MASTER 1 OF PHYSICS
NUMERICAL METHODS AND SIMULATIONS
Strasbourg University - 2019-2020
Contacts : jonathan.chardin@astro.unistra.fr /
christian.boily@astro.unistra.fr

Monte Carlo Radiative Transfer in Astrophysics.

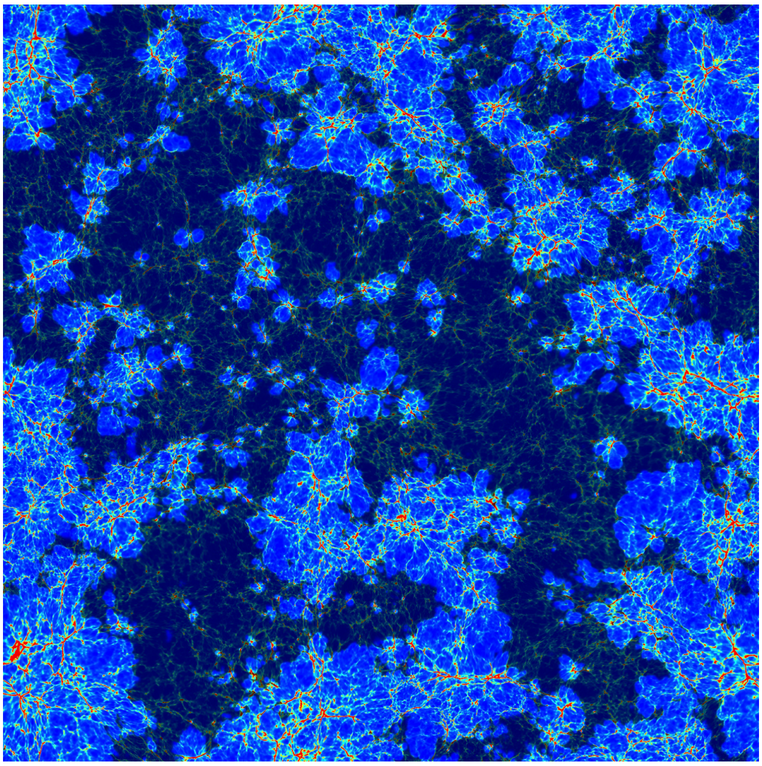

Figure 1 – The CODA radiative transfer simulation developed at the Strasbourg observatory.

Report

This semester you will have to write a report for your project. The report should not exceed 15 pages. The report should be in pdf format and should be sent before the first of May 2020 on the moodle directory. With your report you should also send all of your codes. The codes have to be written in C++, but the codes for making the plots can be done in Python. Both the codes and the report should be sent in a single archive with the following name : *surname_forename.tar*. Report typed using *Latex* would be very appreciated.

Plan

Here are listed the points that will be seen during this project

- Modelization of photons scattering of a point source in a Uniform medium
- Monte Carlo Approach of radiative transfer
- Simulating a photon detector

Material

For this project, you have access to the file *plot_ 3d.py* at the following adress to make fancy 3D plot.

Introduction

In this project, we are going to develop a code that simulate the radiative transfer of astrophysical sources. The theory of radiative transfer provides the link between the physical conditions in an astrophysical object and the observable radiation which it emits. Thus accurately modelling radiative transfer is often a necessary part of testing theoretical models by comparison with observations.

1 Photons transfer from a point source in an homogeneous medium

Introduction

To begin with we introduce the basic concepts and techniques of radiation transfer usingthe Monte Carlo method. We will deal with the propagation of "photon energy packets"and their interaction with matter within a medium. Radiation transfer can be seen as follows : a photon is emitted, it travels a distance, and something happens to it. Here we start with the assumption that a photon has been emitted and determine how far it will travel in a medium. Once this distance has been reached we discuss what can happen next. First of all we define a few basic terms that are central to any study of radiation transfer — photons, cross sections and optical depths.

1.1 Photon, Cross Sections, Optical Depths

In simulating the transfer of radiation we follow photon packets as they are scattered within a medium. The photons interact according to probabilistic interactions determined by the scattering cross sections of the particles within the medium. The cross section is a measure of probability that a specific process will take place in a collision of two particles. For example, the Rutherford cross section is a measure of probability that an alpha particle will be deflected by a given angle during a collision with an atomic nucleus. Cross section is typically denoted σ and is expressed in terms of the transverse area that the incident particle must hit in order for the given process to occur. A cross section thus has dimensions of area [cm2].
Consider now a homogeneous medium filled with scatterers of number density n, and cross section σ. We need to sample how far a photon travels before being scattered. The probability that a photon travels an optical depth τ without an interaction is $e^{-\tau}$. The probability of scattering prior to τ is $1 - e^{-\tau}$

$$\tau = \int_0^L n\sigma dl \tag{1}$$

1.2 Sampling optical depths from the Cumulative Distribution Function

To sample a quantity x_0 from a probability distribution function $P(x)$, which is normalized over all x, we use the fundamental principle which is

$$\xi = \int_a^{x_0} P(x)dx = \psi(x_0) \tag{2}$$

where ξ is a random number sampled uniformly from the range 0 to 1, a is the lower limit of the range over which x is defined, and ψ is the cumulative probability distribution function. From what we have seen in the previous section, we can therefore sample τ from the cumulative probability according to $\xi = 1 - \mathrm{e}^{-\tau}$, giving

$$\tau = -\ln(1 - \xi) \tag{3}$$

Having sampled a random optical depth in this manner we may then calculate the physical distance L that the photon travels from Eq. 1. We now have all the basics necessary to build a simple Monte Carlo radiation transfer code.

