NE 155/255
Numerical Simulations in Radiation Transport
Geometry, Collisions, and Scoring

Kelly L. Rowland

November 13, 2019

Major Components of MC Algorithm

- **PDFs**: the physical/mathematical system must be described by a set of pdfs.
- **Random number generator**: a source of random #s uniformly distributed on the unit interval.
- **Sampling rule**: prescription for sampling the pdf (given having random #s)
- **Scoring**: *the outcomes must be accumulated/tallied for quantities of interest*
- **Error estimation**: an estimate of the statistical error (variance) of the solution
- **Variance Reduction**: methods for reducing the variance and computation time simultaneously
- **Parallelization**: efficient use of computers

Outline

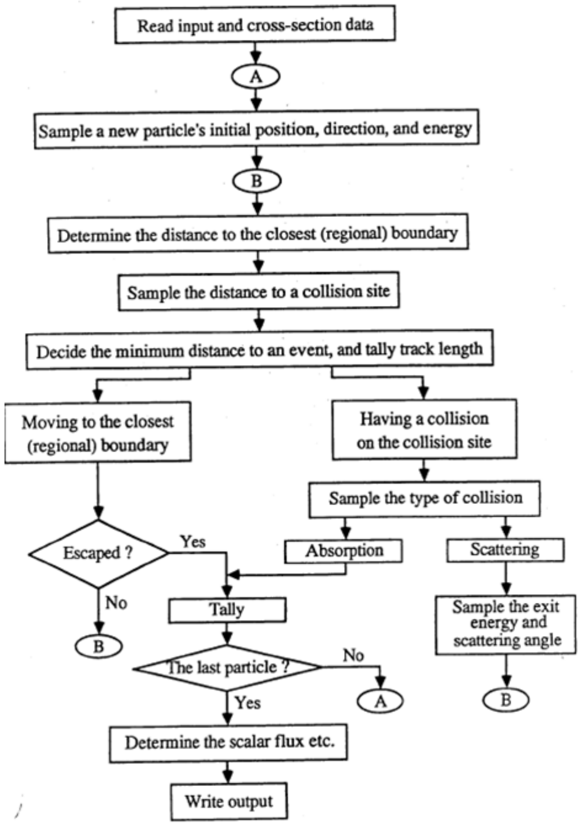
1. Determining next event location
   - Sampling flight path
   - Distance to boundary
   - Next event selection
2. Collision Physics
   - Sampling target nuclide
   - Sampling reaction type
   - Sampling exit direction
   - Sampling exiting particles
3. Scoring

Notes derived from Rachel Slaybaugh, Jasmina Vujic, and Paul Wilson.

Learning Objectives

1. Understand basic tracking of particles through a geometry
   - Understand the steps necessary for tracking particles
   - Understand the use of mean free path
   - Sample the distance to the next physics event
   - Determine next event
2. Understand what sampling needs to happen after a collision
3. Understand how to translate interactions into a score

Monte Carlo for Transport



Possible Futures for a Particle

After we've gotten to Circle B, we have a neutral particle:

- At point (x_p, y_p, z_p)
- Moving in direction (u, v, w)
- With energy E

What are possible next events?

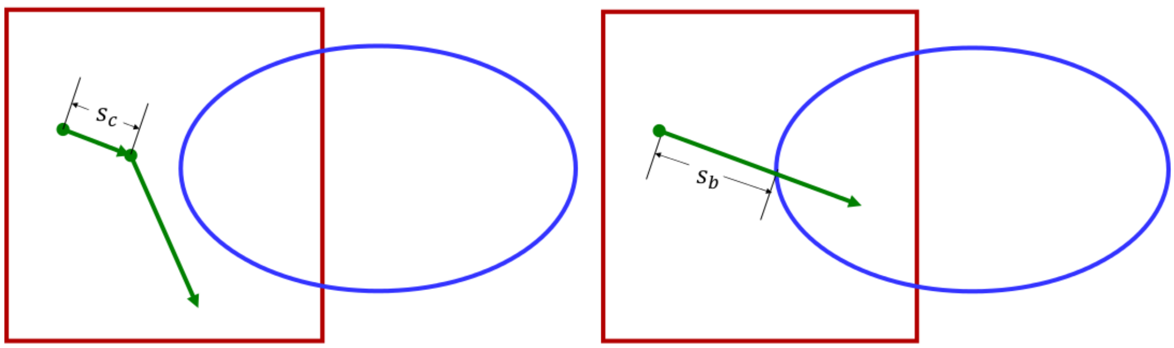


Figure 1: Collisionfig/Surface Crossing

Sampling Distance to Collision

Collisions are probabilistic

- Note that Σ_t, the total macroscopic cross section, will be a function of space if we have multiple materials
- Along a particular path, the *probability of a collision at distance s* from the start:

$$p_c(s)ds = \Sigma_t(s)e^{-\Sigma_t(s)s}ds$$
$$P_c(s) = \int_0^s \Sigma_t(s)e^{-\Sigma_t(s)s'}ds' = -e^{-\Sigma_t(s)s'}|_0^s = 1 - e^{-\Sigma_t(s)s}$$

