NERS 590 – Monte Carlo Software Design
Winter 2017
Project
Due at 8 AM on April 27, 2017

The project for this course shall be a group effort of 5-6 people (one group of 5, and another of 6) to write a 3D, continuous-energy particle transport code in C++ meeting the minimum design requirements. To receive all points, each person on the team should propose and develop a capability beyond those outlined in the minimum requirements. Any proposed capabilities should be discussed with the instructor.

The minimum design requirements are:

- Fixed-source and k-eigenvalue calculations;
- 3-D geometry supporting planes, spheres, axis-aligned cylinders, and axis-aligned cones;
- Multiple regions and constituent isotopes;
- Vacuum and reflecting boundary conditions;
- Energy-dependent neutron cross sections from JANIS data;
- Support for capture, fission, and elastic scattering;
- Isotropic and anisotropic scattering and energy-dependent, free-gas scattering;
- Fixed sources with prescribed distributions in space and direction with support for in space: an arbitrary number of point sources, uniform spherical shell sources, uniform disk and cylindrical annular sources; in direction: isotropic, linearly anisotropic about an arbitrary axis, and a beam source; in energy: an arbitrary number of discrete energies and a Watt fission spectrum; in time: an arbitrary number of discrete times;
- Estimators that can calculate the current crossing a surface and reaction rates in cells; all estimators should be capable of energy or time binning;
- Importance splitting for variance reduction;
- Compilation with a Makefile;
- Accessible from a Github repository;
- XML user input to easily run any of the test problems.

The code should be capable of solving, at a minimum, the following problems:

1. Report the detected absorption (n, p) rate of neutrons in the configurations given in problem 4 of the homework using a graphite sphere, helium-3 as a detector that is surrounded by a polyethylene moderator; the source should be monoenergetic at 1 MeV;
2. Report the detected absorption (n, p) rate of neutrons in the configuration given in problem 5 in the homework using the light-gray material as water, the dark material as boron carbide, and helium-3 as the detector; show a parametric study using importance splitting that attempts to maximize the figure of merit;
3. Report the neutron leakage rate as a function of time for a monoenergetic, isotropic, DT fusion point source at $t = 0$ the center of an HEU metal (93% enriched) sphere with a $k = 0.9$;
4. Report the energy spectrum for a critical infinite fuel cell in a light-water reactor with 3% enriched uranium dioxide fuel, zircaloy-4 cladding, and light-water moderation;

Each person in the group should also develop an additional test problem that uses the additional capability they develop. The final report shall include a description of the problems with results for these problems, and input files shall be provided in the Git repository.

The entire group will be awarded a maximum of 100 points. The group should submit:

1. Source code available on Github;
2. Nuclear and atomic data files for all isotopes and elements necessary to run the problems;
3. Benchmark suite of both unit and integral tests (to include at a minimum those given above) available with the source code and a report documenting the tests and their results;
4. Code manual documenting the included functionality and instructions for use with examples;
5. Poster describing the code capabilities.

The remaining 300 points are awarded to each person individually. Each person in the group should turn in:

1. A report detailing their specific contributions to both the source code and overall project. (up to 250 points, see below)
2. A group presentation wherein each person explains their contributions. (up to 50 points for their part of the presentation)

The individual may receive up to 150 points for contributions to the base-level code and other deliverables (to receive full any points, everyone must contribute some non-trivial portion of the source code itself; the quality of the code each person writes will be graded.). The final 100 points to the individual are awarded if the individual develops a significant capability beyond those described in the project description. All original contributions must also include a specific problem in addition to the three presented, be documented in the code manual, and have appropriate unit tests.