Practical Monte Carlo Using MCNP

1 INTRODUCTION

1.1 Objectives

The aim of this course is to introduce the basic concepts of the Monte Carlo method as used in radiation transport calculations and to provide a practical introduction to the widely used, general purpose, Monte Carlo code, MCNP.

1. To introduce the basic concepts of the Monte Carlo method in radiation transport calculations
2. Introduce concepts of variance reduction
3. To explain some of the features of the general purpose Monte Carlo code, MCNP
4. Provide a practical introduction to and hands-on experience of using MCNP for simple photon and neutron transport problems.

1.2 Why use the Monte Method for radiation transport calculations?

To solve multiple-scatter problems.

Deterministic methods:

• Solves transport equation for average particle behaviour (Eulerian)

• The solution is generally obtained in analytical form throughout the phase-space of the problem

• Gives complete information

• Only in certain easy geometries - but can combine solutions to form more complex examples

Monte Carlo methods:

• Behaviour of individual particles is simulated (Lagrangian) and the mean behaviour of all particles is inferred according to the *law of large numbers* and the *central limit theorem*

• Only answers specific (user-defined) questions (e.g. what is the average flux in a particular volume?)

• Is well suited to complex 3D (time-dependent) problems which would prove difficult, or impossible, to solve analytically

• Can deal with mixed particle neutron/photon/electron coupled problems - e.g. photons produced by neutrons

• Usually no need to group cross-sections, continuous energy can be used

• Compared to deterministic methods in simple geometries, MC is slow!

• Potentially limited in its accuracy only by the accuracy of the cross-sections, the description of the materials in the problem and the statistical nature of the problem (have we run enough histories?)

• ***Warning!*** It is easy for the novice to produce nonsensical answers.

1.3 Examples of use of Monte Carlo for radiation transport

Design of radiation equipment and shielding, radiography, radiotherapy treatment planning, storage of radioactive materials, nuclear reactor modeling (criticality), ...

2 CONCEPTS OF MONTE CARLO RADIATION TRANSPORT

2.1 Conditions for a valid MC simulation

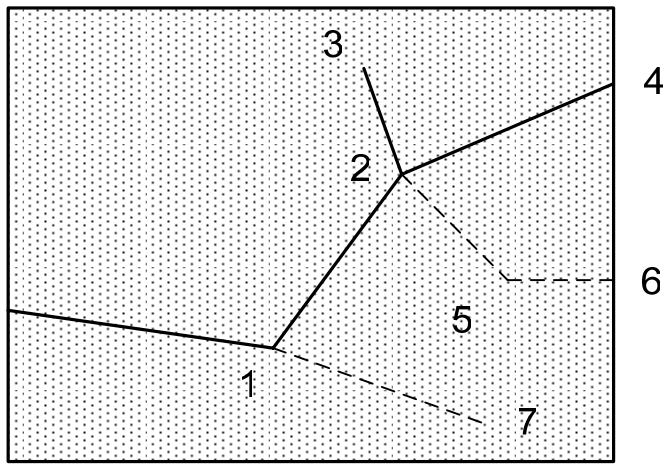
• Source description must represent the real source (energy, angle and spatial distributions and time dependency)

• The problem geometry must adequately represent the real situation

• The materials present in the system must be accurately described, including all significant impurities (especially important in some neutron problems)

2.2 A neutron in fissile material - an *anecdote*

Processing order and the last-in-first-out principle



Example *history* in fissile material

1. Neutron scatter & photon production
2. Fission event: two neutrons & photon production
3. Capture (end)
4. Neutron escapes (end)
5. Photon scatter
6. Photon escapes (end)
7. Photo-absorption (end)

When interaction occur producing two or more outgoing particles a stacking procude most be invoked. In this case the particle with the higher energy is processed first. Other daughter particles are placed on the stack to be processed later. When all particles on the stack are processed and tracked to completion we will have completed one *history*. In practice one may need to run tens of millions of histories to achieve a reasonable variance (see Johns and Cunningham page 186 for a similar "case history" presentation of the Monte Carlo method).

3 COLLISION DISTANCE AND INTERACTION SELECTION

3.1 How does a MC code know where an interaction occurs?

Say that a particle must interact at a point X, distance l into cell 1.

Probability of interaction within dl is

$$P(l) = \exp(-\sigma l)\sigma \mathrm{d}l$$

where σ is the total cross-section of material in cell 1, i.e.

