Free Energy Calculations

Document version: 13.09.2016

Exercises week 1:	22.11. or 24.11.
Exercises week 2:	29.11. or 01.12.
Deadline for the report:	11.12.
Contact:	`david.hahn@phys.chem.ethz.ch`
	David Hahn - HCI G243

Summary

In this fifth exercise, we will focus on free-energy calculations using molecular dynamics (MD) simulations employing GROMOS. You will move from chemists to alchemists, with the thermodynamic integration (TI) method by calculating the relative hydration free energies of benzene, toluene and phenol and comparing them with experimental data. The focus of this tutorial is on: (*i*) the preparation of the perturbation topologies; (*ii*) the set-up of the series of TI simulations; and (*iii*) the analysis of these simulations to obtain estimates for the free-energy differences.

---

1 Introduction

Under isothermal-isobaric conditions, free-energy changes control the spontaneous evolution of all macroscopic physical, chemical, and biological phenomena, as well as the maximal (reversible) work that can be collected during this evolution. Consequently, it has for a long time been one of the central task of molecular simulation to calculate free-energy differences associated with specific processes, and to understand the cause of these differences at the microscopic level. Free-energy simulations have been applied to numerous processes including phase transitions, mixing, solvation, ligand-receptor binding, macromolecular folding, and many others. In order to obtain accurate free-energy differences, two main challenges have to be met. First, a model for the system (in MD, a classical force field) has to be defined that correctly describes its thermodynamic properties. Second, an efficient scheme must be employed to sample relevant microscopic configurations for all the macroscopic states of interest, and include a sufficient number of transitions between these.

The processes for which free-energy differences can be calculated fall into three main categories: (*i*) thermodynamic processes, in which the states correspond to different values of a specific boundary parameter of the system (e.g. pressure or temperature); (*ii*) conformational processes, in which the states are defined as distinct regions of the configurational space of a system (e.g. folded and unfolded conformations of a macromolecule, free and bound state of a ligand-receptor complex); and (*iii*) alchemical processes, in which the states are defined by different molecular topologies (i.e. Hamiltonian functions) for the same set of particles (e.g. chlorobenzene vs. bromobenzene in water). The "traditional" free-energy calculation methods include pressure- or temperature-integration for the thermodynamic case, direct counting (DC) or umbrella sampling (US) for the conformational case, and thermodynamic integration (TI) or free-energy perturbation (FEP) for the alchemical case. Although it would be fun to try out one example for each category (of change as well as of method), for the sake of time the present tutorial will only consider alchemical changes and the TI method.

We will consider three closely-related molecules, displayed in Figure 1 (top) and very dear to the heart of organic chemists, namely benzene (BNZ), toluene (TOL) and phenol (PHE). Well...
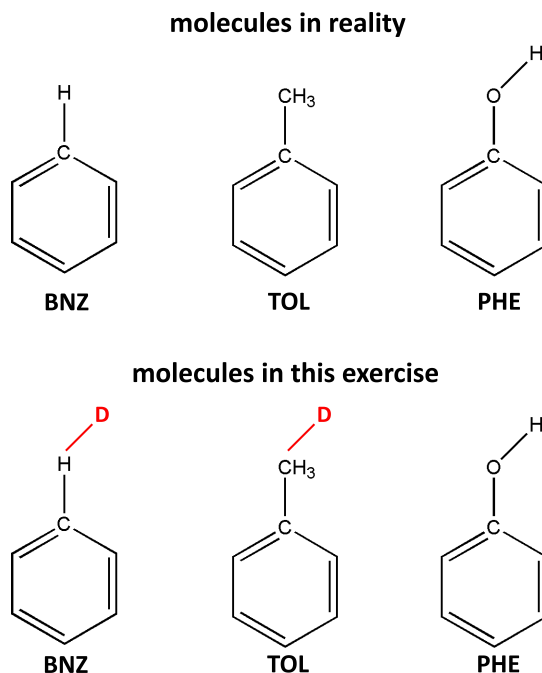

Figure 1: The molecules of benzene (BNZ), toluene (TOL), and phenol (PHE), in reality (top) and as considered in this tutorial (bottom; BNZ and TOL including a dummy atom D).

in fact not exactly. True, we will consider PHE. But this molecule has one more atom (in the GROMOS united-atom representation) compared to BNZ and TOL. So, we will actually consider instead of the true benzene/toluene molecules strange entities consisting of a benzene/toluene with a so-called dummy atom attached to the hydrogen/methyl atom, see Figure 1 (bottom). These weird species will still be labelled BNZ and TOL. The dummy atom is covalently attached to the rest of the molecule, but it has no non-bonded interactions whatsoever (neither intra- nor intermolecular). The choice of intramolecular covalent interactions involving the dummy atom is arbitrary, but must be the same throughout all calculations. Here, for BNZ, we will use everywhere a H-D bond and a C-H-D angle identical to the O-H bond and C-O-H bond-angle in PHE. For TOL, we will do the same for the CH_3-D bond and the C-CH_3-D bond-angle.

Using MD and TI, we will be able to calculate the relative free energies of the three compounds, and we will do this in vacuum (isolated molecules) and in water (hydrated molecules). On their own, the resulting numbers bear no meaning, for three reasons: (i) the GROMOS force field is not engineered to produce the correct quantum-mechanical intramolecular (electronic) energies of molecules; (ii) the mutations involve changes in the atom content of the molecule (alchemy) and have no real-world (chemistry) counter-part; and (iii) the results will be affected by the arbitrary choice of intramolecular interaction parameters for the dummy atoms. So, why should we bother calculating these irrelevant numbers? Simply because all these effects cancel out when we compare the free energies calculated in water to those calculated in vacuum, as illustrated in the thermodynamic cycle of Figure 2, considering the change BNZ→PHE as an example.

