NE585
NUCLEAR FUEL CYCLES
Monte Carlo methods & MCNP 5

R. A. Borrelli

University of Idaho

University of Idaho
Department of Nuclear Engineering
and Industrial Management

Idaho Falls Center for Higher Education

Learning objectives

Modeling of nuclear systems

Demonstrating skill with with MCNP

Developing input files

Interpreting output

Lots of content on the OER

Learning nodes

Monte Carlo theory
Approximating an integral
Computing π
Multidimensional integration
Random sampling
Boltzmann transport equation

Solution procedure
Assumptions
Superposition

Random numbers

MCNP modeling
What it does
Particle tracking
Tally
Input decks
Geometry
Neutron flux example
Surfaces
Sense
Cells
Universes

Other tips

Additional information

More learning nodes

MCNP quality control
Mean & variance
Other statistical checks

Criticality

Variance reduction

Monte Carlo theory

Monte Carlo simulates particle transport

Particles like neutrons, but photons too

Because we're engineers we are going to focus more on application than theory

There are a lot of Monte Carlo codes (OpenMC)

We will use the latest version of MCNP

Fundamentals of Monte Carlo particle transport

Forrest B. Brown

LA-UR-05-4983

Monte Carlo was invented by von Neumann

Fairly successful guy

Basically invented what we call ‘scientific computing’

Monte Carlo is highly accurate but expensive computationally

MCNP can be run on just about all architectures but use linux

Monte Carlo methods solve the transport equation

Mathematical approach is for importance sampling, convergence, variance reduction, random sampling techniques, eigenvalue calculation schemes

Simulation approach is for collision physics, tracking, tallying

Monte Carlo methods solve integral problems, so consider the integral form of the Boltzmann equation

Most theory on Monte Carlo deals with fixed-source problems

Eigenvalue problems are needed for criticality and reactor physics calculations

Should know basic statistics

Approximating an integral

Monte Carlo approximates the area under a curve

$$G = \int_0^1 g(x)dx \tag{1}$$

$$g(x) = \sqrt{1 - x^2} \tag{2}$$

May not always be able to compute the integral

$$G \approx \frac{1}{N} \sum_{k=1}^{N} \sqrt{1 - x_k^2} \tag{3}$$

Choose randomly – $x \in (0, 1)$

Monte Carlo approximates the area under a curve

Or –

$$x_k^2 + y_k^2 \leq 1, k = 1, .., N \tag{4}$$

Sample x and y and count a 'hit' if less than 1

$$G \approx (1 \cdot 1)\frac{hits}{N} \tag{5}$$

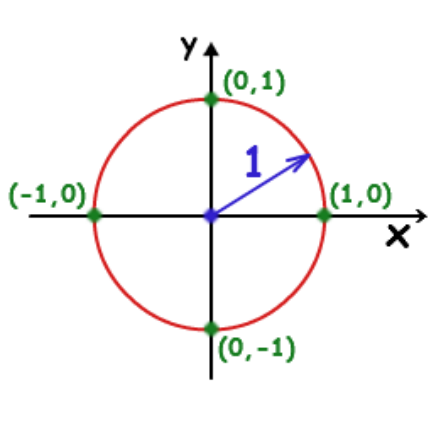
Analytical solution can be obtained here

$$G = \frac{1}{2}(x\sqrt{1-x^2} - sin^{-1}x) \quad (6)$$

So the Monte Carlo routine can be checked

Code on GitHub

See also Brown 1-6



Computing π

Use Monte Carlo to compute π

$$x^2 + y^2 = r^2 \tag{7}$$

$$A = \pi r^2 \tag{8}$$

Sample $x \in (0, r)$

Sample $y \in (0, r)$

That is the first quadrant of the circle

Record hits

Divide hits by sample number like before

$\therefore \pi \approx 4 \cdot \frac{hits}{N}$

Multidimensional integration

Monte Carlo is good for multidimensional integration

The procedure is the same

$$G = \int_{a_1}^{b_1} \dots \int_{a_n}^{b_n} g(r_1, \dots, r_n) dr_1 \dots dr_n \tag{9}$$

$$G \approx (b_1 - a_1) \dots (b_n - a_n) \frac{1}{N} \sum_{k=1}^{N} g(r_1^k, \dots, r_n^k) \tag{10}$$

$$r_n^k \in (a_n, b_n) \tag{11}$$

Random sampling

The key to Monte Carlo methods is the notion of random sampling

For a given pdf $p(x)$ produce actual *Xs* distributed in the same way

Models the outcome of physical events

Neutron scatting, absorption, fission

1-8 – 1-10

Boltzmann transport equation

Monte Carlo is used to simulate the transport equation

$$\Psi(\underline{r},\underline{v}) = \int [\int \Psi(\underline{r}',\underline{v}')C(\underline{v}' \rightarrow \underline{v})d\underline{v}' + Q(\underline{r}',\underline{v})]T(\underline{r}' \rightarrow \underline{r})d\underline{r} \quad (12)$$