Compton + photoabs. + pair prod. (γ) or

elastic + inelastic + reaction (n)

ε is a random number between 0 and 1 [0,1)

The probability of interaction anywhere along the path to X is

$$\varepsilon = \int_0^l \exp(\text{-}\sigma s)\sigma \mathrm{d}s = -\left[\exp(\text{-}\sigma s)\right]_0^l$$

which must be somewhere between 0 and 1

therefore

$$l = -\frac{1}{\sigma}\ln(1-\varepsilon) = -\frac{1}{\sigma_t}\ln(\varepsilon)$$

since ε is uniformly distributed between 0 and 1.

If l < D, an interaction occurs at X within cell 1.

If $l \geq$ D, no interaction occurs in cell 1, repeat for cell 2 (unless a void)

3.2 Which nuclide and what kind of interaction?

Nuclide selected by macroscopic total cross-section

Material with n nuclides: Total material cross-section Σ

$$\Sigma = \sum_{i=1}^{n} \sigma_i$$

Generate random number ε within interval [0,1), kth

Nuclide is chosen if

$$\sum_{i=1}^{k-1} \sigma_i < \varepsilon \sum_{i=1}^{n} \sigma_i \leq \sum_{i=1}^{k} \sigma_i$$

3.3 Angle and energy selection after collision

Cosine of scattering angle (or angle of emitted particle) is sampled from angular distribution tables: energy-dependent for each collision nuclide

Energy of scattered particle is then determined by kinetics

$$E_{out} = E_{in}\left[\frac{1+A^2+2A\mu}{1+A^2}\right]$$

where A = mass of collision nuclide (in units of neutron masses) and μ = the centre-of-mass cosine of scattering angle.

3.4 Random number generation

Non-deterministic sources of random numbers

Requires a natural, random, process; roulette wheel, particle decay

Most 'natural' sources of random numbers show some non-randomness

The calculation is not reproducible

Pseudo-random numbers

Most common RNG, the multiplicative linear congruential algorithm, exploits the properties of prime numbers and the modulus operation

$n_{i+1} = (an_i) \mod m$

For a multiplier a, typically a prime of a similar order to the modulus, m, is used.

For a 32-bit operating system, $m = 2^{32}$, a = 663608941, is shown to be a reasonable generator with a period of around $2^{30} = 10^9$ numbers.

A random number distributed (0, 1) is given by n_{i+1}/m

```
#include <stdio.h>
#include <math.h>
#include <inttypes.h>
#include <sys/types.h>
int main(int argc, char **argv)
{
	int64_t rng, iPi, jPi, tot=0, two_to_the32, two_to_the62;
```

```
    register int i;
    two_to_the62 = pow(2,62);
    two_to_the32 = pow(2,32);
    m = 2227057010910366687;
    ir = 1234567890123456789;
    for (i=0; i<100000000-1; i++) {
        rng = rng*m;
        iPi = (rng/two_to_the32);
        rng = rng*m;
        jPi = (rng/two_to_the32);
        if ((iPi*iPi + jPi*jPi) < (two_to_the62)) tot = tot+1;
    }
    printf("%i \n", tot*4);
}
```

MCNP gives a warning if the period of the pseudo-RNG is reached. There may not be an immediate impact on the estimate of variance because RNG's are used for a variety of different purposes but eventually a history will be repeated as well as all subsequent histories. The variance under these circumstances is stationary.

Quasi-random numbers

Attempts to create a more uniform source of random numbers by avoiding previously sampled regions, e.g. sampling without replacement, 1/2, 1/4, 3/4, 1/8, 3/8, 5/8, 7/8, ...

Error improves as O(1/*N*).

Law of large numbers applies. It can be shown (Evans and Swartz) that as well as for a random variable, ε distributed on (0, 1), that for any irrational number ζ, in the limit n → ∞,

$$\int_0^1 f(s)\,ds = \sum_{i=1}^{n} f(\varepsilon) \Big/ n = \sum_{i=1}^{n} f(i\varsigma \mod 1) \Big/ n$$

4 STATISTICAL ANALYSIS

4.1 Concept of particle weight

Weight, W_j, is a property of a particle, j, and denotes its likelihood relative to the physical (analogue) model.

Used for variance reduction where non-physical modeling may improve the efficiency of the calculation.

4.2 Definitions

• Monte Carlo samples the possible random walks and assigns a *score* x_i to each walk, e.g. x_i is the contribution to a flux *tally* from the ith random *history*.

• Typically random walks will produce a range of scores- including zero scores- towards the tally.

• f(x) is the history scoring *probability density function*

• The *true mean* (expectation value) is

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

However, f(x) is not known.

• The *estimated mean* is

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

where N is number of histories run.

• The *law of large numbers*: If E(x) is bound, x tends to the expectation, E(x), as N approaches infinity.