The vertical legs of the cycle represent the alchemical mutations BNZ→PHE in vacuum (left) and in water (right). The horizontal legs represent the hydration processes of BNZ (top) and PHE (bottom). Because free energy is a state function, we can write the difference in the
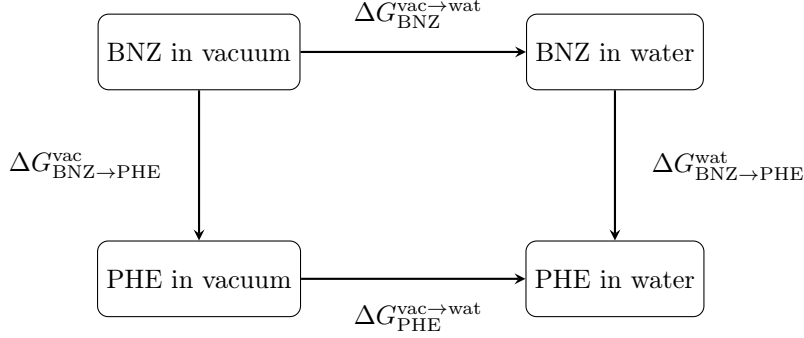
Figure 2: Thermodynamic cycle used to calculate the relative hydration free energies of BNZ and PHE.

hydration free energies of BNZ and PHE as

$$\Delta\Delta G_{\mathrm{BNZ}\to\mathrm{PHE}}^{\mathrm{vac}\to\mathrm{wat}} = \Delta G_{\mathrm{PHE}}^{\mathrm{vac}\to\mathrm{wat}} - \Delta G_{\mathrm{BNZ}}^{\mathrm{vac}\to\mathrm{wat}} = \Delta G_{\mathrm{BNZ}\to\mathrm{PHE}}^{\mathrm{wat}} - \Delta G_{\mathrm{BNZ}\to\mathrm{PHE}}^{\mathrm{vac}} \quad . \tag{1}$$

This approach to calculate the relative free energies of two physical processes using simulations of two alchemical processes is called the thermodynamic-cycle approach. Unlike the alchemical free energies $\Delta G_{\mathrm{BNZ}\to\mathrm{PHE}}^{\mathrm{vac}}$ and $\Delta G_{\mathrm{BNZ}\to\mathrm{PHE}}^{\mathrm{wat}}$, the hydration free energies $\Delta G_{\mathrm{PHE}}^{\mathrm{vac}\to\mathrm{wat}}$ and $\Delta G_{\mathrm{BNZ}}^{\mathrm{vac}\to\mathrm{wat}}$ are physical quantities and could be compared with their experimental counterparts. In particular, the dummy atom in BNZ has no influence on the hydration free energy because it has no non-bonded interactions with the environment (so, in water, it does not "see" the water molecules). However, these physical quantities may be difficult to calculate directly with a high accuracy (we would have to "grow" full molecules from dummy-atom skeletons). The drawback of the thermodynamic-cycle approach is that it only gives access to the difference $\Delta\Delta G_{\mathrm{BNZ}\to\mathrm{PHE}}^{\mathrm{vac}\to\mathrm{wat}}$, not to the individual hydration free energies.

To calculate each of the alchemical free-energy changes, we will use TI, which relies on the statistical-mechanically exact expression

$$\Delta G_{0\to1} = \int_0^1 \mathrm{d}\lambda \left\langle \frac{\partial \mathcal{H}\left(\mathbf{r},\mathbf{p},\mathrm{d}\lambda'\right)}{\mathrm{d}\lambda'} \right\rangle_\lambda \quad , \tag{2}$$

where $\mathcal{H}\left(\mathbf{r},\mathbf{p},\mathrm{d}\lambda\right)$ is the hybrid Hamiltonian, equal to that of the initial state when $\lambda = 0$ and to that of the final state when $\lambda = 1$, and $\langle\cdots\rangle_\lambda$ denotes Boltzmann (trajectory) averaging over configurations generated at given value of λ. We will use a soft-core coupling scheme for the hybrid Hamiltonian, and two approximations will be introduced in Equation 2: (i) the integral will be replaced by a quadrature sum (trapezoidal rule) based on simulations at 11 λ-points; and (ii) the infinite-ensemble averages in the integrand will be replaced by averages over 1 ns simulations (discarding 0.05 ns equilibration).

For the sake of time, you will only be asked to set-up and perform the simulations for the mutation BNZ→PHE in vacuum and in water. For the two other pairs of mutations, BNZ→TOL and TOL→PHE, the simulations have already been set-up and performed for you.

The work for now is described in Part 1. It involves looking at the set-up for the BNZ→TOL and TOL→PHE cases (final files already available), and setting-up the BNZ→PHE case (files to create or modify yourself). The analyses of the generated trajectories are described in Part 2.

2 Week 1

2.1 Simulation Set-up

Before we started, let's login to the beaver cluster and copy a directory that was already prepared for you.

```
cp -r /usr/local/CSCBP/ex5 ~/
```

In the next section we will discuss the set-up of the simulations in water and in vacuum, respectively.

