NENG 685, Fall 17
Introduction to Monte- Carlo
October 30, 2017

Learning Objectives

1. Be able to define what a Monte-Carlo (MC) simulation is
2. Be able to justify the choice of MC for radiation transport
3. Be able to identify and explain the major components of MC radiation transport methods
4. Provide examples of probabilistic representations of physics
5. Distinguish between a PDF and CDF
6. Distinguish between a discrete PDF (CDF) and continuous PDF (CDF)
7. Describe the goal of random sampling
8. Identify and implement the best random sampling technique for a given distribution

What is Monte Carlo?

Monte Carlo methods employ the use of random processes to determine a statistically-expected solution to a problem. These random processes can fulfill two roles:

- Statistical approximation to mathematical equations
- Statistical approximations to physical processes

The basic concept for a MC simulation is to construct a random process for a problem and carry out a numerical simulation by N-fold sampling from a random number sequence. These processes have been used extensively for hundreds of years to solve complicated problems of the time:

- Comte du Buffon (1777): needle tossing experiment to calculate π
- Laplace (1786): random points in a rectangle to calculate π
- Lord Kelvin (1901): used random sampling to aid in evaluating time integrals associated with kinetic theory of gases
- Fermi (1930): was among the first to use random sampling methods to study neutron moderation while still in Rome, Italy

- 1947: Fermi, von Neumann, Frankel, Metropolis, Ulam, and others developed computer-oriented Monte Carlo method at Los Alamos to trace neutrons through fissionable materials; coined the term Monte Carlo
- Berger (1963): first complete coupled electron-photon transport code

MC is applicable to applications that are mathematically equivalent to *integration over many dimensions.* This can be useful when

- Analytic integration is impossible
- Deterministic numerical integration is slow and/or requires error prone approximations.

The relatively straightforward approach of MC methods, combined with the fact that they easily simulation non-deterministic processes has led them to be used in a wide variety of fields:

- High energy physics: many nucleon interactions
- Process engineering: Combine uncertainties in many variables
- Financial sector: Prices and rates of return for many objects (simulate many possible futures given trends and uncertainty)
- Risk analysis: Many individual probabilistic contributes to risk

However, MC methods often require significant computer time to general statistically significant results. This can be overcome with parallel computing since MC methods are "embarrassingly parralelizable". Additionally, variance reduction (VR) methods have been developed that can be used to improve statistics with fewer particles. Finally, hybrid methods have been developed that use deterministic methods to accelerate MC calculations. We will explore all of this over the next couple of lessons.

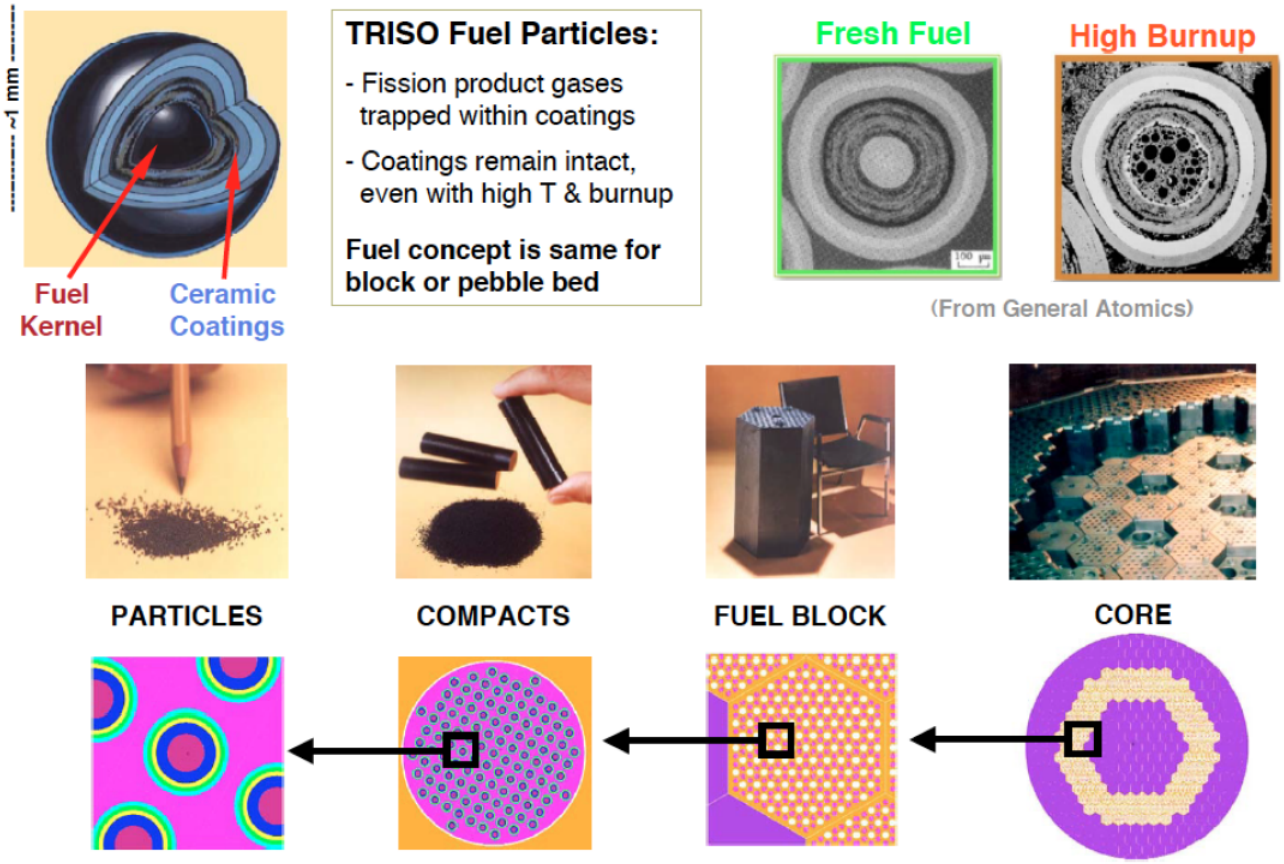
For now, let us explore the earliest use of MC to evaluate π using the iPython notebook.

