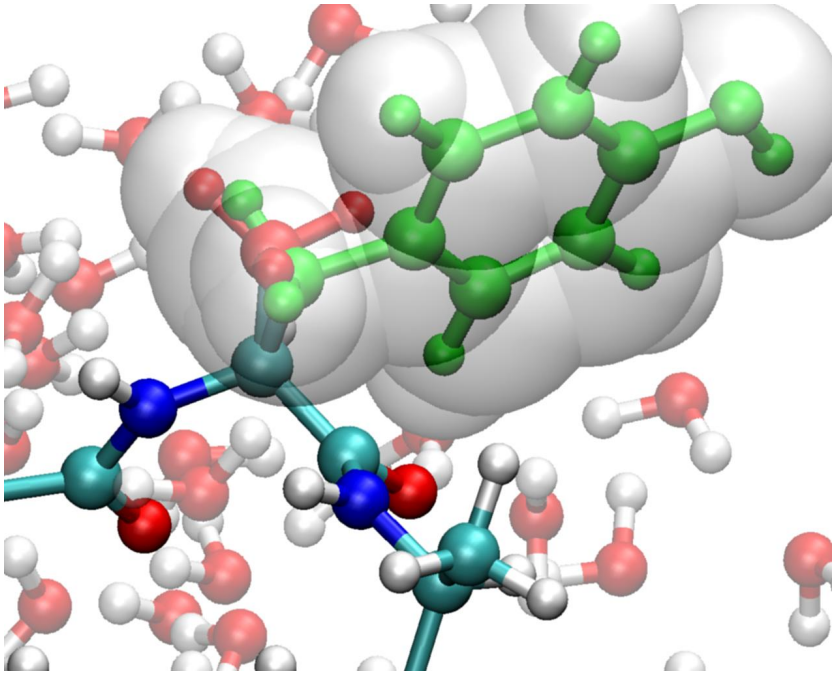
Nancy Université, Université Henri Poincaré
Centre National de la Recherche Scientifique
Équipe de dynamique des assemblages membranaires,

Centre National de la Recherche Scientifique
Laboratoire d'Ingénierie des Systèmes Macromoléculaires

University of Illinois at Urbana-Champaign
Beckman Institute for Advanced Science and Technology
Theoretical and Computational Biophysics Group

In silico alchemy: A tutorial for alchemical free-energy perturbation calculations with NAMD



Jérôme Hénin
James Gumbart
Christopher Harrison
Christophe Chipot

June 29, 2010

Please visit www.ks.uiuc.edu/Training/Tutorials/ to get the latest version of this tutorial, to obtain more tutorials like this one, or to join the `tutorial-l@ks.uiuc.edu` mailing list for additional help.

Abstract

This tutorial explains how NAMD and related tools can be used to setup and perform alchemical free-energy simulations within the free-energy perturbation (FEP) theory. The force-field independent, "zero-sum" transformation of ethane into ethane is used as an introductory, prototypical example. FEP is then used to compute the free energy of charging a naked Lennard-Jones particle into a sodium ion. Next, the variation in solvation free energy upon mutation of a tyrosine residue into alanine is examined in the Ala–Tyr–Ala tripeptide. Last, the concept of standard binding free energy is illustrated in the simple case of a potassium ion binding a ionophore, 18–crown–6. Prior knowledge of NAMD and standard molecular dynamics simulations is assumed.

Contents

Introduction

The goal of this tutorial is to provide a guidance when setting up free energy calculations of alchemical transformations [1] within NAMD. [2, 3] We will first perform the rather simple, "zero-sum" transformation of ethane into ethane in water. In a second case example, the free energy of charging a naked Lennard-Jones sphere into a sodium ion is determined, recovering in a computer simulation a result predicted over eighty years ago by the Born model. [4] Last, as a more practical example, we will compute the difference in hydration free energy resulting from the mutation of tyrosine into alanine in the Ala–Tyr–Ala tripeptide. As has been commented on amply, such *in silico* experiments have not reached yet the maturity to be viewed as black-box, routine jobs, [5, 6, 7] which implies that both the sampling strategy and the analysis of the results should be considered with great care.

