Exercise Lecture X
Lattice gas
Simulated annealing

(results for items in red have to be included in the homework report)

1. **Self-diffusion coefficient in a lattice gas model**

   Consider a finite square lattice with sites randomly occupied by particles with a given density ρ. The particles can move randomly to *empty* nearest sites (two particles can not occupy the same site). It is an example of a restricted random walk. A meaningful physical quantity is the *self-diffusion coefficient D* of an individual particle. D is the limit $t \to \infty$ of $D(t)$, where $D(t)$ is given by:

   $$D(t) = \frac{1}{2dt}\langle \Delta R^2(t)\rangle,$$

   with d which is the dimensionality of the system and $\langle \Delta R^2(t)\rangle$ is the net *instantaneous* mean square displacement per particle, averaged over all particles, after t units of time ($\langle ...\rangle$ *here indicates the average over particles and <u>not</u> temporal averages*).
   The dynamical model can be summarized by the following algorithm:

   i) Occupy at random the $L \times L$ sites of a square lattice with N particles subject to the condition that no double occupancy is allowed, with the desired density $\rho = N/L^2 \leq 1$. Tag each particle, that is, distinguish it from the others, and record its initial position in an array.

   ii) At each step choose a particle (randomly, or, alternatively, in an ordered way) and one of its nearest neighbor sites at random. If the neighbor site is empty, the particle moves to this site; otherwise it does not. Loop over the particles.

   Note 1: The measure of "time" in this context is arbitrary. The usual definition is that during one unit of time or one Monte Carlo step, *each particle* on average *attempts one jump. Time goes on even if the particles do not move, i.e., the tentative move is not accepted.*
   Note 2: Consider periodic boundary conditions, but note that reliable results can be obtained only for $\langle \Delta R^2(t)\rangle < (L/2)^2$ *(this sets a limit to number of MC steps). Otherwise, they could be affected by the imposed periodicity.*

   Do a Monte Carlo simulation to determine D and its dependence on the particles concentration ρ.

   See for instance the code `latticegas.f90`. Internal units for Monte Carlo time step and displacement should be preferred. For comparison with a realistic situation, such as for instance diffusion in solids, we may consider Monte Carlo time step equal to 1 ns and the unit length to 2 Å, properly rescaling the internal quantities at the end of the calculations.

   (a) Study $\langle \Delta R^2(t)\rangle$ as a function of time for a fixed value of $\rho = 0.03$ and for a fixed number of particles (e.g., 13 particles in a 20×20 lattice). What do you see increasing time (within the limit mentioned above)? Make a fit and compare your result (the slope) with the expected behavior of a standard random walk.

(b) Repeat for $\rho = 0.2$.

(c) Plot $D(t)$ as a function of time: after a certain equilibration time, it fluctuates. Calculate the amplitude of the fluctuations as a function of t (from the distribution of data over the particles). These fluctuations remain also by increasing t.

(d) In order to estimate D, which is defined as the limit of $D(t)$ for $t \to \infty$, do a temporal average $\langle D(t) \rangle_T$ ($\langle ... \rangle_T$ here indicates a "global" temporal average on t from 0 to T or some block average). Plot together $D(t)$ and $\langle D(t) \rangle_T$. Change the seed, do another run and compare $\langle D(t) \rangle_T$ with the previous results. An estimate of D can be obtained by averaging $\langle D(t) \rangle_T$ over different runs.

(e) Better statistics for $D(t)$ (and consequently for D) can be obtained by calculating $\langle \Delta R^2(t) \rangle$ as average over many particles (i.e., for a given ρ, considering a lattice with L as large as possible; it is suggested $L \geq 40$). Verify that fluctuations of $D(t)$ (and the deviations of $\langle D(t) \rangle_T$ over more runs from its mean value) are proportional to the inverse square root of the number of particles.

(f) Study the dependence of D on the concentration, using for instance ρ=0.1, 0.2, 0.3, 0.5, and 0.7. You should find that D is a monotonically decreasing function of ρ. Why?