What is MC Radiation Transport?

MC transport works by simulating many independent particles in a system. For radiation transport, this works well as most of the physical processes that affect source generation and transport can be described as probabilistic events. The basic formulation of a MC transport method is to

- Treat each physical process as a probabilistic process
- Randomly sample each process using an independent stream of random numbers
- Follow each particle from birth until it no longer matters
- Accumulate the contributions of each particle to find the statistically-expected mean behavior and variance

Unlike deterministic approaches, MC methods can readily model very complex things such as pebble bed reactors. Why?



TRISO Fuel Particles:

- Fission product gases trapped within coatings
- Coatings remain intact, even with high T & burnup

Fuel concept is same for block or pebble bed

(From General Atomics)

Accurate & explicit modeling at multiple levels

Is it worth considering

- Is the independence assumption valid for all radiation transport?
- If not, where does the independence assumption cause problems?
- Are there radiation transport mechanisms that are deterministic instead of probabilistic?

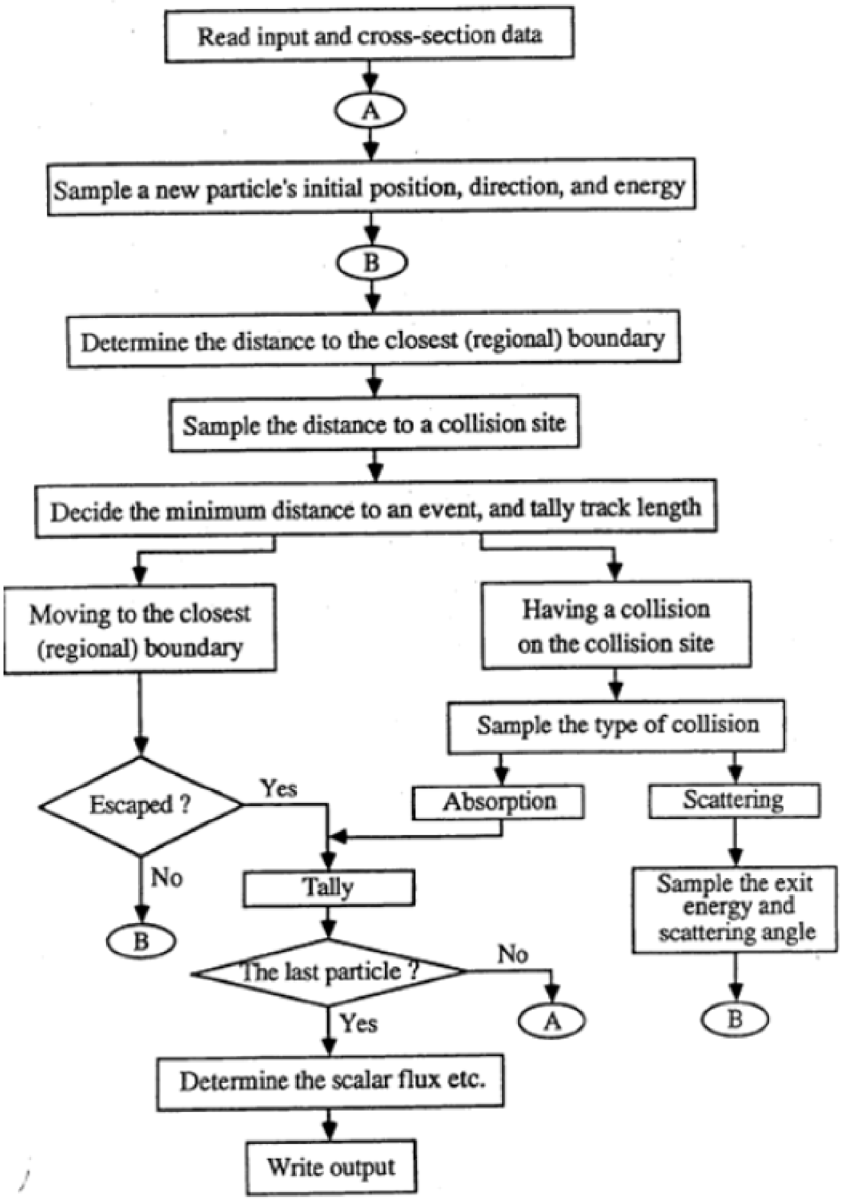
Major Components of a MC Algorithm

There are several common components of a MC algorithm that apply across any implementation considered:

- 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

For radiation transport, and example simple algorithm, using these components might be formulated as:


Read input and cross-section data
A
Sample a new particle's initial position, direction, and energy
B
Determine the distance to the closest (regional) boundary
Sample the distance to a collision site
Decide the minimum distance to an event, and tally track length
Moving to the closest (regional) boundary
Having a collision on the collision site
Sample the type of collision
Escaped ?
Yes
Absorption
Scattering
No
Tally
Sample the exit energy and scattering angle
B
The last particle ?
No
Yes
A
B
Determine the scalar flux etc.
Write output