The paradigm chosen in NAMD for performing alchemical transformations is the so-called *dual topology* approach, [8, 9] in which both the initial state, *viz.* $\lambda = 0$, and the final state, *viz.* $\lambda = 1$, are defined concurrently. As the molecular dynamics (MD) simulation progresses, the potential energy function characteristic of $\lambda = 0$ is scaled into that representative of $\lambda = 1$. Whereas the initial and the final states do interact with the environment, they do not see each other in the course of the transformation. For versions of NAMD prior to 2.7b2, achieving these conditions requires that a list of excluded atoms be defined in the PSF topology file. Since the `psfgen` software supplied with NAMD does not offer a way of building such a list, we provide the `alchemify` program. `alchemify` processes PSF files written by `psfgen` or CHARMM and makes them suitable for simulating alchemical transformations. However, in NAMD 2.7b2, the appropriate exclusions are generated automatically at the start of the simulations.

The reader of this tutorial is assumed to be familiar with the use of NAMD to perform "standard" calculations, including energy minimization and MD simulations. General documentation, tutorials and templates of NAMD configuration files are available from the Documentation section of the NAMD web page. The simulation times in this tutorial are very short in the interest of expediency; however, one should be aware that they are generally insufficient for most applications. Additionally, although not always used in the tutorial, equilibration of a system prior to running FEP simulations is highly recommended.

Completion of this tutorial requires:

- various files contained in the archive `FEP_tutorial.zip`, provided with this document;
- NAMD 2.7b2 [3] or later (`http://ks.uiuc.edu/Research/namd`);
- VMD 1.8.4 [10] or later (`http://ks.uiuc.edu/Research/vmd`);
- `alchemify` (only needed for versions of NAMD before 2.7b2 `http://www.edam.uhp-nancy.fr/Alchemify`).

1. Ethane-to-ethane "zero-sum" transformation

Perhaps the simplest alchemical transformation one could imagine, the result of which is completely independent of the potential energy function utilized, is the *zero-sum* ethane → ethane mutation, [11, 9] wherein a methyl group vanishes at one end of the molecule, while another one appears at the other end. The accuracy of the computed free energy only depends on the sampling strategy adopted, regardless of the force field employed.

1.1. System setup

In this first step, we will build a structure file (PSF) using a manually defined topology, and a set of atomic coordinates (in PDB format) for the hybrid molecule and water. Necessary files are gathered in the `ethane-ethane` subdirectory of the archive. In this first example, Cartesian coordinates for water are provided, so that the reader may get started quickly with free energy simulations. In the second part of this tutorial, use of the VMD plugin `solvate` will be made to prepare a hydrated system.

1.1.1. Generating the PSF file

As can be seen in Figure 1, the hybrid defined for this transformation is a pseudo-propane molecule, consisting of a juxtaposition of two ethane fragments, with a common $–CH_2–$ moiety.

The CHARMM topology file (`zero.top`, provided) for the hybrid ethane molecule reads:

```
* Topology for ethane-to-ethane transformation
27 1    ! pretend we are CHARMM27_1
```

