

Exercise_6 Molecular Dynamics Simulation

Andrea Scaioli

June 2025

1 Introduction

The Gibbs free energy, denoted as ΔG , is a fundamental thermodynamic potential that determines the spontaneity of a process at constant temperature and pressure. It is defined as:

$$\Delta G = \Delta H - T\Delta S, \quad (1)$$

where ΔH is the change in enthalpy, T is the absolute temperature, and ΔS is the change in entropy. A negative value of ΔG indicates that a process can occur spontaneously under the given conditions.

In computational chemistry and biophysics, the Gibbs free energy is crucial for quantifying a wide range of biologically and chemically relevant phenomena, such as protein-ligand binding, conformational stability of macromolecules, and solvation processes. Specifically, the solvation free energy ΔG_{solv} represents the free energy change when a solute transfers from vacuum (or gas phase) into a solvent environment. This property captures the balance of enthalpic and entropic contributions upon solute-solvent interactions and is essential for understanding molecular behavior in solution.

Due to the complexity of solvent environments and molecular interactions, direct calculation of ΔG_{solv} is not straightforward. Instead, an elegant and widely-used approach is the *alchemical route*. In this framework, unphysical but computationally tractable transformations are performed, where the solute's interactions with the environment are gradually turned off using a coupling parameter $\lambda \in [0, 1]$. The fully interacting state (realistic solute in solution) corresponds to $\lambda = 0$, while the non-interacting (ghost) state is reached at $\lambda = 1$.

This transformation is carried out in discrete λ steps, and molecular dynamics simulations are performed at each state to sample relevant configurations. The free energy differences between adjacent λ states are then calculated using statistical methods such as the Bennett Acceptance Ratio (BAR). Summing these differences yields the total free energy change of decoupling the solute from the solvent, which corresponds to $-\Delta G_{\text{solv}}$ due to the direction of the transformation.

The Bennett Acceptance Ratio (BAR) method is a statistically optimal technique for estimating free energy differences between two states sampled via molecular dynamics or Monte Carlo simulations. In the context of alchemical transformations, BAR compares adjacent λ states by evaluating the overlap of their potential energy distributions. Specifically, it uses configurations sampled at state i to compute their potential energy under the Hamiltonians of both state i and a neighboring state j , and vice versa. The method then determines the free energy difference $\Delta G_{i \rightarrow j}$ that best balances the forward and reverse transitions, minimizing statistical error. Mathematically, BAR solves a transcendental equation involving the Fermi function to find the optimal free energy difference. Compared to simpler estimators like exponential averaging (Zwanzig's formula), BAR is more accurate and efficient, particularly when there is sufficient configurational overlap between states. This makes it especially suitable for alchemical free energy calculations involving a series of intermediate λ points.

This exercise focuses on computing the solvation free energy of a methane molecule in water using GROMACS. By applying the alchemical route and BAR analysis, we aim to reproduce the literature value and gain deeper insights into free energy methods.

2 Methods

To compute the solvation free energy via the alchemical route, we performed a series of molecular dynamics simulations at different values of the coupling parameter λ . For each λ window, the simulation protocol fol-

lowed four main stages: energy minimization (EM), constant volume equilibration (NVT), constant pressure equilibration (NPT), and production molecular dynamics (PROD). The energy minimization phase ensures the removal of any steric clashes or unfavorable contacts in the initial configuration. The NVT and NPT phases gradually equilibrate the system first at a fixed volume and then at the desired pressure, stabilizing temperature and density, respectively. The production run then collects data from a fully equilibrated system for further analysis. To efficiently manage the large number of simulations across all λ values (21 windows in total), we developed two bash scripts: `generate_mdp_files.sh`, which generates the necessary MDP input files for each simulation phase and lambda value, and `run_lambda_simulations.sh`, which executes all simulations in the correct sequence. Finally, the free energy differences between adjacent λ states were calculated using the `gmx bar` command, which applies the Bennett Acceptance Ratio (BAR) method to estimate the total solvation free energy ΔG_{solv} . The codes are listed below.

```

1 #####
2 # GENERATE MDP FILES #
3 #####
4
5 #!/bin/bash
6
7 mkdir -p MDP
8 echo "Should we just test the simulation workflow? [y/n]"
9 read TEST
10
11 # for loops a very useful tools in bash and also in almost
12 # every other programming language as well
13 # here, a variable is set iteratively to the values listed after 'in'
14 # These values, are to be set by you. They must match the names already in use
15 # in this directory as can be seen below: f{mdp}.mdp corresponds to the mdp
16 # files in your exercise directory. If you are unsure, ask the tutor or the internet.
17 for mdp in em npt nvt prod
18 do
19     for i in {0..20}
20     do
21         # What means the '-E' option of 'sed'? Add some comments
22         # -E tells sed to use Extended Regular Expressions, which allows for more advanced patterns
23         # without needing to escape every grouping or special character (like () or +).
24
25         sed -E "s/(init_lambda_state\s+=\s+)[0-9]\1$/ " f{mdp}.mdp > ./MDP/f{mdp}_$i.mdp
26
27         if [[ $TEST == y ]];
28         then
29             # what does this command do? Add some comments
30             sed -i 's/nsteps.*/nsteps = 500/g' ./MDP/f{mdp}_$i.mdp
31         fi
32     done
33 done
34
35 #####
36 # RUN LAMBDA SIMULATIONS #
37 #####
38 #!/bin/bash
39
40 # Set some environment variables
41 FREE_ENERGY=$(pwd)
42 MDP=$FREE_ENERGY/MDP
43
44 # make the user aware of his/her choices
45 echo "Free energy home directory set to $FREE_ENERGY"
46 echo ".mdp files are stored in $MDP"
47
48

