GROMACS Tutorial 6 - Excess chemical potential of methane using test particle insertion

In this tutorial we'll be using test particle insertion (TPI) to calculate the excess chemical potential of methane solvation in water. Most users are unaware that GROMACS has a built-in method for running TPI. This tutorial will not be a comprehensive discussion on the statistical mechanics of TPI, but will address issues when needed. The user is encouraged to seek out scientific resources regarding this method.

TPI involves perturbing some state to some other, very similar state. We will be taking bulk water and insertng a methane particle and measuring the potential energy change from this. There is a statistical mechanical relationship between this change in potential energy and the excess chemical potential. For us, state A is the bulk water system, and state B is the water system with a methane.

With GROMACS you need to run state A as a normal MD simulation. We already did this for our case of bulk water in Tutorial 1 (../1-tip4pew-water/#tutorials-gromacs-1-tip4pew-water). We'll reuse the output trajectory files for inserting the methane.

Setup

Create water system

Follow Tutorial 1 to run a system containing TIP4PEW water.

Add test particle to topology file

Our original topology file just had water. In the new topology file we simply need to add 1 test particle, and it needs to be the last molecule in the system. We'll use `opls_066` for the particle's atom type which is OPLS's united atom methane. Here's what my final topology file looks like (the number of waters will be different for your system):

```
#include "oplsaa.ff/forcefield.itp"
#include "oplsaa.ff/tip4pew.itp"

[ moleculetype ]
; Name          nrexcl
Methane         3

[ atoms ]
;   nr       type      resnr residue  atom   cgnr     charge       mass
     1       opls_066  1     CH4      C      1          0     16.043


[ System ]
Methane in water

[ Molecules ]
SOL               395
Methane           1
```

Add test particle to gro file

You also need to add the test particle to the gro file. Simply edit `conf.gro` (or any of the other `.gro` files uses) and add a line at the end containing the test particle's position (right before the box coordinates). The line I added looks like this:

```
396CH4      C 1581   0.000   0.000   0.000
```

The actual position doesn't matter; GROMACSS just wants a placeholder for the test particle. Additionally you need to add 1 to the total number of particles in the system on the second line of the `.gro` file.

Parameter files

We only need one parameter file for TPI. Simply copy `prd.mdp` from your bulk water simulation and change `integrator` to `tpi`. You should change `nsteps` to the number of insertions per frame that you want to attempt. I chose `100000` steps for my simulation. You will also need to change `cutoff-scheme` to `group`, since `Verlet` has not be implemented for TPI.

Simulation

For the simulation we are just rerunning the bulk water simulation using the saved trajectory file (which was named `prd.xtc` in the first tutorial). To do this first run `grompp`:

```
$ gmx grompp -f mdp/tpi.mdp -o tpi -po tpi -pp tpi -c conf.gro
```

Now use the `-rerun` flag with `mdrun`:

```
$ gmx mdrun -deffnm tpi -rerun prd.xtc
```

Analysis

The log file file, in this case named `tpi.log`, contains a line with the average volume and the average excess chemical potential. My two lines looked like this:

```
<V>  =  1.18704e+01 nm^3
<mu> =  8.81230e+00 kJ/mol
```

`<mu>` is output in kJ/mol, but if we convert it to kcal/mol we get 2.106 kcal/mol. This is in line with our results from the free energy of solvation done in Tutorial 4 (../4-methane-fe/#tutorials-gromacs-4-methane-fe) using the lambda-coupling method where I got 2.289 kcal/mol. The difference can be attributed to the usage of an all-atom model with the free energy of solvation simulations and a united-atom model in this case.

Summary

In this tutorial we looked at how to use GROMACS to perform test particle insertion in order to get the excess chemical potential of a united-atom OPLS methane.

Back to top