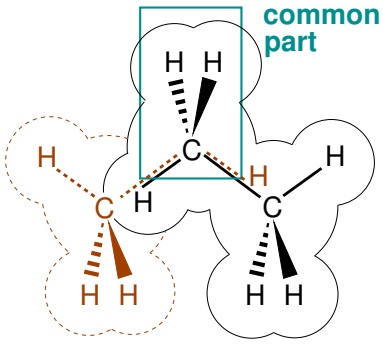


Figure 1: Dual-topology hybrid molecule used for the *zero-sum* ethane → ethane alchemical transformation. The initial state, *viz.* $\lambda = 0$ (black), and the final state *viz.* $\lambda = 1$ (brown), are defined concurrently. The central $–CH_2–$ moiety is common to both topologies.

```
RESI ZERO        0.00       !  ethane -> ethane
GROUP                       !
ATOM CI   CT3   -0.27       !
ATOM HI1  HA     0.09       !
ATOM HI2  HA     0.09       !
ATOM HI3  HA     0.09       !
GROUP                       !   HI1       HM1   HF2  HF3
ATOM CM   CT3   -0.27       !      \       |     |  /
ATOM HM1  HA     0.09       !       \HF    |     | /
ATOM HM2  HA     0.09       !        CI----CM----CF
ATOM HI   HA     0.09       !       / |    |     HI\
ATOM HF   HA     0.09       !      /  |    |        \
GROUP                       !   HI2  HI3  HM2        HF1
ATOM CF   CT3   -0.27       !
ATOM HF1  HA     0.09       !
ATOM HF2  HA     0.09       !
ATOM HF3  HA     0.09       !
BOND     CI  HI1       CI  HI2       CI   HI3       ! ethane 1
BOND     CF  HF1       CF  HF2       CF   HF3       ! ethane 2
BOND     CI  CM        CF  CM                       ! common
BOND     CM  HM1       CM  HM2                      ! common
BOND     CM  HI                                     ! ethane 1
BOND     CM  HF                                     ! ethane 2

! No patching
PATCHING FIRST NONE LAST NONE
END
```

In this transformation, the methyl group `CI-HI1-HI2-HI3` is replaced by hydrogen atom `HF`, while the methyl group `CF-HF1-HF2-HF3` replaces the hydrogen atom `HI`.

The PSF may be prepared using the following `psfgen` script (`setup.pgn`, not provided):

```
topology ../common/top_all22_prot.inp
topology zero.top

segment ZERO { residue 1 ZERO }
segment WAT  { auto none; pdb water.pdb }

coordpdb ethane.pdb ZERO
```

```
coordpdb water.pdb  WAT

writepsf setup.psf
writepdb setup.pdb
```

Executing the script with the command line `psfgen setup.pgn` (or by using `source setup.pgn` in the VMD TkConsole) creates a PDB and a PSF file. Example output is provided.

1.1.2. *Preparing the alchFile*

One should remember that, on account of the dual-topology paradigm, both the initial state and the final states of the alchemical transformation are present simultaneously. It is, therefore, pivotal that the information about the nature of these states be passed to NAMD, indicating which atoms of the hybrid molecule correspond to $\lambda = 0$, and which correspond to $\lambda = 1$. This information is included in the `alchFile`, a file written in the PDB format, in which a -1.0 or 1.0 flag characterizes those atoms of the hybrid molecule that, respectively, vanish or appear in the course of the simulation. A 0.0 flag is assigned to those atoms that are left unchanged as λ varies from 0 to 1. The `alchFile` for the ethane-to-ethane transformation, (`zero.fep`, not provided), is easily prepared by editing a copy of `setup.pdb`. The beginning of the file `zero.fep` should read:

```
ATOM      1  CI  ZERO    1      -1.167   0.224   0.034  1.00 -1.00      ZERO
ATOM      2  HI1 ZERO    1      -2.133  -0.414   0.000  1.00 -1.00      ZERO
ATOM      3  HI2 ZERO    1      -1.260   0.824   0.876  1.00 -1.00      ZERO
ATOM      4  HI3 ZERO    1      -1.258   0.825  -0.874  1.00 -1.00      ZERO
ATOM      5  CM  ZERO    1       0.001  -0.652  -0.002  1.00  0.00      ZERO
ATOM      6  HM1 ZERO    1       0.000  -1.313  -0.890  1.00  0.00      ZERO
ATOM      7  HM2 ZERO    1       0.005  -1.308   0.889  1.00  0.00      ZERO
ATOM      8  HI  ZERO    1       1.234   0.192   0.000  1.00 -1.00      ZERO
ATOM      9  HF  ZERO    1      -1.237   0.190   0.000  1.00  1.00      ZERO
ATOM     10  CF  ZERO    1       1.289   0.150  -0.078  1.00  1.00      ZERO
ATOM     11  HF1 ZERO    1       2.149  -0.425  -0.001  1.00  1.00      ZERO
ATOM     12  HF2 ZERO    1       1.256   0.837  -0.893  1.00  1.00      ZERO
ATOM     13  HF3 ZERO    1       1.131   0.871   0.940  1.00  1.00      ZERO
ATOM     14  OH2 TIP3    1      -5.574  -5.971  -9.203  1.00  0.00      WAT
ATOM     15  H1  TIP3    1      -5.545  -5.020  -9.301  1.00  0.00      WAT
...
```

