749E-01	2.4918E+00	2.3750E+00	5.4572E+00
11	11	4112	3803	9202	3.2429E-03	7.5649E-01	2.4433E+00	2.3664E+00	5.4819E+00
12	12	4293	3948	9827	1.7202E-03	6.9620E-01	2.3424E+00	2.4325E+00	5.3524E+00
13	13	4384	4040	9759	8.0545E-04	6.5721E-01	2.2911E+00	2.4785E+00	5.3043E+00
14	14	4337	4017	9621	3.6893E-04	6.6088E-01	2.2587E+00	2.4924E+00	5.3570E+00
15	15	4312	3977	9935	1.7810E-04	6.6189E-01	2.1957E+00	2.5004E+00	5.3071E+00
16	16	4365	4059	10039	7.9894E-05	7.0947E-01	2.1324E+00	2.5102E+00	5.3140E+00
17	17	4274	3982	9862	3.6959E-05	6.3603E-01	2.0865E+00	2.5629E+00	5.2996E+00
18	18	4248	3935	9946	1.6748E-05	6.6484E-01	2.0783E+00	2.5527E+00	5.3146E+00
19	19	3867	3749	8538	6.2137E-06	7.0403E-01	2.2236E+00	2.4620E+00	5.4511E+00
20	20	4927	18224	0	0.	1.2396E+00	3.2943E+00	7.9810E-01	1.0000+123
21	21	15223	30451	15598	6.5663E-11	1.7785E+00	4.1554E+00	9.1074E-04	7.3805E+02
22	22	1283	1283	0	0.	7.0419E-01	2.7866E+00	3.6840E-05	1.0000+123
TOTAL		107616	130555	197573	4.1643E+00				

SUBSTANTIALLY IMPROVED

NPS	TALLY 1 MEAN	ERROR	FOM	TALLY 4 MEAN	ERROR	FOM	TALLY 5 MEAN	ERROR	FOM
1000	9.91356E-07	.3667	27	1.50695E-13	.3251	34	1.06148E-16	.3673	27
2000	6.41539E-07	.2949	26	1.06818E-13	.2461	38	6.85473E-17	.2928	27
3000	5.70257E-07	.2393	27	1.02834E-13	.1982	40	6.49562E-17	.2387	27
4000	6.76150E-07	.1863	32	1.11525E-13	.1622	43	6.79132E-17	.1910	31
5000	6.76891E-07	.1679	32	1.13338E-13	.1465	42	6.89970E-17	.1686	31
6000	6.73265E-07	.1516	32	1.14934E-13	.1312	43	6.74936E-17	.1510	33
7000	6.89040E-07	.1378	34	1.14298E-13	.1198	45	6.74760E-17	.1359	34
8000	6.86656E-07	.1262	35	1.17711E-13	.1095	46	6.87251E-17	.1245	36
9000	7.04305E-07	.1167	36	1.18117E-13	.1023	47	6.97952E-17	.1161	36
10000	7.25099E-07	.1093	36	1.22116E-13	.0990	44	7.09365E-17	.1114	35
11000	7.00085E-07	.1046	36	1.18453E-13	.0948	45	6.88873E-17	.1060	36
11427	7.32339E-07	.1049	34	1.22412E-13	.0946	42	7.21438E-17	.1049	34
						15 LAST TIME			14 LAST TIME

****

DUMP NO. 2 ON FILE RUNTPH NPS = 11427 CTM = 2.60

PART/MIN = 4395
NO DXTRAN = 6858
DXTRAN w/o DXCPN = 1560

IMPROVED OVER DXTRAN w/o DXCPN = 12
LESS THAN NO DXTRAN = 45
"INCREASING SAMPLING IN ONE REGION GENERALLY IS AT THE EXPENSE OF ANOTHER REGION"

Fig. 18. DXTRAN with DXCPN card.