$\Psi(\underline{r},\underline{v})$ – particle collision density

$Q(\underline{r}',\underline{v})$ – source term

$C(\underline{v}' \rightarrow \underline{v})$ – collision; change velocity at fixed position

$T(\underline{r}' \rightarrow \underline{r})$ – transport; change position at fixed velocity

The source term can become exceedingly complex

$Q(\underline{r}, \underline{v}) = S(\underline{r}, \underline{v})$ – fixed source

$Q(\underline{r}, \underline{v}) = S(\underline{r}, \underline{v}) + \int \Psi(\underline{r}, \underline{v}')F(\underline{v}' \rightarrow \underline{v}, \underline{r})d\underline{v}'$ – fixed source + fission

$Q(\underline{r}, \underline{v}) = \frac{1}{k} \int \Psi(\underline{r}, \underline{v}')F(\underline{v}' \rightarrow \underline{v}, \underline{r})d\underline{v}'$ – eigenvalue

Solution procedure

Start with some assumptions

Static, homogeneous medium

Time independent

Markovian (important)

Neglect relativistic effects

Particles ‘fly’ in straight lines

Develop a Markov scheme with superposition

$$p \equiv (\underline{r}, \underline{v}) \tag{13}$$

$$R(p' \to p) \equiv C(\underline{v}' \to \underline{v}, \underline{r}') \cdot T(\underline{r}' \to \underline{r}, \underline{v}) \tag{14}$$

$$\Psi(p) = \sum_{k=0}^{\infty} \Psi_k(p) \tag{15}$$

$$\Psi_0(p) = \int Q(r', v) T(r' \to r, v) dr' \tag{16}$$

$$\Psi_k(p) = \int \Psi_{k-1}(p') \cdot R(p' \to p) dp' \tag{17}$$

Note that collision (k) then only depends on $(k-1)$

That's the trick of Markov

Histories

Then, develop several histories

$$\Psi_k(p) = \int \Psi_{k-1}(p') \cdot R(p' \rightarrow p) dp' \tag{18}$$

$$\Psi_k(p) = \int \Psi_0(p_0) \cdot R(p_0 \rightarrow p_1) R(p_1 \rightarrow p_2) \cdots R(p_{k-1} \rightarrow p_k) dp_0 \cdots dp_{k-1} \tag{19}$$

'Tally' the occurrences for each collision of each history

Generate $(p_0, p_1, p_2, ...)$ by randomly sampling from pdfs for source and transition

Histories are a sequence of these states

Simulate particle history from birth to death

Random walk for a single particle

Collisions are modeled by physics equations and cross sections

Model free-flight between collisions using computational geometry

Tally the occurrences of events in each region

Save any secondary particles, analyze them later

Then do tons of histories

History = original particle + progeny

Geometry then is extremely important and can be convoluted

Skipping ahead

Random numbers

Random number generation (26–49)

Why is this important?

Linear congruential random number generators

Lots of history of use

$$s_{k+1} = [g \cdot s_k + c] mod(2^N) \tag{20}$$

Don't want to repeat numbers (50–81)

MCNP uses $g = 5^{19},\ N = 63$ and adjustable

Random sampling (82–117)

Probability and statistics

Key to Monte Carlo

Gives several routines for common distributions

MCNP

What it does

MCNP is a very useful code to do all sorts of particle analysis

Calculating critical core size for all different types of reactors

Nearly anything statistically physics based

Skill in MCNP will help with other coding adventures

Good for a basis in nuclear engineering research and other courses

More interested in this class getting to use it for basic problems than higher level techniques

Particle tracking

What happens to a particle?