2.1.1 Simulations in Water

There are three mutations involved here. The most logical directions for the TI simulations would be BNZ→TOL, TOL→PHE and BNZ→PHE (left to right in Figure 1, i.e. every time from a "simpler" to a "more complicated" compound), as suggested in Introduction.[1]

When you do TI, you obviously need in some way to have two molecular topologies, one for the initial state (A) and one for the final state (B). There are two different ways to handle this. In the so-called dual-topology implementation, you will set up two full-blown molecular topology files with the same file structure, one for the state A and one for the state B topology. In the so-called single-topology implementation, you only set up a full-blown topology for the state A. And in a separate file, called a perturbation topology file, you will specify the topology changes getting you from topology A to topology B. The single-topology approach is advantageous when B is not very different from A because: (*i*) the perturbation topology file is small; and (*ii*) when you calculate the energies and forces, it is most efficient to calculate the potential energy of state A and then the terms contributing to the difference A-to-B (if you calculated the potential energies of A and B separately, you would calculate twice all the common force-field terms!). For these reasons and because GROMOS is smart, it relies on a single-topology implementation.[2]

As already stated in Introduction, the TOL→BNZ and PHE→TOL cases are already completely handled. All the files are in the directories: `TI_TOL_BNZ` and `TI_PHE_TOL`, respectively. The case BNZ→PHE is left for you to handle, in directory `TI_PHE_BNZ`.

Now let's have a look at the existing files, starting with the PHE→TOL case.

```
cd ~/ex5/topo
```

Have a look at the topolgy file `phe.top` and convice yourself that this is indeed the appropriate topology file for a single PHE molecule (i.e. if you did not do a free-energy calculation, you could use this file directly for a standard GROMOS simulation of PHE). To help you in this check, have a look at Figure 3 (right column) where the topological information on PHE is reported graphically.

```
cd ~/ex5/TI_PHE_TOL
```

Then have a look at the perturbation topology file `phe_to_tol.ptp` (displayed below), where the changes between the state A and the state B topologies are listed.

---

[1] We have actually set up everything for the reverse mutations, i.e. TOL→BNZ, PHE→TOL and PHE→BNZ.

[2] We have set-up the reverse mutations, simply because the A-state of the two latter mutations, PHE, is a normal molecule without dummy atoms, for which a topology building block is already available in GROMOS. In other words, we can set up the A-state topology for each of these two mutations with a single `make_top` command.

```
TITLE
perturbation topology, phenol to toluene
END
PERTATOMPARAM
# number of perturbed atoms
   3
#
#   NR RES NAME IAC(A)  MASS(A)  CHARGE(A) IAC(B) MASS(B) CHARGE(B)   ALJ   ACRF
    11   1   C6     12   12.011      0.203     12  12.011      0.00     0     0
    12   1   O6      3   15.9994    -0.611     16  15.035      0.00     1     1
    13   1   H6     21    1.008      0.408     22   1.008      0.00     1     1
END
PERTBONDSTRETCH
# number of perturbed bonds
    1
#   atom(i) atom(j) bond_type(A) bond_type(B)
         11      12           13           27
END
```

The `PERTATOMPARAM` block specifies the perturbed atoms. `NR` is the sequence of the atom, `RES` is the sequence of the residue, `NAME` is the name of the atom, `IAC(A)` is the integer atom code in state A, `MASS(A)` is the mass of the atom in state A, `CHARGE(A)` is the atomic partial charge of the atom in state A, `IAC(B)` is the integer atom code in state B, `MASS(B)` is the mass of the atom in state B, `CHARGE(B)` is the atomic partial charge of the atom in state B. Finally, `ALJ` and `ACRF` are scaling factors for the soft-core parameters involved in Lennard-Jones and Coulombic interactions, respectively (see further below). The `PERTBONDSTRETCH` block specifies the perturbed bonds, in which `atom(i)` and `atom(j)` are the sequences of the two atoms defining the bond, while `bond_type(A)` and `bond_type(B)` are the bond types in states A and B, respectively. Have a look again at Figure 3 (middle and right columns), and verify that the changes listed in the file reflect the expected topology changes when going from PHE to TOL. Note in particular that, as discussed in Introduction, the hydroxyl hydrogen atom of PHE has not been deleted but transformed into a dummy atom (atom type 22 in GROMOS).

Next, have a look at the GROMOS input file `TI.inp` (displayed below). By now, you certainly know the structure of these files by heart, but... there is a new type of block here, the `PERTURBATION` block

```
PERTURBATION
#    NTG: 0..1 controls use of free-energy calculation.
#         0: no free-energy calculation (default)
#         1: calculate dH/dRLAM
#  NRDGL: 0,1 controls reading of initial value for RLAM.
#         0: use initial RLAM parameter from PERTURBATION block
#         1: read from configuration
#   RLAM: 0.0..1.0 initial value for lambda
#  DLAMT: >= 0.0 rate of lambda increase in time.
# ALPHLJ: >= 0.0 Lennard-Jones soft-core parameter
#  ALPHC: >= 0.0 Coulomb-RF soft-core parameter
#   NLAM: > 0 power dependence of lambda coupling
# NSCALE: 0..2 controls use of interaction scaling
#         0: no interaction scaling
```

```
#           1: interaction scaling
#           2: perturbation for all atom pairs with scaled
#              interactions. No perturbation for others.
#
#     NTG    NRDGL    RLAM    DLAMT
         1        0     0.0      0.0
#   ALPHLJ    ALPHC    NLAM   NSCALE
       0.5      0.5       1        0
END
```