(g) To gain some insight into this dependence, determine the dependence on ρ of the probability that if a particle jumps to a vacancy at time t, it returns to its previous position at time $t + 1$. Is there a qualitative relation between the density dependence of D and this probability?

2. **Simulated annealing**

Simulated annealing is a stochastic method for global energy minimization, considering the system starting from a sufficiently high temperature; at each temperature it goes towards equilibrium according to the Boltzmann factor (see the application of the Metropolis algorithm in the canonical ensemble); then the temperature is slightly reduced and the equilibration procedure is repeated, and so on, until a global equilibrium state is reached at T=0. The method can be efficiently used for function minimization, even if the function is not representing an energy. In program `simulated_annealing.f90` it is implemented for the minimization of $f(x) = (x + 0.2) * x + cos(14.5 * x - 0.3)$. Initial temperature, initial position and scaling factor for the temperature are input quantities. Test the program by choosing different initial parameters and scaling factor for the temperature. For instance:

(a) *Annealing rate*: Try different annealing factors (0.8, 0.9, 0.95) with different random seeds. How slowly do you have to anneal to settle down in the global minimum over 90% of the time? Just estimate the quantity, but provide some explanation and data to support your answer.

(b) $\sqrt{T}$ *scaling*: It is often convenient to scale by $\sqrt{T}$ the step size used in the Metropolis algorithm. Why? *Hint, look at the acceptance ratios, and think about diffusion and thermal distributions in parabolic wells*

3. **MC simulation of a simple N-particles model**

Consider an ideal gas of N non interacting, distinguishable particles, **confined** in a box (fixed $\mathbf{V}$) and **isolated** (fixed $\mathbf{E}$), divided into left/right with the possibility for one particle at a time to pass through the separation wall, with equal probability from the left to the right or viceversa.

A **macrostate** is specified for instance by the number of particles on the left side, say n, that can correspond to different **microstates** depending on the list of the specific particles there. A Monte Carlo approach consists in generating a certain number of movements, randomly, and consider them as representative of all the possible movements. The program `box.f90` is a possible implementation of the algorithm describing the time evolution of the system in terms of macrostates, i.e. –given an initial number of particles on the left, n– the approach to equilibrium and what the equilibrium macrostate is.

(a) Choose N=4, 10, 20, 40, 80, and $n=N$ initially. Make a plot of n (or, better, of n/N) with respect to time. What is the equilibration time τ_{eq} (=how many MC steps)?

(b) Modify the program so that at each time step t it calculates the number of particles $< n(t) >$ averaged over different runs (e.g. 5 runs). Make a plot to compare $n(t)$ over the individual runs and averaged $< n(t) >$.

(c) *(Optional)* Compare the numerical value of $< n(t) >$ with the exact analytic results for a simple case, for instance N=4.

(d) *(Optional)* Consider only one run. Modify the program to calculate numerically the probability P_n of having *at equilibrium* a macrostate with n particles on the left, by simply counting the number of occurring microstates that correspond to the macrostate n and dividing for the total number of microstates generated in the time evolution. Plot the histogram P_n for N=20, 40, 80 and a "sufficiently" long run. Comment.

(e) Modify the program to measure the statistical fluctuations at the equilibrium, by calculating the variance $\sigma^2 = < n^2 > - < n >^2$, where the average is done over a time interval *after* reaching the equilibrium.

(f) Determine $< n >$ and $\sigma / < n >$ at equilibrium for N=20, 40, 80. Which is the dependence of these quantities on N?

(g) An alternative method to find the equilibrium macrostate is the calculation of the entropy $\mathcal{S}_n$ of the different possible macrostates, by looking at the one with maximum entropy. An efficient numerical implementation is feasible by evaluating the ratio $\mathcal{R}_n=$ *sum of possible coincidences for each microstate/maximum number of possible coincidences for each microstate*, then calculating $\mathcal{S}_n \propto - \log \mathcal{R}_n$. The code `entropy.f90` calculates $\mathcal{R}_n$ and $\mathcal{S}_n$. Use it with N=10. Compare the numerically calculated $\mathcal{S}_n$ with the analytical value.