Born in the source or emerges from a collision

Sample flight distance using total cross sections; i.e., probabilities

Determine which isotope the interaction is with using cross sections

Determine which interaction type for that isotope using cross sections

Determine the energy and direction of the exiting particle with random sampling

Determine if secondary particles were produced

Biasing + weight adjustments

Tallies of quantities of interest

How the flight distance is sampled

Particle at (x_0, y_0, z_0) traveling with direction (u, v, w)

Sample free flight distance

pdf, cdf for flight distance s –

$$f(s) = \Sigma_T e^{-\Sigma_T s} \tag{21}$$

$$F(s) = 1 - e^{-\Sigma_T s} \tag{22}$$

Find s by random sampling –

$$s = -\frac{ln(1-\xi)}{\Sigma_T} \tag{23}$$

Tally

What does mcnp actually tally? (206)

During a history, tally the events of interest

Upon completing a history, accumulate total scores and squares

After completing all histories, compute mean scores and standard deviations

Average flux in a cell (see paper)

Current or flux through a cell wall

Criticality (KCODE)

Detectors

RTGs

Input decks

Making the input deck is not that hard

Geometry
Cylinders are the easiest to do
Visual editor/plotter

Materials
See compendium

Source
`SDEF` for sources
`KCODE` for criticality

Scores & Tallies
Designate flux, detector, etc.

Variance reduction if needed after

MCNP is FORTRAN so it is FICKLE

Length – cm

Energy – MeV

Mass density – g/cc

Atom density – $atoms/barn - cm$

Do NOT use tabs

Get a real text editor

Cards can only span 7th column to 80th column but you can wrap

The input deck structure is very specific

```
One Line Problem Title Card
Cell cards
.
.
.
BLANK LINE
Surface cards
.
.
.
BLANK LINE
Data cards
.
.
.
```

Geometry

Geometry is fickle and an acquired skill (126)

Define surfaces

Define cells using surfaces & operators (intersection, union, complement)

Can also group cells together into a universe, repeat that universe in a lattice arrangement, and embed that universe inside another cell

Assign materials to cells

Assign other properties to cells (e.g., importance weights)

Define tallies using cell or surface numbers

Neutron flux example

Surfaces

Surfaces can be lots of different geometries

$$F(x, y, z) = ax^2 + by^2 + cz^2 + dxy + eyz + fxz + gx + hy + jz + k = 0 \quad (24)$$

A surface is infinite and closed

But you just specify constants on the cards (nothing to actually solve)

$$x^2 + y^2 - R^2 = 0 \quad (25)$$

```
10 CZ 20
```

Just a vertical cylinder (infinite) with 20 cm radius

Keep access to the manual to look up surface cards – see 128

Sense

Here's where things get weird

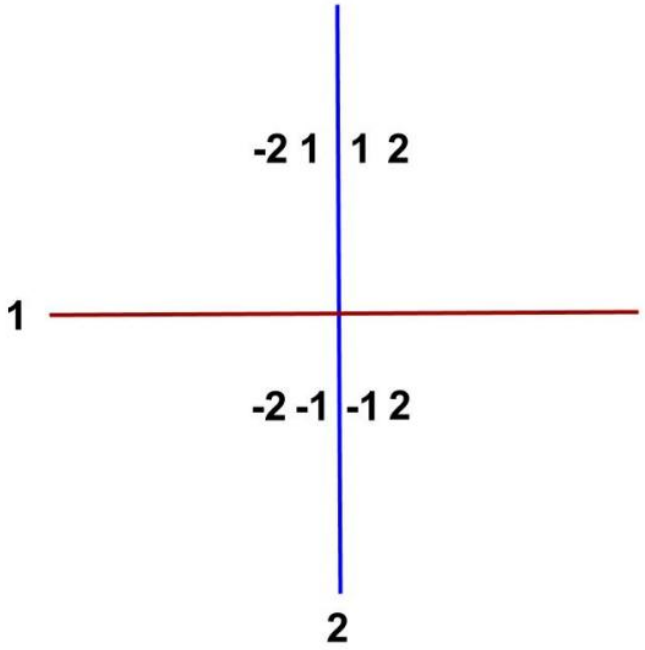
'Sense' is how a point is defined relative to a surface

For a given point in space (x, y, z), the sense is where it is wrt surface

$$F(x, y, z) < 0 \textit{ inside} \tag{26}$$

$$F(x, y, z) > 0 \textit{ outside} \tag{27}$$

$$F(x, y, z) = 0 \textit{ on} \tag{28}$$


-2 1
1 2
1
-2 -1
-1 2
2

Cells

Surfaces define cells

Each has a distinct volume because they are defined by density (134-5)