`NTG` set to one switches on the perturbation code in GROMOS for a TI simulation, `RLAM` is the value of the λ in the specific TI simulation (if `NRDGL` was set to one, the value would be read from the initial configuration file instead), `DLAMT` is set to zero (so λ will be kept constant at `RLAM` throughout the simulation), while `ALPHLJ` and `ALPHC` are the soft-core parameters for Lennard-Jones and Coulombic interactions, respectively. Soft-core interactions are used to avoid singularities when creating or deleting atoms (more precisely: when converting them from or to dummy atoms). The soft-core interaction will only be applied for Lennard-Jones and Coulombic interactions, respectively, to atoms for which the `ALJ` and `ACRF` parameters, respectively, differ from 0 in the perturbation topology (see above). Note that `NRDGL` and `DLAMT` are useful if you do slow growth (an alternative form of TI where you "sweep" λ from 0 to 1 over a run or a series of runs; but TI at a series of fixed λ-points is more accurate). `NLAM` and `NSCALE` allow for more variations in the form of the coupling scheme, which are not relevant for us here.

Finally, have a look at the `mk_script` job-list file `joblist_TI.dat`. Herein, we specify 11 different folders for the simulations at the 11 successive λ-points in water. For each λ-point, 25000 steps (0.05 ns) of equilibration and 500000 steps (1 ns) of sampling will be carried out.

Now let's have a look at the second set of existing files, corresponding to the TOL→BNZ case.

`cd ∼/ex5/topo`

And take a look at the state-A topology file `tol.top` to convince yourself that this is indeed the appropriate topology file for a single TOL molecule (including a dummy atom) by comparison with Figure 3 (middle column).

`cd ∼/ex5/TI_TOL_BNZ`

Then have a look at the perturbation topology file `tol_to_bnz.ptp`. In addition to the blocks you already saw in the previous perturbation, you will find a new block `PERTBONDANGLE` for the perturbed bond angles. `Atom(i)`, `atom(j)`, and `atom(k)` are the atoms defining the perturbed angle, while `type(A)` and `type(B)` are the bond angle types in state A and B, respectively.

```
PERTBONDANGLE
# number of perturbed bond angles
    2
#    atom(i) atom(j) atom(k) type(A) type(B)
          7       11       12      27      25
          9       11       12      27      25
END
```

By comparison with Figure 3 (left and middle columns), verify that the changes listed in the file (`PERTATOMPARAM`, `PERTBONDSTRETCH` and `PERTBONDANGLE` blocks) reflect the expected
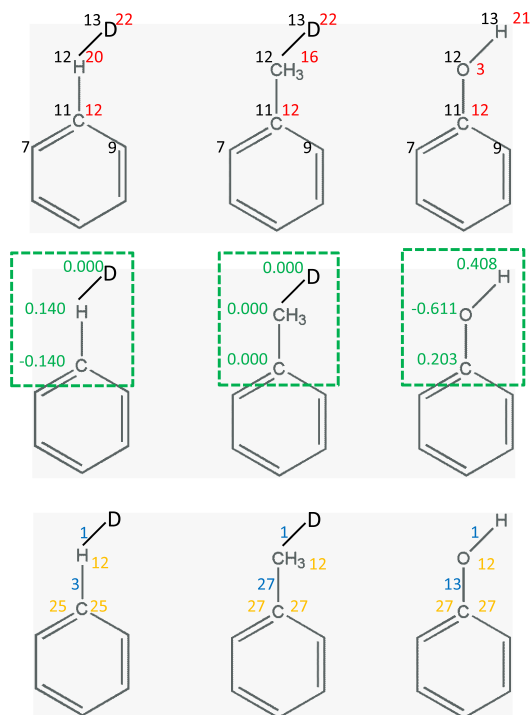
Figure 3: Relevant parameters in the topologies of BNZ, TOL, and PHE. Note that 5 explicit hydrogen atoms on the benzene rings are not indicated. Upper panel: sequence numbers of the atoms (black) and integer atom codes (red). Middle panel: atomic partial charges (green) and charge-group boundary (dashed line). Lower panel: bond types (blue) and bond-angle types (orange).

topology changes when going from TOL to BNZ. In particular can you figure out why the `PERTBONDANGLE` block is needed here, whereas it was not in the previous perturbation?

Note that there are other blocks which are not needed here, but could be used in other types of perturbations. For example, a `PERTIMPROPERDIH` block is used for perturbed improper dihedrals, a `PERTPROPERDIH` block is used for perturbed dihedrals, a `PERTPOLPARAM` block is used for perturbed polarizable atoms, and a `MPERTATOM` block is used for enveloping distribution sampling (EDS) simulations.

The GROMOS input file `TI.inp` and the `mk_script` job-list file `joblist_TI.dat` are exactly identical to the files used in the previous perturbation, so no need to look at them. Then, what is left for us is to set-up the PHE→BNZ mutation in water. So let's roll up our sleeves take a deep breath and start working.

When you feel mentally ready for this big step, just type

```
cd ~/ex5/TI_PHE_BNZ
```

Due to time constraints we will skip the molecular topology preparation steps today, but in a free-energy calculation, the procedure for state A is exactly the same as for a plain MD simulation. The topology and coordinates files are directly provided to you in the `topo` and `coord` directories, respectively. The `mk_script` job-list file `joblist_TI.dat` is also the same as in the two previous perturbations, so it is already prepared.

You first task is to complete the perturbation topology file `phe_to_bnz.ptp` (displayed below).

