Monte Carlo Exercises, August 2019
Kenny Wood
kw25@st-andrews.ac.uk

Using the information given in the lectures on sampling from probability distributions, the rejection technique, and the basic structure of a Monte Carlo scattering code, write computer programs to compute the following.

1. Write a code to calculate a value of π using the rejection method.
2. Write a code to sample random optical depths and plot a histogram of the number of random optical depths versus τ.
3. Write a Monte Carlo radiation transfer code to simulate emission from an isotropic point source at the origin of a uniform density sphere. Assume the Monte Carlo photon packets are scattered isotropically on their random walks through the sphere. For the case where the scattering albedo is unity, compute as a function of the sphere's radial optical depth, τ_r, the average number of scatterings per Monte Carlo packet, $\langle N_{scatt} \rangle$. Plot $\langle N_{scatt} \rangle$ versus τ_r and compare this with the analytic approximation $\langle N_{scatt} \rangle \sim \tau_r + \tau_r^2/2$, derived in the lecture notes.
4. Implement the "next event estimator" (sometimes referred to as "peeling off") in your uniform density sphere code along with the equations for binning packets into image pixels. Make images of uniform density spheres of different optical depths illuminated by a central isotropic point source. Assume that the distance to the observer (image plane) is much greater than distances within the sphere. You should define a direction for viewing your simulation and at each scattering location compute the distance to the edge of the sphere along the direction of viewing. The optical depth is then easily computed (because the sphere is at a uniform density). You will use the scattering location and the viewing direction to bin the weighted Monte Carlo packets into image pixels.
5. For extra credit… can you parallelize your Monte Carlo scattering code?

The subroutine for generating pseudo random numbers, ran2.f from *Numerical Recipes*, is provided on the website. As a guide, the code for calculating π should be around a dozen lines (excluding ran2.f), and the Monte Carlo scattering code (again excluding ran2.f) can be written in less than sixty lines.

The questions below are more involved and designed for those who are interested in developing their own Monte Carlo codes.

The following questions explore sampling random locations on the surface of a sphere, emitting Monte Carlo photon packets from uniformly bright and limb darkened stars, and external illumination that may be used to simulate the illumination of clouds in the interstellar medium.

1. Write a subroutine that generates random locations that will uniformly cover the surface of a unit sphere. This can be done in a couple of ways: (a) by selecting random polar and azimuth angles as for isotropic emission in the notes or (b) generating random (x, y, z) locations in the range -1 to 1, rejecting those with $r^3 > 1$ and renormalizing those with $r^3 < 1$, where $r^2 = x^2 + y^2 + z^2$. Attempt both.
2. Write a subroutine that generates isotropic directions for emission of photon packets, but instead of using the formulae given in the notes, sample locations that will uniformly cover the surface of a unit sphere as in the previous question. Use the location on the unit sphere to determine the polar and azimuthal angles and the associated direction cosines.
3. Write a subroutine that emits photon packets from the surface of a star that will produce a stellar image that appears uniformly bright.
4. Write a subroutine that emits photon packets from the surface of a star that will produce a limb darkened star.
5. Write a code to produce images of the uniformly bright and limb darkened stars. You should bin the emitted photons packets in solid angle and location on an image plane. This will provide a test of the algorithms you developed in the previous questions.
6. Write a subroutine to bathe a spherical cloud with an isotropic radiation field.

The following questions are for calculations of mean intensity and radiation pressure within Monte Carlo codes.

1. Derive the “pathlength formula” that provides an estimator for the mean intensity.
2. For Monte Carlo radiation transfer within a discretized density grid, devise an algorithm for computing radiation force throughout the grid.