General Purpose MC Codes

MC is the most common radiation transport method used across a wide range of applications. A non-inclusive list of common MC codes include:

- MCNP: developed at LANL, distributed via RSICC, `http://rsicc.ornl.gov`
- Geant4: developed by a large collaboration in the HEP community, `http://geant4.web.cern.ch/geant4/`
- EGSnrc: developed at NRC (Canada), `http://www.irs.inms.nrc.ca/EGSnrc/EGSnrc.html`
- SERPENT: Developed by Dr. Jaakko Leppanen, VTT, Finland, `http://montecarlo.vtt.fi/`
- Shift: developed at ORNL, distributed via RSICC, `http://rsicc.ornl.gov`
- Mercury: developed at LLNL, `https://wci.llnl.gov/simulation/computer-codes/mercury`

Physics as Probability

Various radiation related physical phenomena can be represented by probability distributions:

- Photon emission energy: Each possible energy has a different probability (intensity)
- Scattering cross-sections: Each possible scattering angle has a different probability as a function of the energy
- Transmission through a medium: Probability of reaching a particular position depends on the cross-section
- Radioactive decay: Each decay event is probabilistic based on probability of individual atoms to tunnel through energy barriers

To implement MC, we must quantify the underlying probability distributions for the outcome of each physical event. We can then estimate the statistical moments of these distributions to get our physical answers. Let's start by exploring probability density functions (PDFs) is more detail.

PDFs and CDFs

In the MC simulation, all variables, x, have a Probability Density Function (PDF), $p(x)$, with the following characteristics:

Continuous:

$$
\begin{aligned}
p\{a \le x \le b\} &= \int_a^b p(x)dx \\
p(x) &\ge 0 \\
\int_{-\infty}^{\infty} p(x)dx &= 1
\end{aligned}
$$

Discrete:

$$
\begin{aligned}
p(x = x_k) &= p_k \equiv p(x_k) \\
k &= 1, \ldots, N \\
p_k &\ge 0 \\
\sum_{k=1}^{N} &= 1
\end{aligned}
$$

The PDF represents the *collective behavior* of the system. All PDFs, $p(x)$, have an associated Cumulative Distribution Function (CDF), $P(x)$, with the following properties:

Continuous:

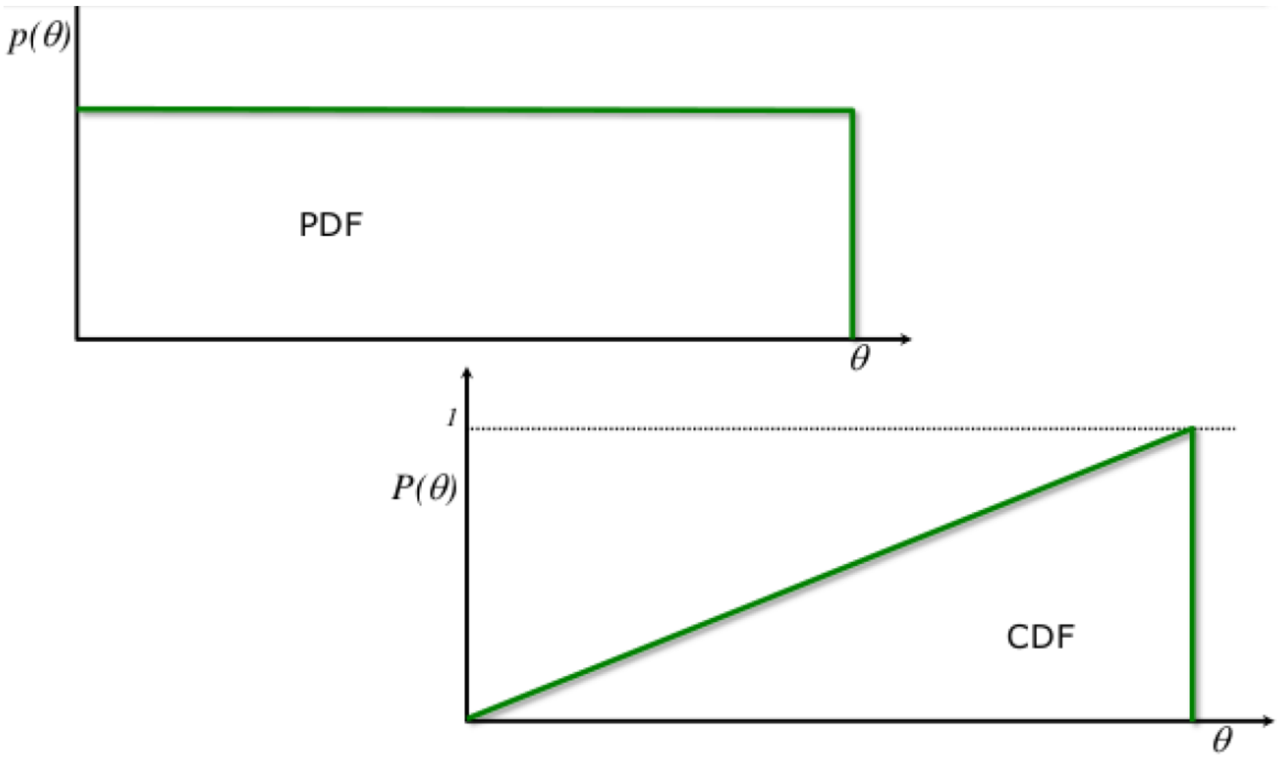
$$p\left\{x' \le x\right\} = P(x) = \int_{-\infty}^{x} p(x')dx'$$

$$
\begin{aligned}
P(-\infty) = 0\,, &\quad P(\infty) = 1 \\
0 \le P(x) &\le 1 \\
\frac{dP(x)}{dx} &\ge 0
\end{aligned}
$$

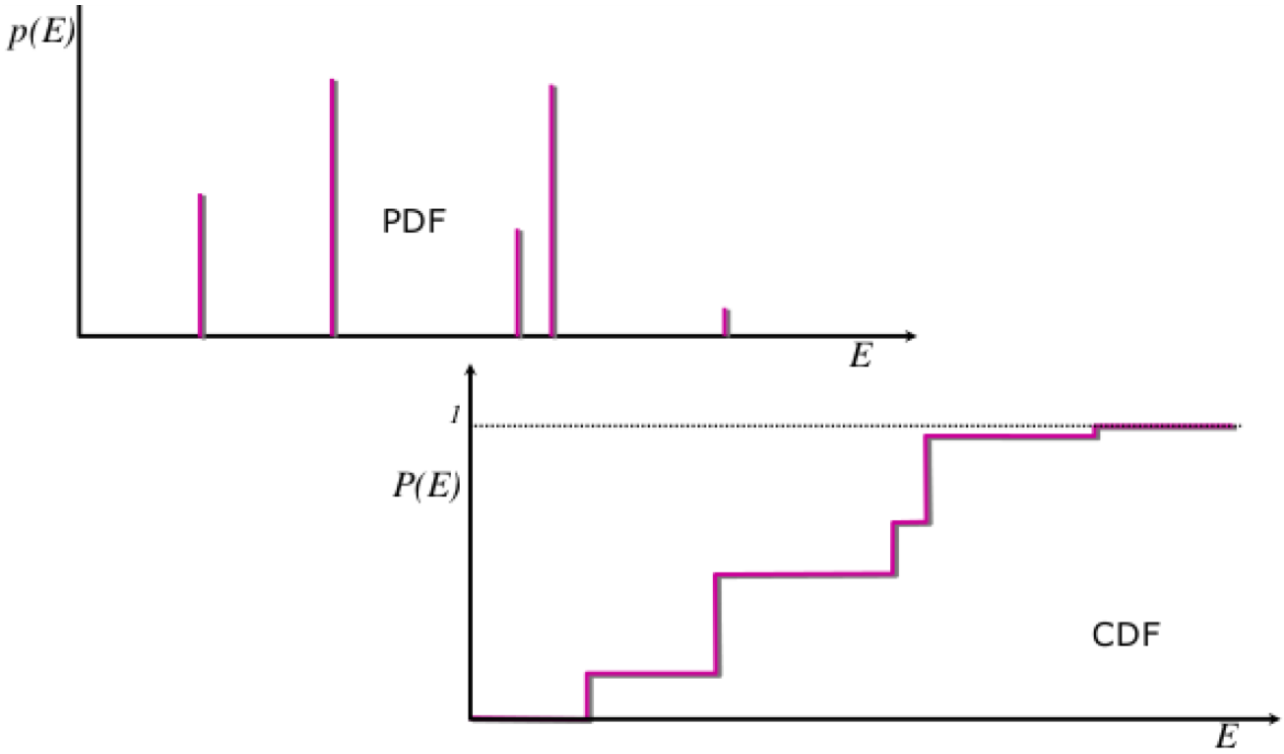
Discrete:

$$
\begin{aligned}
pP\left\{x' \le x\right\} = P_k \equiv P(x_k) &= \sum_{j=1}^{k} p_j \\
k &= 1, \ldots, N \\
P_0 = 0\,, &\quad P_N = 1 \\
0 \le P_k &\le 1 \\
P_k &\ge P_{k-1}
\end{aligned}
$$

To demonstrate, consider a uniform probability distribution:



Or a discrete probability distribution:



Sampling Techniques

Random sampling uses uniformly distributed random variables to choose a value for a variable according to its PDF. The implementations can vary based on the form of the data and the physics process being modeled. Some sampling techniques are

- *Basic* sampling techniques
  - Direct discrete sampling
  - Continuous direct sampling
  - Rejection sampling
- *Advanced* sampling techniques
  - Histogram
  - Piecewise linear
  - Alias sampling
  - Advanced continuous PDFs

The direct continuous sampling and discrete sampling will be explored in the Sampling iPython notebook.

Uniformly-Distributed Random Variable

The standard notation for a uniformly distributed random variable is:

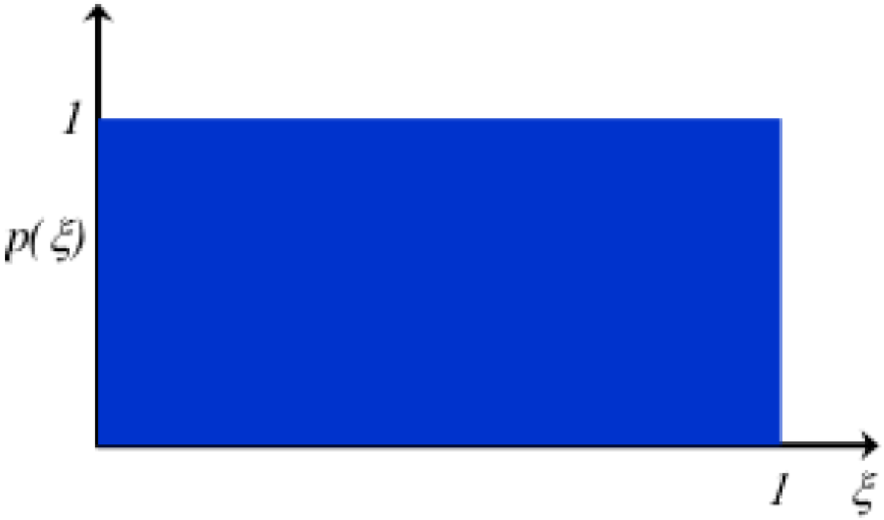
- Single random variable: ξ
- Pair of random variables: (ξ, η)

When we consider the PDF for random variables, it is given by

$$p(\xi) = \begin{cases} 1 & 0 \leq \xi < 1 \\ 0 & \text{otherwise} \end{cases}$$

Direct Discrete Sampling

To sample discrete distributions such as the one shown below, the general pseudocode is given by



- Convert PDF to CDF
- Generate ξ
- Determine k such that $P_{k-1} \leq \xi \leq P_k$
- return $x = x_k$

This is shown graphically using the PDF and CDF below.

This process requires a table search on P_k. A linear search requires $\mathcal{O}(N)$, and a binary search requires $\mathcal{O}(log_2 N)$ time.