• The *central limit theorem* says that if N is large but finite, *x* is distributed about E(x) in accordance with a normal (Gaussian) distribution.

• *Variance:*

$$s^2 = \int (x - E(x))^2 f(x) dx = E(x)^2 - (E(x))^2$$

the 2nd moment of f(x)

- The *standard deviation* is s
- Estimated *variance* of a *population*

$$S^2 = \frac{1}{N\text{-}1}\sum_{i=1}^{N}\left(x_i - \overline{x}\right)^2 \approx \overline{x^2} - \overline{x}^2$$

for large N.

- Estimated *variance* of the *mean*,

$$S_{\overline{x}}^2 = \frac{S^2}{N}$$

- Estimated relative error, R

$$R = S_{\overline{x}} / \overline{x}$$

Note that $S_{\overline{x}} \propto \frac{1}{\sqrt{N}}$ is a drawback of Monte Carlo. Need to run four times the number of histories to reduce the error in the mean by a factor of two.

MCNP uses tests to check whether the tally is well behaved.

1. Is the mean fluctuating randomly?
2. Is the relative error less than 0.1? (0.05 for point detectors)
3. Is the relative error decreasing as $1/\sqrt{N}$?
4. Is the variance of the variance decreasing at $1/N$?

The v-o-v is 4*th* moment of $f(x) = \int x^4 f(x)\,dx$, so it's very sensitive to large x contributions.

5 VARIANCE REDUCTION METHODS

Alternatively, reduce S^2 by better random walk sampling to reduce the spread of the tally results about the mean. Variance reduction methods can achieve this.

Also, we will have some histories which will contribute nothing to the tally because of absorption or motion in the 'wrong' direction. It can reduce the computer time spent on these 'zero contribution' histories by implicit capture and population control.

5.1 Implicit Capture

Let's consider a process which occurs after nuclide selection. Particle with weight W_j collides with absorbing nuclide i.

σ_i = total cross-section for nuclide *i*

$\sigma_{ab,i}$ = absorption cross-section for nuclide *i*

Instead of a particle being absorbed, particle continues on its way but with its weight reduced by

$$W_j(\text{new}) = W_j\left(1 - \sigma_{ab,i}/\sigma_i\right)$$

which is unbiased. A lower weight cut-off limit is employed to prevent very low weight particles being tracked.

5.2 Population Control

Not one method, but a type of method used by MCNP to boost the particle population in the region of interest at the expense of other regions.

Example: Geometry splitting by cell importance

Suppose we wish to track neutrons or photons through a very thick scattering (and absorbing) medium- e.g. a radiation shield.

If the transmission is only 0.001%, then we will need to run 10^9 histories to have at least 10000 particles emerging from the other side (1% error in total fluence).

Divide medium into 13 slices of thickness approximately one mean free path and have increasing "importances" within each slice. The particle splits into two at each surface within the material and each new particle has half the weight of the parent. Weight is conserved, not number of particles.

This way we still have many particles surviving through to the tally side. It should aim to keep the particle population constant in each slice through the shield.

IMP:N 1 1 2 4 8 16 32 64 128 256 ... 8192

Now approximately 1 particle will emerge per history.

Russian Roulette

For particle moving in the 'wrong' direction we kill them off with a known probability, but surviving particle have correspondingly higher weight.

We can use the same splitting and Russian roulette methods with particle energy.

5.3 Other VR methods

Energy and Weight Cut-offs

Kill particles below a specified energy or weight. This introduces a bias and the user should be aware of the significance of this. However it is typical and arguably necessary (avoids a kind of "Xeno paradox" relating to MC). An appropriate choice of energy cut-off can save a great deal of computer time.

Source Biasing

If we are interested only in a particular direction, we can bias the source in this direction by changing the angular distribution of the number of source particles from its natural distribution, with a corresponding adjustment of the weights.

Point Detector Tally (Type 5, e.g. F15:N)

It is a type of tally which is often more efficient than cell or surface tallies.

Cannot use track length or surface tallies to determine the flux at a point.

User specifies a point in (x, y, z) coordinates.

At each collision and for each source particle, MCNP calculates the probability of the scattered particle reaching the point according to the relation

$$P(\text{detection}) = \frac{Wf(\mu)\exp(-\sigma R)}{2\pi R^2}$$

where R = distance from scatter point to detector

W = particle weight

$f(\mu)$ = pdf at μ (cosine of angle between particle trajectory and direction to detector)

Contributions are made to $f(x)$ at every interaction point, hence good statistics built up quickly. However as R becomes small, we have very large contributions - unstable tally means if geometry is complicated around the position of the point tally.