The flag that distinguishes between “growing” and “shrinking” atoms will be read from the B column by default. In this case, atoms `CI`, `HI1`, `HI2`, `HI3` and `HI` of the initial state vanish, while atoms `CF`, `HF1`, `HF2`, `HF3` and `HF` of the final state appear.

Visual inspection in VMD

At this stage, visualizing the system with its initial and final groups may prove very useful to detect possible errors in the previous steps. Run VMD with the following command: `vmd setup.psf -pdb zero.fep`. In the Graphics/Representations menu, set the selection text to "not water" and select the coloring method `Beta`. Appearing atoms are colored blue and vanishing atoms are colored red, while the unperturbed part of the molecule appears in green. Compare the result with Figure 1.

1.1.3. *Cleaning up the PSF file (Optional if using NAMD 2.7b2 or later)*

In the system as we have defined it, appearing and vanishing atoms interact in two ways: through non–bonded forces, and because some of the angle and dihedral parameters automatically generated by `psfgen` couple the two end-points of the transformation. For instance, the `CI-CM-CF` angle should not be assigned a force field term by NAMD. To prevent unwanted non–bonded interactions, `alchemify` processes PSF files and creates the appropriate non–bonded exclusion list. At the same time, irrelevant bonded terms that involve appearing and vanishing atoms are removed. `alchemify` retrieves the necessary data about the transformation from `alchFile`. This step is unnecessary when using NAMD 2.7b2 or later as these unwanted terms are automatically ignored.

It should be called using the following command line:

```
alchemify setup.psf zero.psf zero.fep
```

The resulting file `zero.psf` will be used when performing the FEP calculations.

1.2. Running the free energy calculation

Now that the we have built a PSF file containing a suitable description of the hybrid molecule, we will detail how the free energy calculation proceeds in NAMD.

The traditional MD section of the NAMD configuration file should be written for an MD run at a constant temperature of 300 K and pressure of 1 bar, using particle-mesh Ewald (PME) electrostatics. Set the `rigidBonds` option to `all` and choose a time step of 2 fs. To impose isotropic fluctuations of the

periodic box dimensions, set the `flexibleCell` variable to `no`. Define the initial periodic box as a cube with an edge length of 21.8 Åor use the provided restart files from an equilibration run.

TCL scripts allow to set up the protocol of the free energy calculation in a straightforward fashion. The file `fep.tcl` defines three commands that simplify the syntax of FEP scripts:

- `runFEP` runs a series of FEP windows between equally-spaced λ-points, whereas
- `runFEPlist` uses an arbitrary, user-supplied list of λ values.
- `runFEPmin` is identical to runFEP but adds an initial minimization step as well.

We will use the following sampling strategy for the forward transformation:

```
# FEP PARAMETERS

source                  ../tools/fep.tcl

alch                    on
alchType                FEP
alchFile                zero.fep
alchCol                 B
alchOutFile             forward-noshift.fepout
alchOutFreq             10

alchVdwLambdaEnd        1.0
alchElecLambdaStart     1.0
alchVdWShiftCoeff       0.0
alchDecouple            no

alchEquilSteps          100
set numSteps            500

runFEP 0.0 1.0 0.0625 $numSteps
```

In the above example, the potential energy function of the system is scaled from $\lambda = 0$ to $\lambda = 1$ by increments $\delta\lambda = 0.0625$, *i.e.* 16 intermediate λ-states or "windows". [7] In each window, the system is equilibrated over `alchEquilSteps` MD steps, here 100 steps, prior to `$numSteps` - `alchEquilSteps` = 400 steps of data collection, from which the ensemble average is evaluated.