Direct Continuous Sampling

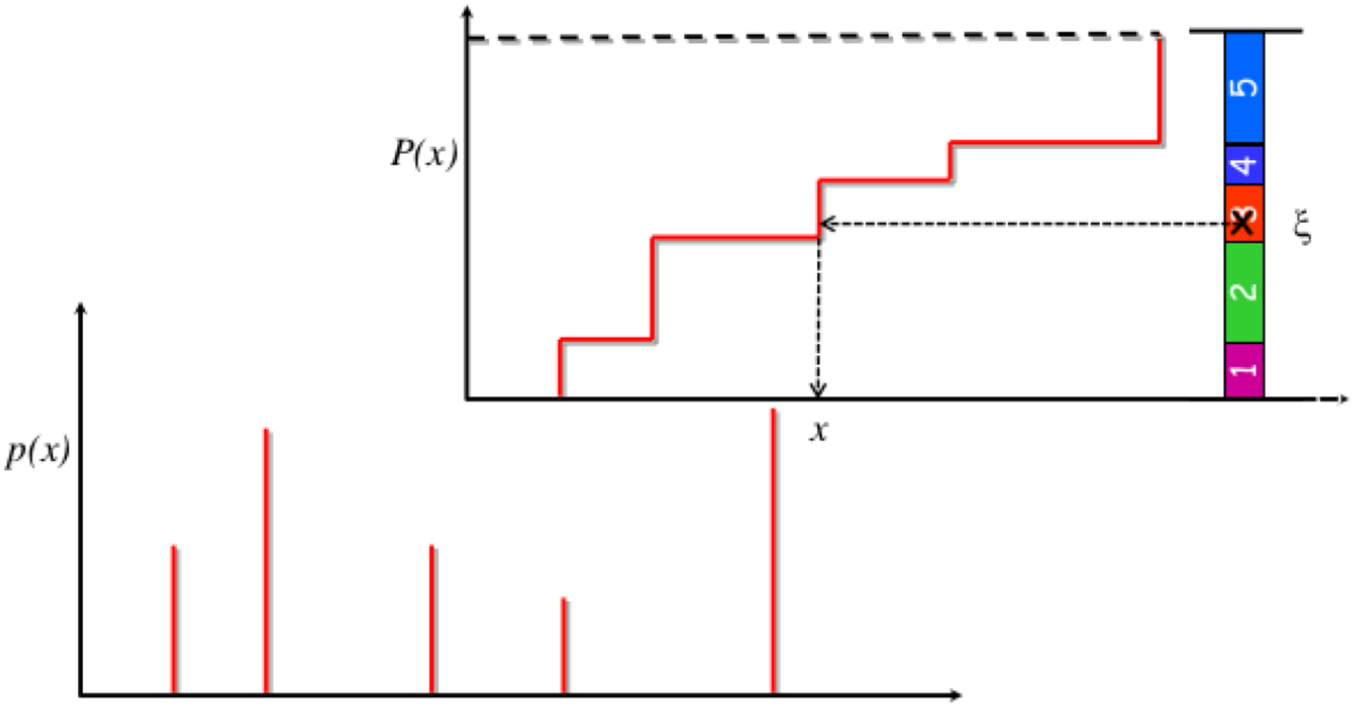
Direct sampling of a continuous distribution is only possible if the CDF can be inverted. If the CDF can be inverted the sampling procedure pseudocode is

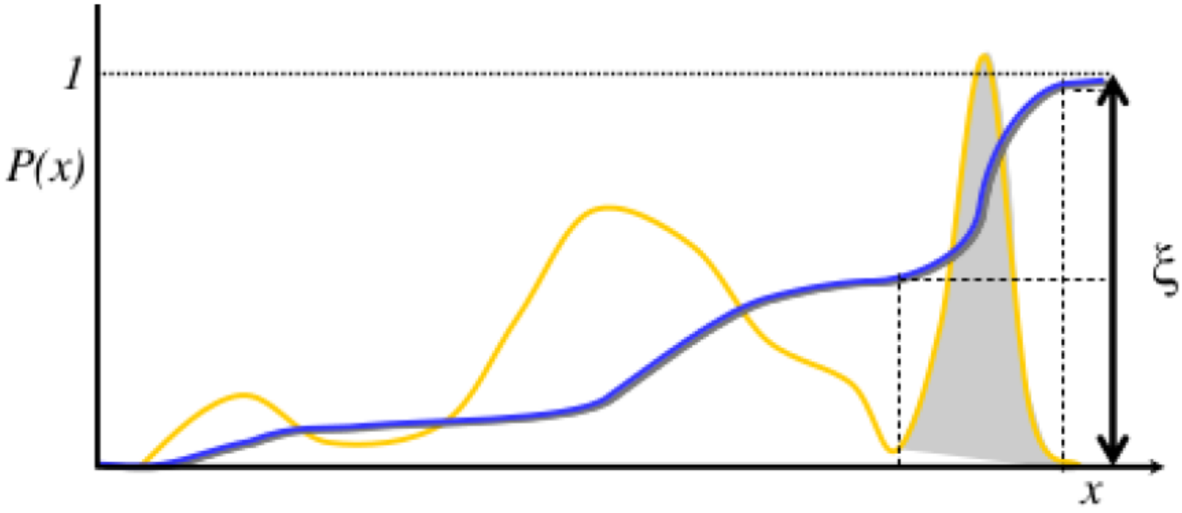
- Generate ξ
- Determine $x = P^{-1}(\xi)$

This is demonstrated graphically for two sampled values of ξ below where the yellow line is the PDF and the blue is the CDF.

