

# Introduction to Molecular Simulation and Statistical Thermodynamics

email suggestions and errors to [t.j.h.vlugt \(at\) tudelft.nl](mailto:t.j.h.vlugt@tudelft.nl)

## 1 Suggested/Required Software

The following software is suggested or required:

- fortran compiler (`gfortran`)
- vmd  
visualization software for molecular systems.  
See also <http://www.ks.uiuc.edu/Research/vmd>
- xmgr or xmgrace  
software to make 2D graphs (<http://plasma-gate.weizmann.ac.il/Xmgr>).  
Alternatively, one could use `gnuplot` instead.
- fig2dev  
part of the `transfig` package. See also <http://www.xfig.org>
- convert and display  
Part of the ImageMagick package. See also <http://www.imagemagick.org>

## 2 Introduction to the Computer Exercises

Go to the newly created directory `IMSST` by typing `cd IMSST` (`cd` means change directory). You can go back to the previous directory by typing `"cd .."`. The command `pwd` displays the name of the current directory. By typing `ls` or `ll`, you can list the contents of this directory. As an example, go to one of the subdirectories by typing `cd HarmonicOscillators`. In this directory you find the directories `Run` and `Source`. The directory `Source` contains the sourcecode as well as the executable (here: `harmonic`). Usually, it is not needed to modify or studying the given sourcecode before making the questions. The executable in the `Source` directory is created by compiling the code. This can be done by going to the `Source` directory and typing `make`. If one would change the code, the code *has* to be recompiled. The directory `Run`, that can be accessed by `cd ../Run` (from the `Source` directory), contains a script called `run`. To start the executable (in the directory `Source`) you have to start this script by typing `./run`. In this script the input parameters for the program are given as well as a brief

Question Number	Program Directory
4	HarmonicOscillators
8	IdealGasEnergy
11	IncreasingEntropy
13	IncreasingEntropy
17	HarmonicOscillators
18	CentralLimit
25	GasMixtureAB
27	GasMixtureAB
29	GasMixtureAB2C
33	Ising-Random
37	Ising-MC
51	LJ-MC
53	LJ-MC
61	LJ-DifferentEnsembles
65	LJ-DifferentEnsembles
70	LJ-DifferentEnsembles
74	Umbrella-Ising
78	EinsteinCrystal and LJ-DifferentEnsembles
82	MolecularDynamics
84	MolecularDynamics
86	MolecularDynamics

Table 1: Program directories.

summary of the input/output files. In most exercises you will have to modify the input parameters, which means that you will have to edit the script `run`. This can be done by using a text editor like `emacs` or `vi`.

In most exercises, the output of a computer program can be (1) a file containing data, (2) an animation (animated `gif` file) or (3) a trajectory. In the script `run` a description of the various output files is listed. Animations (animated `gif` files) can be visualized by issuing the command `display animation.gif`. Graphs can be plotted using the program `xmgr`. In this document, a short description of this program is presented. Trajectory files have the extension `.xyz` and can be visualized using the program `vmd`.

Often, one would like to compare graphs for various input parameters. Therefore, it can be useful to rename some of the output files of a computer program. This can be done using the `mv` (“move”) command, i.e. `mv result.dat result-100particles.dat` will rename the file `result.dat` (and overwrites a file `result-100particles.dat` if present).

