PHYS20762 Computational Physics

Project 3: Monte Carlo Techniques
Penetration of Neutrons Through Shielding

2019/2020 Session, Semester 2

Project Description

The task in this project is to develop a simulation of penetration of neutrons through a infinitely-wide slab of shielding of thickness **T**, considering only thermal neutrons, and the processes of absorption and scattering. You should write your code using the following steps:

- Confirm that you can generate tables of numbers using the Matlab/Python in-built random number generator **rand()/numpy.random.uniform()**, and that the numbers produced are uniform over a chosen range.
- Write a generator that produces tables of points in 3 dimensions (x,y,z). Display those points using **plot3/scatter** and confirm that the spectral problem is not present using the Matlab/Python **rand()/numpy.random.uniform()** function. In Python, to display the scatter plot, you will need **import matplotlib.pyplot as plt** and **from mpl_toolkits.mplot3d import Axes3D** (alternative plotting tools are acceptable as well). To rotate the plot in Jupyter nootbook, use **%matplotlib notebook**. Try using randssp.m and randssp.py (available in the Project 3 folder on Blackboard for Matlab/Python users) and show that the spectral problem occurs.
- Write a random number generator that generates samples distributed according to an exponential function $\exp(-x/\lambda)$ as described in the lectures. Use this to show, in the absence of scattering, that the characteristic attenuation length for water is about 45 cm, using the data in Table 1 below. (Hint: do a **polyfit/numpy.polyfit** of binned data using **hist/numpy.histogram** on the set of generated points; what is the error on the attenuation length?) Include representative plots in your report.
- Write a function that generates isotropic unit vectors **r = (x, y, z).** (***Hint***: use spherical polar co-ordinates and then do a trigonometric conversion). Confirm by eye using **plot3/scatter** for Matlab/Python that these generated points are uniformly distributed over a sphere.
- Using the function above, write another function that generates isotropic steps with lengths distributed as $\exp(-x/\lambda)$. Again, show the results using **plot3/plot** for Matlab/Python (for the latter case, you will again need **import matplotlib.pyplot as plt** and **from mpl_toolkits.mplot3d import Axes3D**). Include representative plots in your report.

Once you have written the above functions you will be able to write a complete simulation for absorption. Some of the data you will need is given in Table 1; you can look up the other data.

Table 1: Thermal Neutron Data

	Water	Lead	Graphite
Absorption σ_a/barn	0.6652	0.158	0.0045
Scattering σ_s/barn	103.0	11.221	4.74
Density /g cm^{-3}	1.00	11.35	1.67

Your project scripts will need to contain the following elements to calculate how neutrons travel through a slab of material:

- For each material, you will need to determine the macroscopic absorption and scattering cross sections, and the resultant total mean free path λ.
- Each neutron starts from one surface of the slab (take as $x=0$), and then performs a random walk where each step is drawn from the distribution $\exp(-x/\lambda)$ as above.
- After each step, you will need to check whether the neutron is absorbed in the slab, has escaped from the slab ($x<0$ or $x>T$), or continues onto a following step. (Hint: Use a while loop, generating the entire set of steps and their locations; a flag variable **is_absorbed** can be set **=1** to tell the while loop to terminate, whilst another variable **i** can be used to keep track of how many steps have been taken. The while loop should be continued if all of these conditions are true: x>0, x<T, **is_absorbed == 0**.
- Remember, the *first step* in the neutron random walks is in the *x* direction only (a different but similar function to the isotropic step function). You will probably want to use an **if** statement to choose.
- You should visualise some of these simulated random walks, again using **plot3/plot** for Matlab/Python. Include representative plots in your report.
- The absorption, reflection and transmission of N simulated neutrons (neutron '*histories*') can be determined by looking at the final positions of the neutrons, and counting up (using a **for** loop) which neutron ended up where. For example, the fraction of neutrons transmitted is just N_{trans}/N, where N_{trans} is the number of simulated neutrons whose final position have x>T.
- Determine the fraction of normally-incident neutrons transmitted, reflected, and absorbed in water, lead and graphite slabs which are 10 cm thick. How does error vary with number of simulated neutrons? Include summary data in your report.
- Calculate the variation in neutron transmission, reflection and absorption with slab thickness for the three materials. Determine the characteristic attenuation lengths for the three materials from the transmitted intensity, with estimated errors. Include summary numerical data and representative graphs. **Hint**: when fitting the transmitted neutron data you may have some thickness values give zero transmitted neutrons, which will stop **polyfit/np.polyfit** from working (**log(0) = -Inf**). You need to remove data points containing **–Inf** before fitting.

Final Work: Use the Woodcock method to calculate the fraction of neutrons transmitted through two slabs of different materials in contact with each other, each of thickness 10 cm. How can you check your simulations are correct?

The report for the project should be written up in the style of a scientific paper, as you do for the experiments in the general teaching laboratories. Include a brief introduction, a brief description of the basics of the method, all graphs with comments/explanations and a table with summary of main numerical results. The report should also include the program code (as an appendix). Once again, the instruction on Blackboard regarding the formatting of the report and its length apply. Out of a total of 100 marks for the project, 20 will be allocated for presentation and scientific style, 30 for the code (correctness, style), 10 for the use of graphics, 15 for the output, 15 for initiative and originality, and the remaining 10 for the 'Final Work'. The project contributes 35% to the total mark for the course.

Using Plots

Once you have a plotted figure, you can save it in a variety of formats. **`.png`** (Portable Network Graphic) format is usually best for inserting into Word reports, or whilst **`.eps`** (Encapsulated Postscript) are best for LaTeX. Don't forget to add appropriate axes labels, and to make sure that the text on the plot is readable.

21/01/2020 Dr Hywel Owen (adapted from previous material)