Normalization

Random sampling depends on shape, so we need to define a method of normalization that can be used to scale the sampled solution. First, consider the exponential decay probability given by


P(x)
5
4
3
2
1
ξ
x
p(x)


1
P(x)
ξ
x

$$g(t)dt = e^{-\lambda t}dt, \;\; t > 0$$

The evaluation of the function for a given time, t, is given by

$$G(t) = \int_{-\infty}^{t} g(t')dt' = \int_{0}^{t} g(t')dt' = \left[-\frac{e^{-\lambda t'}}{\lambda}\right]_0^t = \frac{1}{\lambda}\big(1 - e^{-\lambda t}\big)$$

Evaluating at $t = \infty$,

$$G(\infty) = \frac{1}{\lambda}$$

The probability of a decay at time, t is given by

$$p(t) = \frac{g(t)}{G(\infty)} = \lambda g(t) = \lambda e^{-\lambda t}\,, \quad t > 0$$

The CDF can then be evaluated as

$$\begin{aligned} P(t) &= \int_{-\infty}^{t} p(t')dt' = \int_{0}^{t} \lambda p(t')dt' = \big[e^{-\lambda t'}\big]_0^t = 1 - e^{-\lambda t} \\ P(\infty) &= 1 \end{aligned}$$

This procedure can be followed for any PDF governed by a functional form.

Shifted Uniform Distribution

A shifted uniform distribution is given as a uniform distribution over some interval other than [0,1). Following the procedure in the normalization section, the sampling can be determined as

$$
\begin{aligned}
g(x)dx &= Cdx \quad a \le x < b \\
G(x) &= \int_{-\infty}^{x} g(x')dx' = C\int_{a}^{x} dx' = C\big[x'\big]_a^x = C\big(x-a\big) \\
G(\infty) &= G(b) = C(b-a)
\end{aligned}
$$

$$
\begin{aligned}
p(x) &= \frac{g(x)}{G(\infty)} = \frac{C}{C(b-a)} = \frac{1}{b-a} \quad a \le x < b \\
P(x) &= \int_{-\infty}^{x} p(x')dx' = \frac{1}{b-a}\int_{a}^{x} dx' = \frac{x-a}{b-a}
\end{aligned}
$$

$$
x = P^{-1}(\xi) = \xi\big(b-a\big) + a
$$

Simple Line, Slope=m

$$
\begin{aligned}
g(x)dx &= mx\,dx \qquad 0 \le x < 1 \\
G(x) &= \int_{-\infty}^{x} g(x')dx' = \int_{0}^{x} mx'dx' = \frac{m}{2}\big[x'^2\big]_0^x = \frac{m}{2}x^2 \\
G(\infty) &= G(1) = \frac{m}{2}
\end{aligned}
$$

$$
\begin{aligned}
p(x) &= \frac{mx}{\frac{m}{2}} = 2x \qquad 0 \le x < 1 \\
P(x) &= \int_{-\infty}^{x} p(x')dx' = \int_{0}^{x} 2x'dx' = \big[x'^2\big]_0^x = x^2
\end{aligned}
$$

$$
x = P^{-1}(\xi) = \sqrt{\xi} \qquad \text{Independent of } m
$$

Shifted Line, Slope=m

$$
\begin{aligned}
g(x)dx &= m(x-a)\,dx \qquad a \le x < b \\
G(x) &= \int_{-\infty}^{x} g(x')dx' = \int_{a}^{x} m(x'-a)dx' = \frac{m}{2}\left[(x'-a)^2\right]_0^x = \frac{m}{2}(x-a)^2 \\
G(\infty) &= G(1) = \frac{m}{2}(b-a)^2
\end{aligned}
$$

$$
\begin{aligned}
p(x) &= \frac{m(x-a)}{\frac{m}{2}(b-a)^2} = 2\frac{x-a}{(b-a)^2} \qquad a \le x < b \\
P(x) &= \int_{-\infty}^{x} p(x')dx' = \frac{1}{(b-a)^2}\int_{a}^{x} 2(x'-a)dx' = \frac{(x-a)^2}{(b-a)^2}
\end{aligned}
$$

$$
x = P^{-1}(\xi) = \sqrt{\xi}(b-a) + a \qquad \text{Independent of } m
$$

Rejection Sampling

Many CDFs cannot be inverted. For these, we can use a graphical approach similar to the approach used to evaluate π:

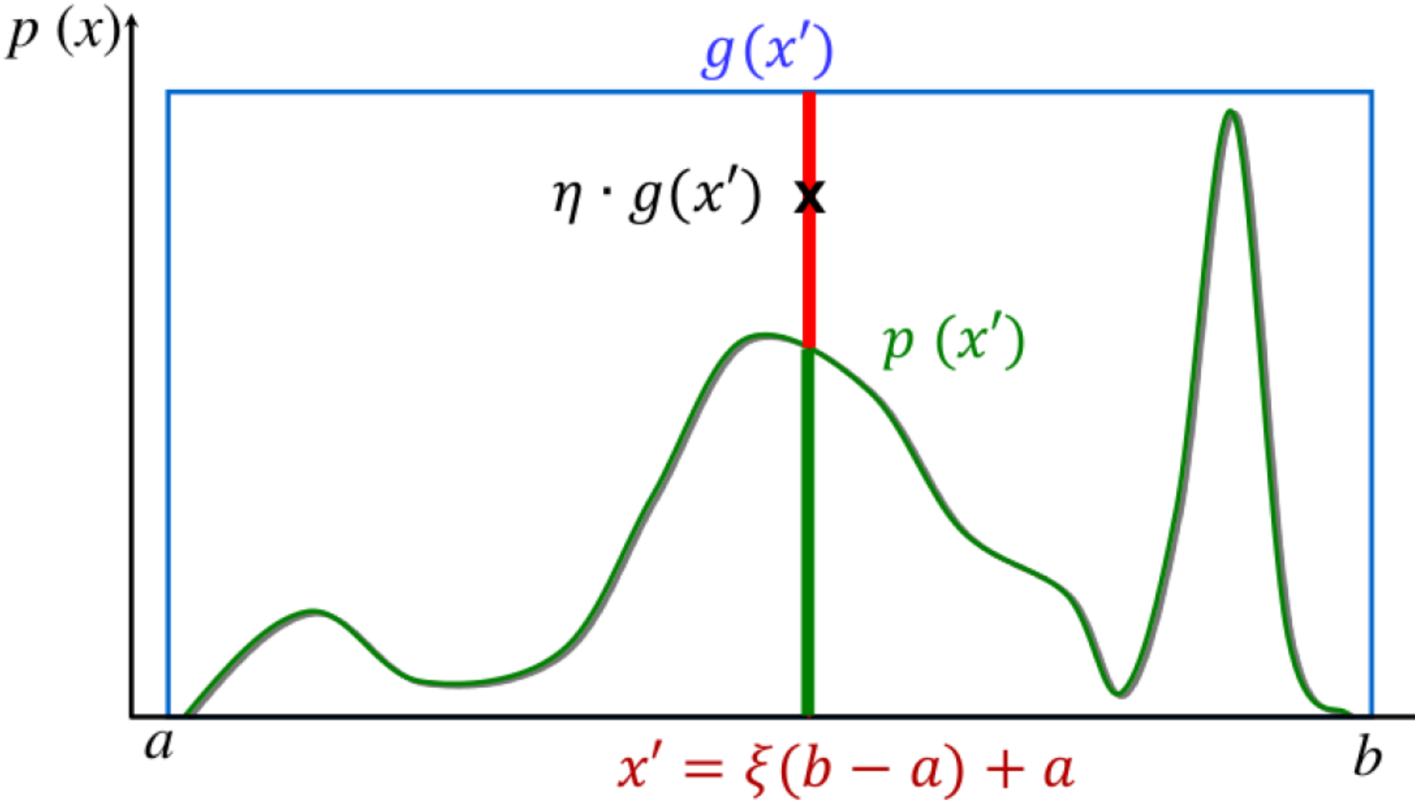
- Select a point in a 2-D domain
- Determine whether that point is above or below the PDF
- Keep those that are below
- Start over if above

To implement this, we must select a bounding function, $g(x)$, such that $g(x) \ge p(x)$ for all x and $g(x)$ is easy to sample. For example, the simplest choice is $g(x) = C$, but this may not always be the best choice. We then generate a pair of random variables, (xi, η) and

- $x' = G^{-1}(\xi)$
- If $\eta < \frac{p(x')}{g(x')}$, accept x'
- Else, reject x'

This method has the advantage that it is computationally simple and always works. However, it will be inefficient if the shapes of $g(x)$ and $p(x)$ are dissimilar:

$$
Efficiency = \frac{\int p(x)dx}{\int g(x)dx} \tag{1}
$$


$p\ (x)$
$g(x')$
$\eta \cdot g(x')$
$p\ (x')$
a
$x' = \xi(b-a)+a$
b