7

```
TITLE
TO_DO
END
PERTATOMPARAM
# number of perturbed atoms
    3
#
#   NR RES NAME IAC(A)  MASS(A)  CHARGE(A) IAC(B) MASS(B) CHARGE(B)   ALJ   ACRF
    TO_DO
END
PERTBONDSTRETCH
# number of perturbed bonds
    1
#   atom(i) atom(j) bond_type(A) bond_type(B)
    TO_DO
END
PERTBONDANGLE
# number of perturbed bond angles
    2
#   atom(i) atom(j) atom(k) type(A) type(B)
    TO_DO
END
```

The number of perturbed atoms, bonds, and bond angles are already given. Based on Figure 3, and what you have learned from the examples in the previous two TI simulations, finalize the perturbation topology file.

The second task is to complete the parameter input file `TI.inp` by replacing the `TO_DO` with the `PERTURBATION` block. In this case, you only need to insert in the following `PERTURBATION` block with the correct parameters.

```
PERTURBATION
#    NTG: 0..1 controls use of free-energy calculation.
#         0: no free-energy calculation (default)
#         1: calculate dH/dRLAM
#  NRDGL: 0,1 controls reading of initial value for RLAM.
#         0: use initial RLAM parameter from PERTURBATION block
#         1: read from configuration
#   RLAM: 0.0..1.0 initial value for lambda
#  DLAMT: >= 0.0 rate of lambda increase in time.
# ALPHLJ: >= 0.0 Lennard-Jones soft-core parameter
#  ALPHC: >= 0.0 Coulomb-RF soft-core parameter
#   NLAM: > 0 power dependence of lambda coupling
# NSCALE: 0..2 controls use of interaction scaling
#         0: no interaction scaling
#         1: interaction scaling
#         2: perturbation for all atom pairs with scaled
#            interactions. No perturbation for others.
#
#     NTG    NRDGL    RLAM    DLAMT
        ?        ?       ?        ?
```

```
#   ALPHLJ    ALPHC    NLAM  NSCALE
       ?         ?        ?       ?
END
```

The last step is to finalize the `mk_script_TI.arg` file. In the `@sys` argument, give a name to your first TI simulation (and keep it in mind, because you need to use this name several times later). And there are several other entries you need to take care of as well.

```
@sys      TO_DO
@joblist TO_DO
@bin      /usr/local/gromos-1.3.2/bin/md_mpi
@dir      $PWD
@files
        topo          ../topo/phe.top
        input         TO_DO
        coord         ../coord/phe.cnf
        pttopo        TO_DO
@template   ../lib/mk_script.4.procs.lib
@version md++
```

Once you set up everything correctly, you can use `mk_script` to generate the job files for TI simulations.

```
mk_script @f mk_script_TI.arg
```

Now you should have eleven folders called `L_*` in your directory. Look into these folders!!!

To submit your TI simulation, open the file `job_submit.sh`, and change `TO_DO` to the name of your simulation, which should be the same as what you put in as `@sys` argument in the `mk_script_TI.arg` file.

```
TI=TO_DO

cd L_0.0
qsub -N ${TI}_1 -cwd -j y -o ${TI}_1 -pe orte 4 ./${TI}_1.run
```

Check whether your simulation is running properly on the cluster using

```
qstat
```

If so, you can move to the next step. If not, don't swear, keep the faith, and just find out what is wrong (in this tutorial, failure is not an option!).

2.1.2   Vacuum Simulations

In order to complete the thermodynamic cycle shown in Figure 2, three TI simulations involving the same perturbations need to be carried out in vacuum. As for the aqueous situation, the TOL→BNZ and PHE→TOL cases are already completely handled in directories `TI_vacuum_TOL_BNZ` and `TI_vacuum_PHE_TOL`, respectively. The case BNZ→PHE is left for you to handle in directory `TI_vacuum_PHE_BNZ`. Look inside the completed folders, e.g.

```
cd ~/ex5/TI_vacuum_PHE_TOL
```

When you consider a single small molecule in vacuum, it is often unwise to use MD with a Berendsen thermostat. There are two reasons for this. First, the Berendsen thermostat maintains the correct average temperature, but does not lead to rigorously correct temperature fluctuations. This is generally no problem for systems with hundreds of degrees of freedom or more, but can lead to artifacts when considering a single molecule with very few atoms. Second, small systems tend to be closer to being "harmonic", and being trapped in long-timescale quasi-harmonic oscillations may impair the sampling. We can solve the problem of representing a molecule in vacuum by considering an "ideal gas" consisting of many copies of the molecules at very large distances from each other, and all coupled to the same Berendsen thermostat (as applied in the simulations in water). Here, we will use an alternative approach: consider a single molecule, but replace the MD/Berendsen simulations by stochastic dynamics (SD) simulations. In SD simulations, random kicks and frictional forces are introduced, which has the effect of both randomizing (no long-timescale quasi-harmonic motions) and thermostating (temperature determined by the balance between random and frictional forces) the system.