Intersection, union, of defined surfaces; NOT cell

```
100 1 -2.03 -10 11 -12 $metal/salt-filled cathode basket
101 2 -3.58 (10:-11) -12 -13 14 $MgO cathode basket
102 3 -0.00178 (12:13:-14) -15 16 -17 $Ar content in equipment
103 4 -7.92 (15:-16:17) -18 -19 20 $steel equipment
104 3 -0.00178 (18:19:-20) -50 51 -52 53 54 -55 $rest of SE hot cell
110 3 -0.00178 -52 53 54 -55 -61 70 $SW hot cell
111 3 -0.00178 54 -55 -61 62 70 -72 $NW hot cell
112 3 -0.00178 -50 51 54 -55 62 -72 $NE hot cell
```

Finite cylinder with equal radius and height

```
10 CZ 22.23 $metal/cathode basket radius
11 PZ 100 $metal/cathode basket above the floor
12 PZ 122.23 $metal/cathode basket height+floor height
```

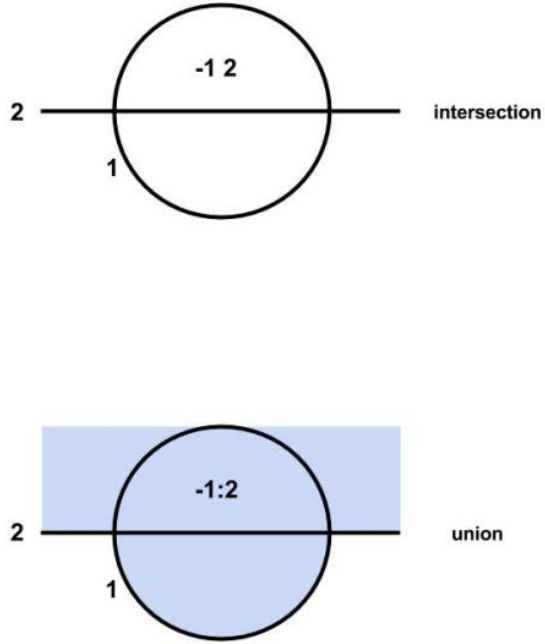
100 cm above the floor

Use the union operator to assemble larger cells or get around harder shapes

```
10 CZ 22.23 $metal/cathode basket radius
11 PZ 100 $metal/cathode basket above the floor
12 PZ 122.23 $metal/cathode basket height+floor height
13 CZ 27.23 $5 cm thick MgO cathode basket
14 PZ 95 $height above the floor/cathode basket bottom

100 (10:-11) -12 -13 14 $MgO cathode basket
```

Use of `NOT` cell is actually cleaner


-1 2
2
1
intersection
-1:2
2
1
union

Universes

Universes are useful for repeating units

Or ‘nested geometries’ at multiple levels

For a reactor core

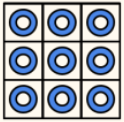
Remember ‘unit cells’ from reactor theory?

Group a collection of cells into a universe

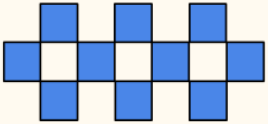
Use VisEd for help on geometries

unit cell

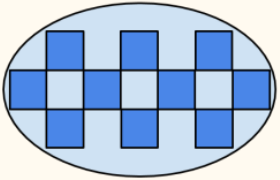
lattice of cells for fuel assembly



fill the lattice



lattice of lattices (core)



real world geometry

Other tips

Other tips

Someone already has made the same error as you – look it up

Do not have coinciding surfaces

Check the model space with VisEd or the plotter

It will show where there are problems

Use the compendium in the OER to get weight fractions of common materials

Then look up the cross section libraries (.70c, etc.) in Appendix G

Additional information

Additional information in the slides

HTGR modeling (148-165)

TRISO pebbles

Example of embedded universes

Collision physics (166-205)

How particles travel

Interactions

Determine which isotope the interaction is with

Determine which interaction type for that isotope

Determine the energy & direction of the exiting particle

Determine if secondary particles were produced

Sample flight distance

Biasing + weight adjustments

Tallies of quantities of interest based on pdfs

MCNP quality control

Mean & variance

Really we only need mean and variance for most problems

For random variable x of function $r(x)$ with pdf $f(x)$

$$\mu \equiv \int r(x)f(x)dx \tag{29}$$

$$\sigma^2 \equiv \int r^2(x)f(x)dx - \mu^2 \tag{30}$$

Monte Carlo randomly samples to get moments

$$\bar{r} \approx \frac{1}{N} \sum_{j=1}^{N} r(x_j) \tag{31}$$

$$\sigma_{\bar{r}}^2 \approx \frac{1}{N-1} \left(\frac{1}{N} \sum_{j=1}^{N} r(x_j) - \bar{r}^2 \right) \tag{32}$$