## Question 4 (Simulation of a System of Oscillators)

Note that in this exercise  $\epsilon = \hbar\omega$  is set to 1, which means that the energy of an oscillator can be  $0, 1, 2, 3, \dots$ . Go to the directory `HarmonicOscillators` (so type `cd HarmonicOscillators`). In the script `run` (in the directory `Run`) you can change the total energy stored in the system (`Ettotal`), the number of oscillators (`Noscillator`), the number of cycles (`Ncycle`) and the number of oscillators used to calculate the energy distribution (`m`). In subquestions 1–4, the energy distribution of the *first* oscillator is studied so `m` is set to 1 (note that in subquestion 5, `m` has to be set to 2). The number of cycles is proportional to the number of times that an attempt is made to transfer an energy package from one oscillator to another. A larger number of cycles means more accurate results, but it also takes more computer time (this is essentially true for *all* exercises of table 1). You will have to develop some sort of feeling for how many cycles are needed to obtain results that are accurate enough, i.e. try the simulation using twice (or half) the number of cycles and compare the results with those of the previous simulation. When you run the program again using other values for the input parameters, the old output files are overwritten. To keep the old files you will have to rename those files as explained before. One of the output files is called `animation.gif`. This animation can be displayed by typing `display animation.gif` and it contains 100 snapshots of the energy distributed over all oscillators. As you can see, the frames definitely do not resemble a quiet “sea” (why is this ?) Another output file is `results.dat`, which is a histogram of the energy distribution of the first `m` oscillators. By typing `more results.dat` you can see two columns. In the first column all energy levels are printed and in the second column the (normalized) probabilities that an oscillator is in that energy level are printed. The result can be plotted by `xmgr results.dat`. Subquestion 4 requires a transformation of  $P(k)$  to  $\ln P(k)$ . In `xmgr` this can be done by using `Data - Transformations - Evaluate Expression` and the formula `y=ln(y)`. Linear curve fitting can be done by `Data - Transformations - Regression`.

## Question 8 (Ideal Gas Energy)

Note that in this exercise the value of  $\hbar^2/8mL^2$  is set to 1. Use the program in the directory `IdealGasEnergy` (so type `cd IdealGasEnergy`) and modify the input data in the file `run` (directory `Run`). The measurements of  $P(\mathbf{k})$  are stored in `result.dat`. In the first column you find the energy  $E_{\mathbf{k}}$  of state  $\mathbf{k}$ . Note that different states  $\mathbf{k}$  may have the same energy  $E_{\mathbf{k}}$ , so some energies appear more than once in this file. The second column lists the probability that the first particle is in state  $\mathbf{k}$ . The other three columns are the three quantum numbers of state  $\mathbf{k}$ . To plot  $P(\mathbf{k})$  as a function of the energy  $E_{\mathbf{k}}$  of state  $\mathbf{k}$ , you have to plot the first two columns of `results.dat` by typing `xmgr results.dat`. To see what is going on, you need to change some settings, `Plot - Symbols - Line properties: Style: none - Accept` and `Plot - Symbols - All symbols`. When you zoom in, you see that states that have the

same energy  $E_k$  also have the same probability  $P(\mathbf{k})$ . Plotting  $\ln[P(\mathbf{k})]$  as a function of  $E_k$  shows that at larger  $E_k$  it *appears* that not all states with equal  $E_k$  are equally likely. Why is this? Hint: try increasing the number of cycles.

### Question 11 ( $\ln W(\mathcal{N})$ Increases with Time)

Use the program in the directory `IncreasingEntropy`. In this question, we will consider a system of  $N$  (`Npart`) harmonic oscillators with total energy  $E$  (`Etotal`). At time zero, all energy is inside the first oscillator so the system distribution equals  $\mathcal{N} = (N - 1, 0, 0, \dots, 0, 1)$ . At each timestep, a single unit of energy is exchanged between two oscillators. This process (starting from  $\mathcal{N} = (N - 1, 0, 0, \dots, 0, 1)$  and exchanging energy packets) is repeated a large number of times (proportional to `Ncycle`). The file `entropy.dat` lists the entropy ( $\ln W(\mathcal{N})$ ) as a function of time. The final system distribution  $\mathcal{N}$  at the end of the simulation is plotted in `systemdist.dat`. Only the non-zero elements of  $\mathcal{N}$  are listed in this file.

### Question 13 (Temperature of a Distribution)

Use the program in the directory `IncreasingEntropy`. See the description of question 10.

### Question 17 (Division into Subsystems)

Use the program in the directory `HarmonicOscillators`.  $E_m$  is the energy in the first  $m$  oscillators. Change the variable  $m$  in the script `run` and save the “average energy” ( $\langle E_m \rangle$ ) and the “standard deviation” ( $\sigma_{E_m}$ ) of the energy distribution from the *output of the program*. Note that you have to consider  $\sigma_{E_m} / \langle E_m \rangle$  as a function of  $m$  and not  $\sigma_{E_m}$  as a function of  $m$ .