In the GROMOS input file `TI.inp`, have a look at the `STOCHDYN` block. `NTSD` set to one switches on the stochastic dynamics, `CFRIC` sets the friction coefficient, and `TEMPSD` is the temperature of the stochastic bath. The magnitude of the stochastic kicks will be automatically determined from `CFRIC` and `TEMPSD` in such a way that we generate a canonical ensemble at temperature `TEMPSD`. Note that the value of `CFRIC` influences the dynamics of the system but not its thermodynamics (and thus the free-energy results). Here we arbitrarily took a value mimicking friction in water (the only constraint is that the stochastic coupling should be strong enough to kill possible quasi-harmonic motions in the molecule).

```
STOCHDYN
# NTSD    0,1 controls stochastic dynamics mode
#         0: do not do stochastic dynamics (default)
#         1: do stochastic dynamics
# NTFR    0..3 defines atomic friction coefficients gamma
#         0: set gamma to 0.0 (default)
#         1: set gamma to CFRIC
#         2: set gamma to CFRIC*GAM0
#         3: calculate gamma using subroutine FRIC (based on CFRIC)
# NSFR    > 0 recalculate gamma every NSFR steps
# NBREF   > 0 threshold number of neighbour atoms for a buried atom
# RCUTF   >= 0.0 interatomic distance considered when calculating gamma
# CFRIC   >= 0.0 global weighting for gamma
# TEMPSD  >= 0.0 temperature of stochastic bath
#
#     NTSD     NTFR     NSFR    NBREF   RCUTF    CFRIC    TEMPSD
         1        1        1        6     0.3     91.0     300.0
END
```

As in the Section 2.1.1, you are asked to set up the TI simulations for the perturbation PHE→BNZ in vacuum by yourself. So, go to the folder

```
cd ~/ex5/TI_vacuum_PHE_BNZ
```

Here, you have to repeat what you did for the corresponding TI simulation in water.[3] Pay special attention to the `STOCHDYN` and `PERTURBATION` blocks in the parameter input file `TI.inp`.

---

[3]Be smart before doing hard work... E.g. for the perturbation topology, do you think there will be any difference between the file for a perturbation in vacuum and the one for the perturbation in water?

```
STOCHDYN
# NTSD    0,1 controls stochastic dynamics mode
#         0: do not do stochastic dynamics (default)
#         1: do stochastic dynamics
# NTFR    0..3 defines atomic friction coefficients gamma
#         0: set gamma to 0.0 (default)
#         1: set gamma to CFRIC
#         2: set gamma to CFRIC*GAM0
#         3: calculate gamma using subroutine FRIC (based on CFRIC)
# NSFR    > 0 recalculate gamma every NSFR steps
# NBREF   > 0 threshold number of neighbour atoms for a buried atom
# RCUTF   >= 0.0 interatomic distance considered when calculating gamma
# CFRIC   >= 0.0 global weighting for gamma
# TEMPSD  >= 0.0 temperature of stochastic bath
#
#     NTSD     NTFR     NSFR    NBREF   RCUTF    CFRIC     TEMPSD
        ?        ?        ?        ?       ?        ?          ?
END
PERTURBATION
#    NTG: 0..1 controls use of free-energy calculation.
#         0: no free-energy calculation (default)
#         1: calculate dH/dRLAM
#  NRDGL: 0,1 controls reading of initial value for RLAM.
#         0: use initial RLAM parameter from PERTURBATION block
#         1: read from configuration
#   RLAM: 0.0..1.0 initial value for lambda
#  DLAMT: >= 0.0 rate of lambda increase in time.
# ALPHLJ: >= 0.0 Lennard-Jones soft-core parameter
#  ALPHC: >= 0.0 Coulomb-RF soft-core parameter
#   NLAM: > 0 power dependence of lambda coupling
# NSCALE: 0..2 controls use of interaction scaling
#         0: no interaction scaling
#         1: interaction scaling
#         2: perturbation for all atom pairs with scaled
#            interactions. No perturbation for others.
#
#     NTG    NRDGL     RLAM    DLAMT
        ?        ?        ?        ?
#  ALPHLJ    ALPHC     NLAM   NSCALE
        ?        ?        ?        ?
END
```

After submitting, check whether your simulation is running properly on the cluster by

```
qstat
```

3 Week 2

3.1 Analyses

First of all, let's check whether all went smoothly with your simulations. Go in turn to the folders `TI_PHE_BNZ` and `TI_vacuum_PHE_BNZ`, and try

`ls *`

Did your simulations finish successfully? If not, try to find out why or/and contact the assistants (committing seppuku might be considered as a very last resort!). If all is well, you can move on to the analysis part.

In the TI method, the alchemical free-energy differences are calculated using Equation 2. During the TI simulations, $\partial\mathcal{H}/\partial\lambda$ was calculated and written out to the energy trajectory files (*.tre.gz) as well as to the GROMOS output files (`*.omd`). So the analysis involves the following steps: (i) extract the values of $\partial\mathcal{H}/\partial\lambda$ from the energy trajectory files and calculate the corresponding averages and statistical uncertainties using the GROMOS++ program `ene_ana` for all the simulations of different λ values; (ii) carry out the integral of the averages over λ using numerical quadrature (trapezoidal rule), and estimate the associated overall uncertainty.

We have already prepared a script `analyse.sh` which does these tasks automatically for all six simulations. Have a look at it.

```
cd ~/ex5/scripts/analyse.sh
vi analyse.sh
```

Try to understand how the script works. For every line in `dirfile`, which is given as an argument to the script, it changes to the specified directory (first column in `dirfile`) and newly creates a directory `ana` (if it already exists, it will be deleted first (!)). In this directory `ana`, two other scripts are executed: Firstly `jdhdl.sh`, which analyses the trajectories of the simulations and secondly `integrate_plot.sh`, which integrates the average $\partial\mathcal{H}/\partial\lambda$ values and creates plots afterwards.