```

```

14  # loop over all values of lambda
15  #  $\ell()$  is called a command substitution, where the output of that command
16  # can be transferred into a new command or, in this case into
17  # a variable used in a for loop for more information google 'command substitution bash'
18  # and for more information about seq, google it or type 'man seq' in your terminal
19  for l in $(seq 0 1)
20  do
21      mkdir -p lambda_$l
22      echo "Starting minimization for lambda = $l ..."
23
24      # Define Variables
25      wd_prev=.
26      sim_prev=methane_water
27      lambdadir=$FREE_ENERGY/lambda_$l          # current lambda value
28
29      # loop over all different simulation steps
30      for sim in em nvt npt prod
31      do
32          # preliminaries before the simulations are executed
33          cwd="$lambdadir/$sim"                  # set the current working directory
34          cat $sim.banner                        # Concatenate the according banner
35          mkdir -p $cwd                          # create the current working directory
36
37          # start the run of the corresponding lambda and simulation step
38          gmx grompp -f $MDP/${sim}_${l}.mdp -c $wd_prev/$sim_prev.gro -p $FREE_ENERGY/topol.top \
39                  -o $cwd/$sim.tpr -po $cwd/mdout.mdp &>$cwd/grompp.$sim \
40                  || { echo "something went wrong, check $cwd/grompp.$sim"; exit; }
41
42          gmx mdrun -deffnm $cwd/$sim &>$cwd/mdrun.$sim \
43                  || { echo "something went wrong, check $cwd/mdrun.$sim"; exit; }
44
45          # save the name of the current simulation step to access it in the next iteration
46          sim_prev=$sim                          # name of next simulations input file
47          wd_prev="$lambdadir/$sim"              # name of next simulations input directory

```

After we ran all the lambda simulations we analyzed the results through the `gmx bar` command. The `gmx bar` command analyzes the output of alchemical simulations and estimates the free energy differences between adjacent λ states using the Bennett Acceptance Ratio (BAR) method. It produces several output files:

- `bar.xvg` — Contains the relative free energy differences $\Delta G_{i \rightarrow i+1}$ between consecutive λ states. This file is useful for identifying which transitions contribute most to the total free energy change.
- `barint.xvg` — Shows the cumulative free energy ΔG as a function of λ , i.e., the integrated free energy change across all windows. The final value in this file corresponds to the total solvation free energy ΔG_{solv} .
- `histogram.xvg` — Provides information on the overlap between sampled states by plotting the energy distribution histograms. Good overlap is crucial for reliable free energy estimates.

3 Results

The computed Gibbs free energy profile exhibits an initial positive value, which gradually decreases and becomes negative as the transformation proceeds. This behaviour qualitatively reflects a favourable overall process. However, the agreement with experimental values is not quantitatively satisfactory. Several factors may contribute to this discrepancy: firstly, the number of intermediate λ states used in the simulation is limited, reducing the resolution of the free energy integration. Secondly, the number of simulation steps per λ is insufficient to ensure proper sampling and convergence. Lastly, electrostatic interactions have been neglected

Free Gibbs Energy analysis for a methane molecule in water

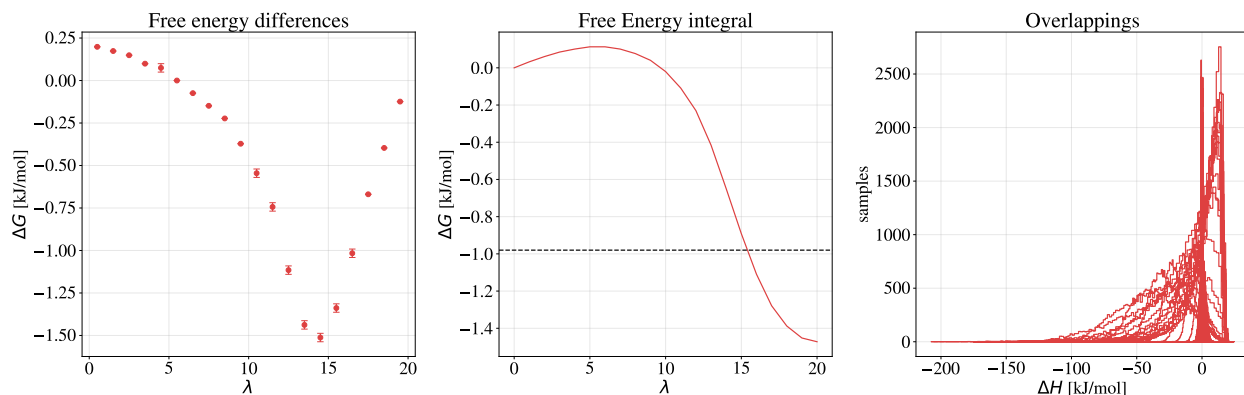


Figure 1: The Gibbs free energy profile initially exhibits positive values and later becomes negative, indicating that the process is overall spontaneous. The integration plot shows a noticeable deviation between the simulated and experimental values, which can be attributed to the approximations discussed earlier. Finally, the enthalpy distributions at various λ values were plotted to assess the degree of overlap. While most distributions overlap well, a few show significant differences. This suggests that increasing the number of λ points could improve the accuracy of the results by enhancing sampling and overlap quality.

in this setup, which can influence the accuracy of the free energy estimation. These limitations highlight the need for a more refined simulation protocol to achieve results in closer agreement with experimental data. The plots are listed below.