### Question 18 (Central Limit Theorem)

Use the program in the directory `CentralLimit`. Note that  $N$  should be 1 to make a histogram  $p(R_1)$  (subquestion 1). In subquestions 1 and 2,  $R$  is a uniformly distributed random number between 0 and 1 and therefore `Lran` should be set to `.true..` For subquestion 3 only, change the parameter `Lran` in the script `run` from `.true.` to `.false.` to use a random number that is 0 with probability  $P$  and 1 with probability  $1 - P$ . The distribution  $p(a)$  is plotted in the file `result.dat`. To fit this distribution to a Gaussian shape, click `Data - Transformations - Nonlinear curve fitting` and use as equation  $y = A0 * \exp(-(x-A1) * (x-A1) / (2 * A2 * A2))$  with 3 fit parameters  $A0, A1, A2$ . Make sure that the initial guesses of  $A0, A1, A2$  are reasonable.

### Question 25 (Probability Distribution of Energy in Mixtures)

The program in the directory `GasMixtureAB` simulates an ideal gas mixture of molecules A and B. Note that in this question we leave out a chemical reaction, thus `Lreaction` in the script `run` should be set to `.false..` In this simulation only energy can be exchanged between two molecules. It does not matter which gases these particles are. The number of `trialmoves` is proportional to `Ncycle`. The distributions  $E_k$  for A and B are given in the files `DistributionA` and `DistributionB` respectively.

### Question 27 (Simulation of an Isomerization Reaction)

Use the program in the directory `GasMixtureAB`. In this question `Lreaction` should be set to `.true.` which means that the reaction can take place. Note that the fitting constant  $\beta$  may change if  $E_{0,A}$  or  $E_{0,B}$  is changed (why is this?). The instantaneous value of the equilibrium constant is given in `Kt` and its probability distribution in `DistriKt`.

### Question 29 (Equilibrium Constant of an Ideal Gas Reaction: Simulations)

Use the program in the directory `GasMixtureAB2C`. Note that the total number of molecules ( $N_A + N_B + N_C$ ) is constant during the reaction. The instantaneous value of the equilibrium constant is given in `Kt` and its probability distribution in `DistriKt`.

### Question 33 (Random Sampling of the 2D Ising System)

The program in the directory `Ising-Random` computes average properties of the 2D Ising system for a given value of the inverse temperature  $\beta\epsilon = 1$ . The lattice size  $N$  (total number of spins equals  $N^2$ ) can be adjusted in the script `run`. Our suggestion is to start with  $N = 5$  and to check if it is possible to reproduce the estimation of  $\langle E \rangle$  for low temperatures (Eq. 5.15). The distribution  $p(M)$  in which  $M$  is the magnetization is plotted in the file `magnetic.dat`. The file `animation.gif` shows 100 of these randomly generated configurations. The total number of random configurations considered in the calculations is proportional to `Ncycle`, so increasing `Ncycle` will lead to a better estimate of ensemble averages.

### Question 37 (Monte Carlo Simulation of the 2D Ising System)

The program in the directory `Ising-MC` uses a Monte Carlo simulation to compute the same average properties as in question 30. The main difference is that not random configurations are generated, but instead the Monte Carlo algorithm generates configurations (states) with a probability proportional to their Boltzmann factor, which means

that averages can be calculated using Eq. 5.16. Perform a simulation at low temperature (large value of  $\beta$ ) and compare the animation with the animation of question 30 (using the same values for  $\beta$  and  $N$ ). Use this to explain why random sampling is usually extremely inefficient. The total number of Monte Carlo trial moves is proportional to `Ncycle`. The first 33% of all cycles is not used for updating ensemble averages.

### Question 51 (Equation of State of the Lennard–Jones System)

The program in the directory `LJ-MC` simulates a system of  $N$  particles in volume  $V$  that interact using a Lennard–Jones potential. Note that not the volume  $V$  but the boxlength `Box` is specified and that the volume  $V$  equals `Box*Box*Box`. The number of particle displacements is proportional to the number of cycles (`Ncycle`) and during the first `Ninit` cycles no data is collected for the calculation of ensemble averages. The maximum displacement `Deltax` should be set according to the guidelines outlined in section 6.7.2 of the lecture notes, note here that `Deltax` may be different for different densities. When `Linit=.true.`, a random initial configuration is generated and when `Linit=.false.` the initial configuration is read from the file `Coordold`. The file `Results` lists the energy per particle (second column) as well as the instantaneous pressure (third column) as a function of time. This information can be used to investigate if `Ninit` is set correctly. The trajectory can be visualized by typing `vmd Traject.xyz` and set `Display` to `Orthographic` and `Graphics - Representations - Drawing Method` to `VDW`. The radial distribution function  $g(r)$  is listed in the file `Radial`.