First have a look at `jdhdl.sh`. You should recognize a part running `ene_ana` for the second trajectory of each λ value and therefore does not take into account of the equilibration period and a part gathering the averages and uncertainties which a then written to a file called `ave_dhdl.dat`. Then, script `integrate_plot.py` takes over, carries out the integration of both the averages and uncertainties (trapezoidal quadrature), interpolates the data with cubic splines and integrates these afterwards, too. The results are then appended to the file `ave_dhdl.dat`. A plot of the average $\partial\mathcal{H}/\partial\lambda$ values plus their statistical uncertainties, its splines and the two running integrals as a function of λ is generated. If you have any questions about these scripts, do not hesitate to ask the assistants. What you still need to do is to put the name of the directory, the name of the corresponding simulation (`@sys` argument in the `mk_script_TI.arg` file in Section 2.1.1) and the desired title of the plots (no blanks, only underscores to separate words) to the file `~/ex5/dir_list.txt`. A first line is given as an example. Please fill ou the `TO_DO`s.

Now run the script using

```
cd ~/ex5
./scripts/analyse.sh dir_list.txt
```

In every `ana` directory, ou will see some outputs from the `ene_ana` program, and the plot of $\partial\mathcal{H}/\partial\lambda$ as a function of λ (average and estimated statistical uncertainties). Check the `ave_dhdl.dat` file.

Here, you will find the final estimate for the free-energy difference, along with the estimated overall statistical uncertainty. What value do you get? You could also do the integration using `xmgrace`, and it is actually a nice exercise to try this out.

```
xmgrace -settype xydy ave_dhdl.dat
```

On the `xmgrace` plot, go to the menu Data/Transformations/Integration, then click on Accept. What value do you get? Now you can see one red curve on the plot, which is the running integral.

We could also do a spline fitting of the curve before the integration. Click on the menu Data/Transformations/Interpolation/splines and set the parameters according to Figure 4.
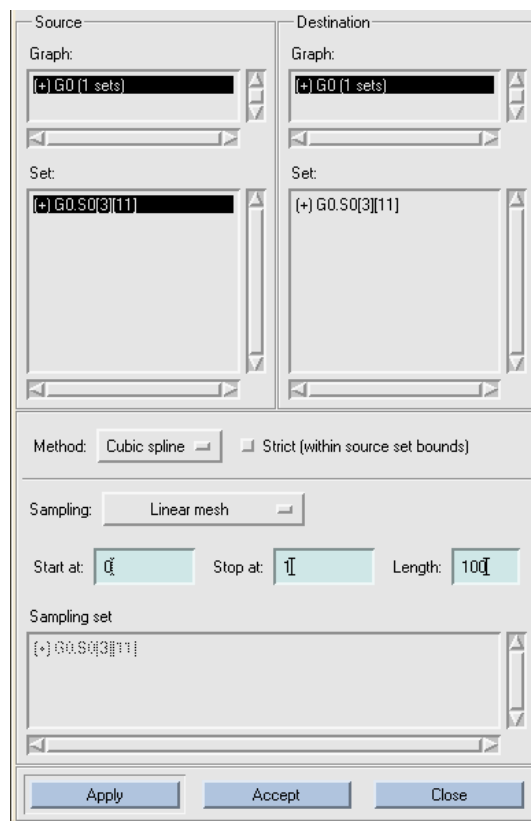

Figure 4: `xmgrace` setting for spline fitting

Now, we can do the integration of the fitted (smooth) curve by clicking on the menu Data/Transformations/Integration, selecting the set S1, and then clicking on Accept. What value do you get now?

Collect the results and statistical error estimates for the six free-energy changes considered. What about modifying the script `analyse.sh` so that it writes out the results of all six free-energy changes into one file? When you have all of them, you can compare the values against Table 1.

4 Report

4.1 Format

Please refer to the corresponding section of Exercise 1 for information on the goal and expected structure of the report.

4.2 Simulation Results

Answer the following questions:

A. Look at the GROMOS output files you generated (`*.omd` files). Which terms contribute to $\partial\mathcal{H}/\partial\lambda$? Are the contributing terms different in vacuum and in water?

B. Plot the average $\partial\mathcal{H}/\partial\lambda$ curves for the TI simulations, along with the running integral of the curve.

C. Compare the integration results obtained from two different approaches: (1) trapezoidal integration; (2) trapedzoidal integration after spline fitting. Comment on the differences.

   Note that from here on, use the free energy values obtained from trapezoidal integration.

D. Give all the six values of the alchemical free-energy differences (including estimated error bars) using the template shown in Figure 5.

in vacuum — Benzene, Toluene, Phenol

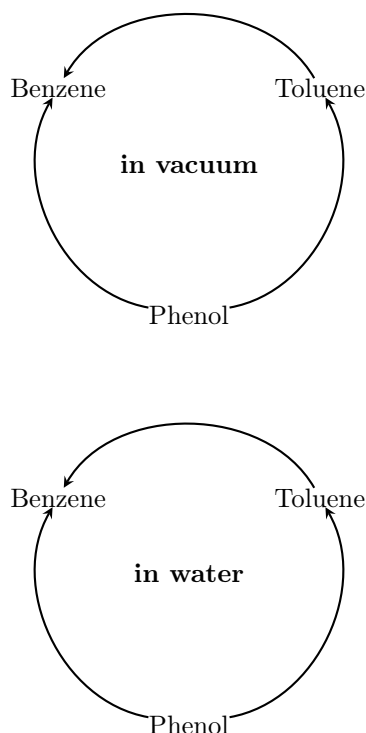**in water** — Benzene, Toluene, Phenol

Figure 5: Template for free energy results.