1.3 Computing the random direction for isotropic scattering

For isotropic emission the photon is re-emitted in any direction with equal probability, and independent of the direction of the incoming photon. Here we first pick two random numbers r_1 and r_2 (uniformely distributed between 0 and 1), from which we compute the angles

$$\varphi = 2\pi r_1 \tag{4}$$

and

$$\theta = \pi r_2 \tag{5}$$

and then work out the direction represented by the components of a vector of unity length :

$$vx = \sin\theta\cos\varphi \tag{6}$$

$$vy = \sin\theta\sin\varphi \tag{7}$$

$$vz = \cos\theta \tag{8}$$

1.4 How to compute the point of next scattering

Suppose the photon is emitted at some point (x, y, z) and flies into direction (vx, vy, vz). The next scattering should occur at some distance whose optical depth should be a random variable and distributed like the exponential function. We proceed this way :

- pick a random number r uniformely distributed between 0 and 1 and compute the expected random optical depth $\tau = -\ln(1 - r)$
- let the photon fly into its direction, until the optical depth along this path is equal to τ . Let's do it this way :
- Since the density n could change in space according to whatever you had prescribed, we have to compute the optical depth integral numerically, by stepping forward along the flight path.
- a first estimate of the integration step could be something like $\Delta l = \tau/(n\sigma)/100$
- we now follow the flight path, compute the optical depth at each new position by a simple trapezoidal rule : $\tau_s = \tau_s + [(n\sigma)_{i+1} + (n\sigma)_i] * \Delta l/2$
- we do this as long as τ_s is smaller than our target value τ
- When we finally reach $\tau_s > \tau$, we might have already overshot our target value. In order to avoid inaccurate behaviour of our simulation, it would be good to do an interpolation for the target point, using the data from that last interval. A linear interpolation would be good enough.

In Fig. 2, you have an example of photon scattering for 10 photons.

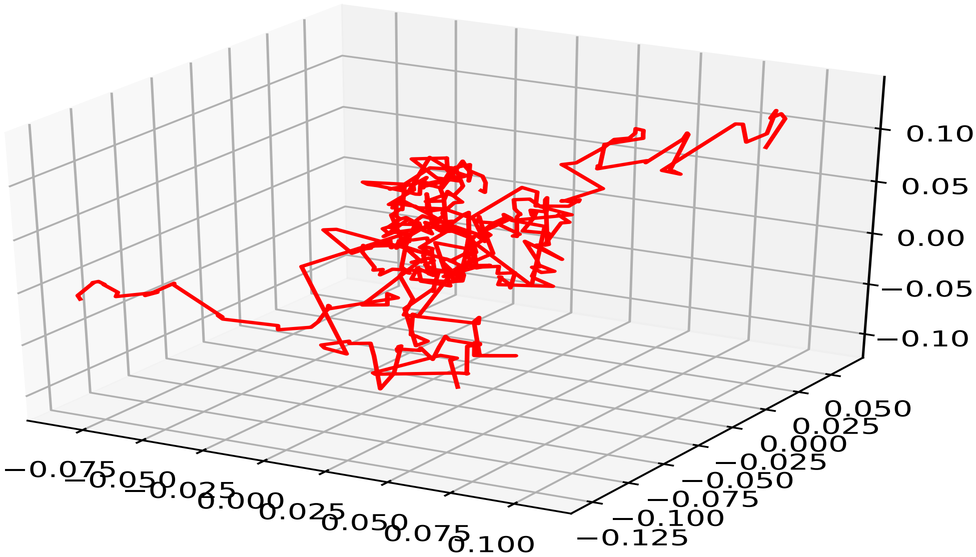

FIGURE 2 – Example of photons scattering for a packet of 10 photons in a uniform medium from a point source.

1.5 Simulating a telescope

Here we expose some idea on how we can simulate a photon detector. Our code will not be very efficient, if we count as observed only the photons that enter our telescope, which one could represent by a small aperture at some distance of the object. To make better use of the photons, we shall use the following approach :

- we shall assume that our object has rotational symmetry about the vertical (z) axis.
- once a photon escapes from the outer spherical boundary of the scattering medium, we note its position $\vec{r} = (x, y, z)$ and its direction, also given in components (vx, vy, vz)
- we construct a picture by two coordinates : the vertical coordinate shall be given by the z position of the photon.
- the horizontal coordinate in the image shall be computed with the angles $\tan\phi = y/x$, $\tan\theta = z/r$, and $\tan\beta = vy/vx$.

$$x_{\text{image}} = ||\vec{r}|| \times \sin(\beta - \phi) \times \cos\theta \tag{9}$$

This means that we compute the horizontal offset from the centre of the object, if we look from the direction in which each photon flies ... in other words, we always rotate the object around the z-axis so that the photon can hit the telescope. The factor cos takes care of the shortening of these offsets with higher latitudes of the object.

2 What you should do and paths of exploration

In this project, you will first have to write a code that simulate the radiative transfer of a source. To do so, you will have to implement what is explained in the previous section. This will constitute your starting point to create more complicated scenario.
Here are some scenario tou can explore :

- Try to simulate the radiative transfer of photons from a single source in a non uniform medium whose you can choose the geometry.

- Try to make your code more physically motivated by introducing some probability of absorption of your photons
- Try to implement multiple sources raditiave transfer.
- Try to implement mutli-frequency radiative transfer.
- Try to mimic real radiative transfer of known object in the litterature.

To explore all of this different scenario, you will have to proceed as follows : Write first simple functions and validate them before using them inside more complicated functions.