6 MCNP - AN OVERVIEW AND SOME FEATURES

- **M**onte **C**arlo **N** - **P**article transport code
- A general-purpose code - not written for treatment planning or reactor design, therefore *not ideal* for either
- However, it is very versatile and ideally suited for research and design/engineering work

6.1 Types of interactions handled by MCNP

Neutrons:

Elastic scattering, inelastic scattering, absorption, fission, (n, xn), photon production, $S(\alpha, \beta)$ scattering at low neutron energies.

Photons and neutrons, thus produced, are tracked in mode NP.

Photons:

Compton (incoherent scattering) - angle/energy determined by Klein-Nishina formula

Thompson (coherent) scattering - no energy loss

Photoelectric effect - absorption of incident photon, then emission of fluorescent photons (and electron for a mode PE problem)

Pair production - incident photon track terminates. In mode P both 511 keV photons are tracked. In mode PE, the electron/positron pair are tracked, with subsequent positron annihilation.

Electrons and positrons:

Strongly interacting due to Coulombic forces - single (Rutherford) interactions not

feasible Multi-scattering/condensed history steps - energy loss calculated for fixed energy loss, E, or step length, s, from calculated collisional stopping power, dE_{coll}/ds, Bremsstrahlung, fluorescent photons and knock-on electrons are sampled in an analogue fashion.

Positrons create two annihilation, 511 keV photons when the energy cut-off is reached.

6.2 Format of MCNP input 'cards'

A card is a single line of MCNP input, contains instructions on how to perform the simulation - geometry, tallies, mode etc.

Columns 1-5 are reserved for the card type declarator (e.g. surface, cell, comment, tally, etc.), columns 6-80 are for the parameters associated with this card.

6.3 MCNP input cards

6.3.1 Surfaces

There is a considerable array of surface types which can be made to nearly fit most geometries of interest. See table 3.1 from MCNP manual for a summary of surface types and parameters.

Planes:

PX	1.0	$plane perpendicular to x-axis at x=1.0cm
PZ	-10.3	$plane perpendicular to z-axis at z=-10.3cm
P x1 y1 z1 x2 y2 z2 x3 y3 z3		$plane defined by 3 points with the co-ordinates (x1,y1,z1)

Spheres:

SO	100.1	$sphere centred on origin of radius 100.1 cm
SY	10.0 3.0	$sphere centred on the y-axis at y = 10.0 cm, radius = 3.0cm
S	1.0 2.0 4.5 2.0	$sphere centred on (1.0, 2.0, 4.5) of radius 2.0 cm

Cylinders:

CY 1.0 $cylinder with axis along y-axis, radius = 1.0 cm

C/Z 3.0 5.0 2.4 $cylinder parallel to z-axis centered on (x, y) = (3, 5)cm, radius = 2.4 cm

Can also have cones (single and double), general ellipsoids, elliptical or circular tori, hyperboloids, paraboloids (general quadratics).

6.3.2 Cells

Cells are defined with respect to one or more surfaces Each cell has a unique identifying number. Each cell contains a single material or is void The material in each cell has only one density.

Examples:

```
C Cell cards
1 0          -1 $ Void cell, inside surface 1 (negative sense)
2 1 -2.7 1 -2 $ Aluminium shell, outside 1, inside 2
3 0          -2 $ Void cell outside 2
(blank line delimiter)
C Surface cards
1 SO      10.0     $ Sphere centred on the origin (radius 10 cm)
2 SO      15.0     $ Sphere centred on the origin (radius 15 cm)
3 CX      5.0      $ Cylinder on x-axis
4 PX      25.0     $ Plane perpendicular to x-axis
5 PX      0.0      $ Plane perpendicular to x-axis
(blank line delimiter)
C Material cards
M1 13000 1.0       $ Aluminium, 100%
(blank line delimiter)
```

Surface are combined either with the intersection operator (blank space) or the union operator, (: the colon) e.g.

```
2    1 -2.7    (1 -2):(2 -3 -4 5)          $ Spherical shell on a cylindrical stick
```

Parentheses are used here to group the surfaces 1 and -2 into one subcell and surfaces 2, -3, -4 and 5 into another. These are then combined with the union operator to give the geometry shown as shaded in the figure.

6.3.3 Material cards

The *substance* of your simulations

• Materials are defined by a unique number preceded by M, e.g. M4

• Defined using the atomic number of the element(s) followed by three 0s (for photon cross-section files) or a three digit atomic mass (for neutrons)

• After the ZAID, we have the fraction of that element (+ve for atom fraction, -ve for weight fraction)

e.g. for a photon and neutron problems the specifications are different,

M3 1000 0.6667 8000 0.3333

M3 1002.55C 0.66 1001.50C 0.00667 8016.50C 0.33333 $ 99% enriched heavy water

The .55C and .50C refer to the particular recommended continuous energy cross-section files for these nuclides - these are not actually included in the 'lite' distribution. Refer to appendix G in the MCNP manual for the full suite of cross-section data.