This is based on the law of large numbers

Sample average will approach the true expected value with increasing N

Central limit theorem states it will take on a normal distribution (210)

So the tally is the most probable result (basically)

Statistical checks

Statistical checks are provided for each tally

Relative error (R) = standard deviation/mean

R should decrease with increasing sqrt N smoothly

FOM is a measure of relative error and computer time

FOM should be roughly constant

Variance of Variance (VOV) is relative variance of R

Essentially verifies the central limit theorem

Decrease smoothly with N

Explained in depth in the manual

Statistical checks are to avoid undersampling

More about statistical checks (2-132)

Mean

(1) A nonmonotonic behavior (no up or down trend) in the estimated mean as a function of the number histories N for the last half of the problem

R

(2) An acceptable magnitude of the estimated R of the estimated mean (less than 0.05 for a point detector tally or less than 0.10 for a non-point detector tally)

(3) A monotonically decreasing R as a function of the number histories N for the last half of the problem

(4) $N^{0.5}$ decrease in the R as a function of N for the last half of the problem

VOV

(5) The magnitude of the estimated VOV should be less than 0.10 for all types of tallies

(6) A monotonically decreasing VOV as a function of N for the last half of the problem

(7) A $\frac{1}{N}$ decrease in the VOV as a function of N for the last half of the problem

FOM

(8) A statistically constant value of FOM as a function of N for the last half of the problem

(9) A nonmonotonic behavior in FOM as a function of N for the last half of the problem

$FOM \equiv \frac{1}{R^2 T}$

f(x)

(10) The SLOPE (see page 2–127 Pareto fit) of the 25 to 201 largest positive (negative with a negative DBCN(16) entry) history scores x should be greater than 3.0 so that the second moment will exist if the SLOPE is extrapolated to infinity

Criticality

For criticality, a little different (247)

Assume an initial k ($k = 1$, just use default `KCODE`; `KSRC = x y z` for a point in the middle of the source)

Fission sites are sampled from the initial source distribution

Follow a 'batch' of histories to estimate k (collectively called cycles)

When they converge, discard the tallies

Then start the new iterations until variances decrease

The idea is to run from 1 to N+D cycles (D = discard number) to converge the initial guess

Then run for N cycles to get a confident k

The gist is when obtaining k, the value is statistically robust

Typically, just start with the MCNP defaults and that should be good enough

Having some reasonable idea of the theory can help with higher level problems

Variance reduction

MCNP also comes with many many variance reduction techniques (310)

Increase N for the first move (or cycles)

Modified sampling

Modify the pdfs for physics interactions to favor events/tallies of interest

Population control

Use splitting/roulette to increase particles in certain geometric regions

Control number of samples taken in specified regions of the phase space

Weight Window Generator (WWG) and importance functions

Truncation

Kill particles in uninteresting parts of problem

May be necessary in order to sample rare events

More samples (with less weight each) $\rightarrow$ smaller variance in tallies

Deterministic methods

Replace portions of a particle random walk by the expected results obtained from a deterministic calculation

Not a huge problem with basic criticality problems

There are cards to do this so you don't have to actually code

$$\mu = \int r(x)\frac{f(x)}{g(x)}g(x)dx \tag{33}$$

$$\sigma^2 = [r(x)\frac{f(x)}{g(x)}]^2 g^2(x)dx - \mu^2 \tag{34}$$

Choose $g(x)$ to reduce variance; e.g., R, VOV

Favor directions more important to tallies

Change weights, importance, bias pdfs, and so much more

It isn't a science and comes with experience

Different problems have different approaches

(1) Time and energy cutoffs
(2) Geometry splitting & roulette
(3) WWGs
(4) Exponential transform
(5) Forced collisions
(6) Energy splitting & roulette
(7) Time splitting & roulette
(8) Point and ring detectors
(9) DXTRAN
(10) Implicit capture
(11) Weight cutoff
(12) General source biasing
(13) Secondary particle biasing
(14) Bremsstrahlung energy biasing

Keep the manual handy

Jezebel example

Deck on GitHub

Set up examples with radii

Criticality guide