CELL PROGR	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	12267	12069	14170	1.9999E+00	1.9012E+00	5.8573E+00	1.3269E+00	6.8191E+00
3	3	5437	5058	12296	1.0869E+00	1.1533E+00	4.0162E+00	1.6748E+00	5.9734E+00
4	4	5161	4795	11784	5.8976E-01	8.2627E-01	3.2750E+00	1.9771E+00	5.6004E+00
5	5	4995	4650	11188	2.7224E-01	8.2094E-01	3.0841E+00	2.0808E+00	5.6112E+00
6	6	4531	4188	9948	1.2760E-01	7.8146E-01	2.9328E+00	2.1763E+00	5.5937E+00
7	7	4356	3990	9656	6.1955E-02	7.7673E-01	2.8035E+00	2.2300E+00	5.5322E+00
8	8	4249	3911	9806	3.3255E-02	7.2755E-01	2.5684E+00	2.3299E+00	5.4246E+00
9	9	4262	3953	9630	1.5418E-02	7.5269E-01	2.4927E+00	2.3562E+00	5.4483E+00
10	10	4227	3940	9239	7.1770E-03	7.1804E-01	2.4483E+00	2.3960E+00	5.4227E+00
11	11	4360	4043	9877	3.5594E-03	6.9409E-01	2.4199E+00	2.4038E+00	5.4096E+00
12	12	4523	4166	10278	1.7099E-03	7.1522E-01	2.3994E+00	2.3964E+00	5.4387E+00
13	13	4502	4173	10136	7.7209E-04	7.3122E-01	2.3975E+00	2.3815E+00	5.4710E+00
14	14	4367	4043	10118	3.6972E-04	7.0748E-01	2.2789E+00	2.4366E+00	5.4459E+00
15	15	4254	3951	9909	1.7815E-04	6.2624E-01	2.1304E+00	2.5420E+00	5.3036E+00
16	16	4238	3935	9817	8.2112E-05	5.6490E-01	2.0276E+00	2.6416E+00	5.1548E+00
17	17	4138	3844	9647	3.5603E-05	6.3267E-01	2.0781E+00	2.5646E+00	5.2647E+00
18	18	4038	3769	8985	1.4550E-05	6.8669E-01	2.1483E+00	2.5146E+00	5.3785E+00
19	19	3752	3656	8076	5.7453E-06	7.1626E-01	2.2956E+00	2.4092E+00	5.5517E+00
20	20	4746	17086	0	0.	1.3419E+00	3.3144E+00	7.3638E-01	1.0000+123
21	21	14236	28475	14579	6.1953E-11	1.6976E+00	4.1392E+00	9.5542E-04	7.5010E+02
22	22	1203	1203	0	0.	5.2820E-01	2.0685E+00	5.5808E-05	1.0000+123
TOTAL		107842	128898	199139	4.2009E+00				

≈ SAME AS BEFORE

NPS	TALLY 1 MEAN	ERROR	FOM	TALLY 4 MEAN	ERROR	FOM	TALLY 5 MEAN	ERROR	FOM
1000	8.28190E-07	.3164	37	1.46477E-13	.3050	40	7.26710E-17	.2824	46
2000	7.03344E-07	.2359	38	1.29340E-13	.2278	40	6.75743E-17	.2308	39
3000	6.14756E-07	.2082	34	1.10453E-13	.1961	39	5.88587E-17	.1959	39
4000	6.85420E-07	.1707	37	1.18986E-13	.1612	42	6.70328E-17	.1642	40
5000	6.68980E-07	.1523	39	1.12148E-13	.1417	45	6.32091E-17	.1454	42
6000	6.26705E-07	.1427	38	1.08285E-13	.1321	44	6.11213E-17	.1363	41
7000	6.31120E-07	.1302	38	1.10299E-13	.1217	44	6.27649E-17	.1267	41
8000	6.48337E-07	.1196	39	1.13456E-13	.1108	46	6.52710E-17	.1185	40
9000	6.79583E-07	.1123	39	1.17471E-13	.1045	45	6.79360E-17	.1102	40
10000	6.78869E-07	.1045	40	1.18658E-13	.0984	45	6.86693E-17	.1028	41
11000	6.65474E-07	.1032	38	1.17063E-13	.0965	43	6.74821E-17	.1001	40
11755	6.53648E-07	.1000	38	1.16162E-13	.0930	44	6.77615E-17	.0964	41

****
DUMP NO. 2 ON FILE RUNTPE NPS = 11755 CTM = 2.60

≈ SAME

INCREASED FROM 34 WITH POINT DETECTOR
(MORE DIFFERENCE WOULD BE SEEN IN A SCATTERING MEDIUM)

Fig. 19. Ring detector.