6.4 Tallies

What do you want to know?

Seven standard tallies (can also supply your own)

• F1:N, F1:P Surface current (e.g. F1:N, F11:N, F21:N to F991:N, 100 tallies max)

• F2:N, F2:P Surface flux

• F4:N(P) Track length estimate of cell flux

• F5:N(P) Flux at a point (point detector - or ring for symmetric problems)

• F6:N(P) Track length estimate of energy deposition in a cell

• F7:N Track length estimate of fission energy deposition

• F8:N(P) Pulse height tally (for physical detector modeling)

F42:P 103 will tally the photon flux at surface 103

F104:N 12 will tally the neutron flux averaged over cell 12

Flux tallies can be multiplied by other factors (using an FM card) to give nuclear reaction rates, dose or kerma, nuclide production rates.

6.4.1 Energy binning

We obtain spectral information from a tally using the En card, e.g.

F11:N 109

E11 1E-7 1E-6 1E-5 1E-4 ... 1.0 10.0

FM11 1E7 1.111E6 1.111E5 1.111E4.... 1.111 0.1111

will provide a neutron spectrum divided into decade bins in units of n/cm2/MeV per source particle. Use E0 (FM0) to apply to all tallies.

6.4.2 Source specification

To run a Monte Carlo simulation we need a source of particles

MODE	Default particle type
MODE P	Photons, also MODE P E
MODE N	Neutrons
MODE N P	Neutrons (neutron-induced photons)
MODE N P E	Neutrons
MODE E	Electrons
MODE P E	Photons

Source definition card

MODE N P

SDEF POS=0 1.0 10.0 ERG=14.0 VEC=0 1 0 DIR=D1

SI1 H 0.966 1.0

SP1 0.0 1.0

Defines a mono-energetic neutron source at (x = 0, y = 1, z = 10 cm) of energy 14 MeV. The direction is a cone of half angle 15° = $\cos^{-1}(0.966)$ pointing in the y direction.

It can have angle-dependent energy distribution, position-dependent angle distribution, volume or area distribution.

6.5 MCNP on the server

A 'lite' version of MCNP4C is available on the undergraduate server (ugs5). A full set (Z = 1-99) set of neutron, photon and electron are provided, including the publically available ENDL '85 and '92 database.

To access the server you will need to run the eXceed windows application. eXceed is a X server from which you can enter command-line instructions and run graphics oriented packages.

6.5.1 Starting off with Unix

You will need to make a local copy of MCNP and its cross-sections in the working directory of your user account.

```
mkdir mcnp_working_directory
cd mcnp_working_directory
ln -s /usr/local/mcnp4c/xsdir xsdir
ln -s /usr/local/mcnp4c/mcnp mcnp
cp /usr/local/mcnp4c/*.ip . $ copy the input files to your local directory
xedit demo1.ip &
```

The basic form of the command line execution of mcnp is

mcnp ip inp=demo1.ip outp=demo1.op

where "ip" denotes that the plotting package is begin used and is ommitted when the input file is properly debugged.

7 EXTENSIONS

7.1 Electrons: Transport mechanics

Analogue Monte Carlo is too slow.

1 MeV electron takes around 10^5 interactions to reduce energy by 1/8th c.f. 20 for a 1 MeV photon.

The *continuous slowing down approximation*

The energy of the particle is a deterministic function of the path length

$$E(s) = E_0 + \int_0^s \frac{dE(s')}{ds} ds'$$

dE/ds is the total stopping power, S_{tot}

Multi-scattering steps across surfaces (separating materials of different composition) most be treated carefully (See Berger 1963).

7.2 Adjoint Monte Carlo

Transport occurs with time reversed i.e. particles generally increase energy.

It is useful for the optimization of a detector problem where most of the interest is in particles that are absorbed by the detector.

It allows us determine the importance, specific to the problem at hand, of regions contributing the most scatter into the detector.

Adjoint MC → importance map → forward MC.

7.3 Criticality

Use the KCODE card.

Choose N histories per iteration and guess an initial value of k_{eff}, the eigenvalue of the criticality problem.

Number of neutrons per cycle is described by $n \rightarrow nk_{eff}$

Critical when $k_{eff} = 1$

(see MCNP manual 2-143, Hammersley and Handscomb, Davis 1963)