- The cross section, $\Sigma_t(s)$, is piecewise constant, <u>but changing</u>

Sampling Distance to Collision

- Variable transformation: measure distance in units of *mean free path*:

$$n = \Sigma_t(s)s\,, \quad dn = \Sigma_t(s)ds$$

- We'll start with the PDF and integrate to get the CDF

$$p_c(n)dn = e^{-n}dn$$
$$P_c(n) = \int_0^n e^{-n'}dn' = -e^{-n'}|_0^n = 1 - e^{-n}$$

- Importantly, this is now <u>independent of the material</u>

Sampling Distance to Collision

Randomly sample to determine number of mean free paths until next collision, n_c

- $g(n_c)dn_c = e^{-n_c}dn_c$
- $G(n_c)dn_c = 1 - e^{-n_c}$
- Directly invert to get: $\boxed{n_c = -\ln(1-\xi)}$
  [note $(1-\xi)$ is equivalent to ξ]
- In the absence of material boundaries ($\Sigma_t \neq f(s)$), the distance to a collision, s_c, is

$$s_c = \frac{n_c}{\Sigma_t}$$

Calculating Distance to Boundary

- Usually have *more than one material*
- Distance to boundary is deterministic
- Algebra to determine distance between point and surface, s_b
- Convert it to units of mean free path for the current cell's material,

$$n_b = s_b \Sigma_t$$

Geometry Representations

- Combinatorial Surfaces
  - Define surfaces
  - Boolean operations combine surfaces to create cells
- Combinatorial Solids
  - Choose solid objects
  - Boolean operations combine objects to create regions
- B-Rep (Vertex-Edge)
  - Each object is a single set of vertices and edges connecting them

We're skipping how to find s_b, just know that we can find it using the internal geometry representation

Option A: Collision

$n_b > n_c$:

- Boundary is further away than collision
- Collision occurs
- Using physics models and/or cross-sections
  - Sample target nuclide
  - Sample reaction type
  - Sample new direction
  - Sample new energy
  - Sample exiting particles
- Some of these may depend on one another
- Repeat
  - Sample new n_c following collision
  - Calculate new n_b in new direction

Option B: Cell Boundary

$n_b < n_c$:

- Boundary is closer than collision
- Boundary crossing occurs

- Move particle along ray
  - Update $n_c = n_c - n_b$
- **DO NOT SAMPLE** for new n_c

- Calculate new n_b in new cell
  - New set of boundaries
  - New value of Σ_t

So You Had a Collision?

- Sample **target nuclide** for a mixture with J nuclides

$$\Sigma_t = \sum_{j=1}^{J} N_j \sigma_{t,j}$$

- *Discrete PDF* to determine which nuclide is hit

$$p_j = \frac{\Sigma_{t,j}}{\Sigma_t}$$

- Sample **reaction type** for an nuclide with R types of reactions

$$\Sigma_{t,j} = \sum_{r=1}^{R} \Sigma_{r,j}$$

- *Discrete PDF* to determine which reaction occurs

$$p_r = \frac{\Sigma_{r,j}}{\Sigma_{t,j}}$$

Outcome of Reaction

- Particle maybe absorbed
- Particle may continue its history in a *different direction* and/or with a *different energy*
- Energy-angle distributions are tabulated in different formats
  - Scattering laws have analytic forms with parameters in data tables (Direct inversion or rejection sampling)
  - Tabulated data that describes a piecewise analytic interpolation (Hybrid sampling; we skipped this)

Using a Scattering Angle

Scattering angles are defined relative to the original direction (considered as the z-axis)

- Polar angle, θ, determined by sampling from data
- Azimuthal angle, ϕ, determined by sampling isotropically
- The new direction is $\big(\sin(\theta)\cos(\phi), \sin(\theta)\sin(\phi), \cos(\theta)\big)$

$$= \big(\sqrt{1-\mu^2}\cos(\phi), \sqrt{1-\mu^2}\sin(\phi), \mu\big)$$

Summary of Part I

We've developed a general sense of using MC for neutron transport

- Basic Algorithm
- We can determine if particles have collisions or cross boundaries
- *After a collisions* we need to determine many things associated with the collisions (target, reaction, direction, energy)
- Repeat analysis for collisions/crossing until particle **terminates**