### Question 53 (Direct Simulation of Coexistence Densities)

Use the program in the directory `LJ-MC`. Set the temperature and density to the values listed in the lecture notes. Make sure that `Ncycle` is large enough (for example 10000). Play the “movie” of this simulation by typing `vmd Traject.xyz` and set `Display` to `Orthographic` and `Graphics - Representations - Drawing Method` to `VDW`.

### Question 61 (Comparison between Different Ensembles)

Use the program in the directory `LJ-DifferentEnsembles`. This is an extended version of the program `LJ-MC`. It is suitable for the  $NVT$ ,  $\mu VT$ ,  $NPT$  and Gibbs ensemble. In the  $NVT$ ,  $NPT$  and Gibbs ensemble, the chemical potential  $\mu$  as well as the excess chemical potential  $\mu_{ex}$  are computed automatically using Widom test particle method (Eqs. 8.11, 9.22 and 9.24). If `Linit=.True.`, an initial configuration is generated by placing all particles at random positions in the simulation box. If `Linit=.False.`, initial coordinates are read from the file `Coordold`. A coordinate file `Coord-fcc-rho1.2-256particles` is provided as a starting configuration for an

*fcc* crystal of 256 particles at  $\rho^* = 1.2$ . To start with these coordinates, this file can be copied to `Coordold`. A description of the various input/output files can be found in `LJ-DifferentEnsembles/Run/run`.

### Question 65 (Gibbs Ensemble Simulation)

Use the program in the directory `LJ-DifferentEnsembles`. A description of the various input/output files can be found in `LJ-DifferentEnsembles/Run/run`.

### Question 70 (Widom Test Particle Method)

Use the program in the directory `LJ-DifferentEnsembles`. The chemical potential  $\mu$  as well as the excess chemical potential  $\mu_{\text{ex}}$  are computed automatically using Widom test particle method. A description of the various input/output files can be found in `LJ-DifferentEnsembles/Run/run`.

### Question 74 (Umbrella Sampling of a 2D Ising Model)

Use the program in the directory `Umbrella-Ising`. If `Linit=False.`, the weightfunction  $W(M)$  is read from the file `wold.dat`. If `Linit=True.`, the weightfunction is set to zero for all  $M$  (so then  $W(M) = 0$ ) and the content of the file `wold.dat` is ignored. The program produces a better approximation for the weightfunction (`wnew.dat`) in such a way that the distribution  $p(M)$  is more flat. The following initial weightfunctions are provided by us: `w.type1.dat`, `w.type2.dat`, `w.type3.dat`.

### Question 78 (Melting of a Lennard-Jones crystal)

Use the program in the directory `EinsteinCrystal` to calculate the ensemble average  $\langle \partial U / \partial \lambda \rangle$  as a function of  $\lambda$  for an *fcc* crystal at a fixed center of mass. Note that in this program, the volume  $V$  of the system is used as input instead of the boxlength  $L = V^{1/3}$ . To simulate an *fcc* crystal *without* a fixed center of mass (e.g. to calculate the average energy or pressure), use the program in the directory `LJ-DifferentEnsembles` and `Coord-fcc-rho1.2-256particles` as starting coordinates.

### Question 82 (Molecular Dynamics of a Lennard-Jones System)

Use the program in the directory `MolecularDynamics`. Make sure that `freq` is set to 0 for simulating a NVE ensemble. Note that you first have to find the 3 errors in the program. A description of the various input/output files can be found in `MolecularDynamics/Run/run`.

### **Question 84 (Self Diffusion)**

Use the program in the directory `MolecularDynamics`. Make sure that `freq` is set to 0 for simulating a NVE ensemble. Make sure that the three errors of question 79 are corrected. A description of the various input/output files can be found in `MolecularDynamics/Run/run`.

### **Question 86 (Andersen Thermostat)**

Use the program in the directory `MolecularDynamics`. Make sure that the three errors of question 79 are corrected. A description of the various input/output files can be found in `MolecularDynamics/Run/run`.