

Ay190 – Final Project

Xiangcheng Ma

Date: March 20, 2014

This document contains a description of the structure of the code and an instruction of how to use this code. A detailed study of scattering is not included. The code is well-designed and can be easily fixed to a variety of situations of scattering.

The main structure of the routine is the python script `rad_transfer.py`. This routine contains three parts. The first part is to set up the grid, the second part calls the C routine `ray_trace.so` to trace the scattering for a given number of photons and the final part returns the coordinate where the photon first escape the grid. Currently, the grid is set by hand, but it can be easily adjusted to setting the grid from a hdf5 file. And one can modify the output and make it return anything in need.

The C routine `ray_trace.c` works as following. In the head file, I define a structure “`ray_structure`” to describe a single light ray. After setting the parameter for the grid, it first allocates the memory for given number of light rays. Then I initialize the light rays by specify its coordinates and directions where it first incident into the grid. After initialization, it goes to the main loop which trace the scattering process for all light rays, where I use a specific technique to avoid photon staying at the boundary of any cell at any step. Finally, it writes the output, frees the memory and returns back to the python routine.