FYS4711 - Mandatory exercises in Monte Carlo simulations

This exercise can be done using any programming language, but it is an advantage to employ a high-level language with in-built functions such as Matlab or Python+Numpy.

Part 1

1. *Know your random number generator*. Generate a list of 1000 random numbers sampled from a) a uniform distribution between 0 and 1 and b) a normal (Gaussian) distribution with a mean (μ) of zero and standard deviation (σ) of 1. Plot the distributions as histograms with a suitable bin size. Are the distributions as expected?

2. *Random walk in 1 dimension*. A 'particle' is to move in one dimension where the step length Δs follows a normal distribution with μ=0 and σ=1. Simulate 1000 particles with 100 steps ('collisions') for each particle. Save the position at each collision for each particle. Plot the position (y) vs the collision number (x) for 10 arbitrary particles. Plot the full 1D position *distribution* (for all particles) as a histogram. Is the distribution as expected, qualitatively speaking?

3. *Random walk in 1 dimension including energy loss*. Start with the code developed under task 2. Keep the settings for step length. Introduce a kinetic 'energy' T_0 (use a suitable number, e.g. 100) for each simulated particle at the start of the simulation. Assume that following each step the kinetic energy is reduced with $k\Delta s$, where k is the energy loss per step. Find a k-value so that each particle *on average* needs 100 steps to reduce its kinetic energy to zero. Thereafter, run a simulation with 1000 particles. Save the position and energy at each step for each particle. Plot the resulting position distribution as a histogram. How does it compare with the distribution in task 2? Discuss the difference. Make a dot plot where you show the energy (y) as a function of position (x) for all collisions (in this case, show data for e.g. 100 particles). Does the plot make sense from a 'physics' perspective, also considering the stochastic nature of the simulated processes?

Part 2

The following simulations are for *photons in water*. The number of electrons per volume unit in water is n_V = 3.43×10^{23} cm^{-3}. We assume that the only interaction taking place is Compton scattering. The minimum and maximum photon energy is 50 and 2000 keV, respectively

4. *Create attenuation coefficient function*. Implement the electronic cross section for Compton scattering in your code (define it as an energy-dependent function) using eq. 7.15 in Attix. Express σ in [cm^{-2}]. Compare with values in figure 7.6 in Attix. Calculate the attenuation coefficient μ (again as a function) from σ and n_V. Make a plot of μ (in [cm^{-1}]) as a function of photon energy. Calculate explicitly values of μ for 200 keV and 2 MeV photons. Compare to published values of 0.137 cm^{-1} and 0.049 cm^{-1}, respectively. If you find large differences (> 10 %), you may want to check your code before you proceed….!

5. *Probability distributions.* Make a probability distribution function (PDF) and cumulative distribution function (CPD) for simulating the depth of interaction. Use μ defined under 4. Plot the PDF and CPD for 200 keV and 2 MeV photons.

6. *Sampling photon steplengths using the inverse transform.* Sample photon steplength Δs using the inverse transform method for 200 keV and 2 MeV photons. Use the μ–function from above. Simulate 1000 photons. Present the data as histograms and compare with respective PDFs. Are the computational (MC) and analytical (~ $e^{-\mu x}$) depth distributions similar? Also calculate mean MC steplength and compare to mean free path for photons. *[Additional task for those with particular drive/interest: sample photon steplengths using the rejection technique (see also 8.) and compare with results from inverse transform.]*

7. *Normalized differential scattering cross section.* Define the differential cross section $d\sigma/d\theta$ for Compton scattering (eq. 7.13) as a function. Remember to include the 'sin θ'-term from the solid angle element (slide 40, MC lecture). Further hint: employ Compton's formula (slide 39, $h\nu'$ = …) directly into this implementation, as it links the scattered photon energy to the scattering angle. Find the maximum of $d\sigma/d\theta$ numerically and normalize the function with this value. Plot the normalized $d\sigma/d\theta$ as a function of θ for 200 keV and 2 MeV photons.

8. *Sampling Compton scattering angles using the rejection technique.* Use the rejection technique to sample scattering angles for 200 keV and 2 MeV photons. Simulate 1000 photons in each case (remember that the number of pairs of random numbers will be roughly double due to the rejections). Plot the resulting angular MC samples as histograms together with $d\sigma/d\theta$.

9. *Photon trajectory simulation.* Use the 'recurrence transformation formula' (see paper by Persliden or MC lecture notes slide 44-45) to simulate photon trajectories. Employ code you have developed above. Remember to also include sampling of the azimuthal angle. Save x, y and z position and energy for each photon and interaction. Plot example trajectories (2D or 3D), e.g. for 100 photons. Estimate the mean photon energy as a function of depth z (employ more than 1000 photons). Perform for both 200 keV and 2 MeV photons.