E. Do the thermodynamic cycles in vacuum and in water close? (i.e. zero free-energy change around the cycle, within the error bars).

F. Calculate the hydration free energies of PHE and TOL relative to that of BNZ using the results of the PHE→BNZ and TOL→BNZ mutations and Eq. 1. Are the results compatible with the relative hydration free energy of PHE relative to TOL calculated based on the PHE→TOL mutations?

G. Compare these calculated relative hydration free energies with the experimental values provided in Table 1. Do the experimental differences make sense chemically? Are these differences reproduced in the calculated numbers?

4.3 Thinking Questions

Answer the following questions:

1. In this exercise, you calculated relative hydration free energies (e.g. of TOL relative to BNZ). What would you do if you needed to calculate an absolute hydration free energy (e.g. for BNZ)? Why would the corresponding calculation require more effort to get converged results (compared to that of a relative hydration free energy)?

2. Now imagine you want to calculate the absolute hydration free energies of 100 benzene derivatives in an efficient way (more efficient than 100 times the procedure of question!). How would you do it ?

3. When you calculate an absolute hydration free energy and want to compare with experiment, you have to worry about standard states. In simulations, the value characterizes the transfer of a molecule from a fixed point in vacuum to a fixed point in water. In experiment, standard data refers to the transfer from an ideal gas at 1 bar to an infinitely dilute solution extrapolated to 1 mol solute per kg solvent. Can you infer the form of the standard-state correction term you must apply to the simulated value before comparing with experiment? And can you explain why we don't need to worry about this when comparing relative hydration free energies to experiment?

4. For the simulations in vacuum, we used SD to overcome possible harmonic oscillation and improper thermostating problems. But in Exercise 1, we instead used a collection of molecules at large distances with MD/Berendsen. Could we also have applied a similar approach here?

5. In the three perturbations considered in this exercise, we never discussed the situation of improper dihedral angles and of torsional dihedral angles. In principle, the perturbations might affect the improper dihedral controlling the planarity at the C bearing the substituent (H, CH_3 or O). Why is it not the case? Similarly, PHE has a dihedral angle controlling the orientation of the OH group. What happens with this one when moving to BNZ or TOL?

6. We have carried out the perturbations in the reverse direction (TOL→BNZ, PHE→TOL and PHE→BNZ) because then, the A-state of the two latter mutations, PHE, is a "normal" molecule without dummy atom which is directly available as a building block in GROMOS. We could have benefitted from the same advantage by setting up the TOL→BNZ mutation as converting a "normal" TOL to a "normal" BNZ (i.e. no dummy atoms involved). (1) Can you explain why the final results in terms of hydration free energies would have been the same; (2) what about the thermodynamic cycles in vacuum and in water (question E above), would they still have closed exactly?

7. One might introduce a sort of "hierarchy" in the types of perturbations converting a molecule A into a molecule B: (1) the perturbation is chemically acceptable (i.e. it could have an experimental counterpart); (2) the perturbation is alchemical (i.e. it has no experimental counterpart) but involves no dummy atoms; (3) the perturbation is alchemical and requires using dummy atoms. What are the conditions on the molecules A and B for the perturbation to be in category (1), (2) or (3). Why is the result of a calculation using GROMOS still generally incorrect for category (1), i.e. only the difference of such results calculated in different environments (e.g. vacuum vs. water) is likely to be correct?

8. There would in principle be no need for the dummy atom to be attached to the same site as the atom to appear, or even to be attached to the molecule at all. Why is it still wise to covalently attach the dummy to the molecule, and to do this where the new atom will appear?

9. And here is a last one for the real hard-thinkers. In Section 1, we boldly stated that "the choice of intramolecular covalent interactions involving the dummy atom is arbitrary" and that it does not really matter "because all these effects cancel out when we compare the free energies calculated in water to those calculated in vacuum". In fact, this is not entirely true. Can you explain why (think of the lecture discussion about free-energy components), think of a (pathological) counterexample, and formulate a more accurate condition of the allowed form of covalent interactions for the dummy atoms?

Appendix A. Experimental Data

Table 1: Experimental relative hydration free energies [1] of TOL and PHE relative to BNZ. .

Compound	Relative Hydration Free [kJ mol^{-1}]
BNZ	0.0
TOL	-0.1
PHE	-24.0

Appendix B. List of files

In the main exercise directory, you will find `ex5.pdf`, the digital version of the document you are reading and nine sub-directories:

topo/ Contains the topology files for phenol and methylbenzene.

coord/ Contains the coordinate files for phenol and methylbenzene in both vacuum and water box.

lib/ Contains the library file for the GROMOS++ program mk_script.

TI_TOL_BNZ/ Contains the TI simulation from toluene to benzene.

TI_PHE_TOL/ Contains the TI simulation from phenol to toluene.

TI_PHE_BNZ/ Contains some files required for the TI simulation from phenol to benzene.

TI_vacuum_TOL_BNZ/ Contains the TI simulation from toluene to benzene in vacuum.

TI_vacuum_PHE_TOL/ Contains the TI simulation from phenol to toluene in vacuum.

TI_vacuum_PHE_BNZ/ Contains some files required for the TI simulation from phenol to benzene in vacuum.

References

[1] Mobley, D.L., Bayly, C.I, Cooper, M.D., Shirts, M.R., and Dill, K.A. *J. Chem. Theory. Comput* **2009** 350-358