Run the forward and backward simulations using, e.g., the commands

```
namd2 forward-noshift.namd > forward-noshift.log
namd2 backward-noshift.namd > backward-noshift.log
```

in a terminal window. Check your local installation of NAMD and run it accordingly. Each simulation should require only a few minutes.

1.3. Results

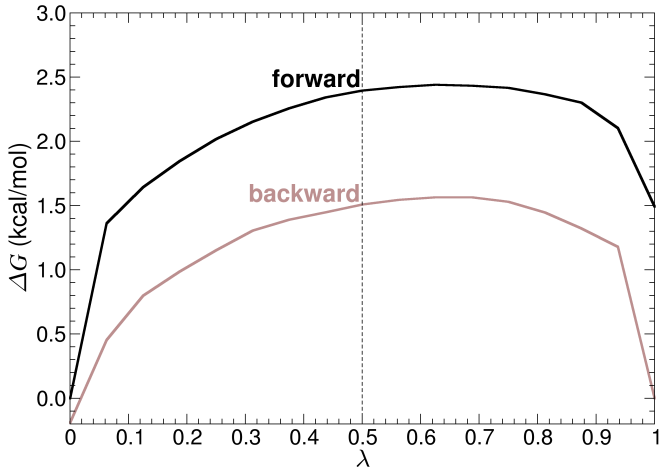


Figure 2: Free-energy change for the ethane → ethane *zero-sum* mutation, measured in the forward and in the backward direction. This simulation was performed in the absence of a soft-core potential. [12, 13] Singularities are manifested at the end points, when $\lambda = 0$ or 1.

The `alchOutFile` file contains all data resulting from the FEP calculation. On a UNIX system, a simple way to extract the $\Delta G(\lambda)$ profile is to use this command line:

```
grep change alchemy.alchOutFile | awk '{print $9, $19}' > alchemy.dat
```

This creates a two-column data file that can be read by most plotting programs, *e.g.* xmgrace, gnuplot, or any spreadsheet application. Alternatively, one can use the provided script `deltaA.awk` in the `tools` directory, i.e.,

```
../tools/deltaA.awk forward-noshift.fepout > forward-noshift.dat
```

to analyze the data.

An example of the expected result is plotted in Figure 2. One of the notorious shortcomings of the dual-topology paradigm can be observed in the $\Delta G(\lambda)$ profile when λ approaches 0 or 1. In these regions, interaction of the reference or the target topology with its environment is minute, yet strictly nonzero. Molecules of the surroundings can in turn clash against incoming or outgoing chemical moieties, which is conducive to numerical instabilities in the trajectory, manifested in large fluctuations in the average potential energy and, hence, slow convergence issues. [7] These so-called "end-point catastrophes" can be attenuated significantly, employing a better-adapted sampling strategy. They can also

be essentially circumvented through the use of a soft-core potential, [12, 13, 14] which effectively eliminates the singularities at $\lambda = 0$ or 1. This feature is available by default in NAMD. For pedagogical purposes, however, the parameters that control the soft-core potential — *viz.* `alchVdwLambdaEnd`, `alchElecLambdaStart`, `alchVdWShiftCoeff` and `alchDecouple`, were set in the above example in such a way that no correction was applied. In other words, to highlight the deleterious effects of possible end-point singularities, no shift in the van der Waals potential was introduced in the present simulation. In what follows, a soft-core potential will be used systematically to prevent such singularities to occur.

1.4. Why a soft-core potential should always be included

In the following FEP script, a soft-core potential is employed, obviating the need of narrower windows as λ gets closer to 0 or 1.

```
# FEP PARAMETERS

source                  ../tools/fep.tcl

alch                    on
alchType                FEP
alchFile                zero.fep
alchCol                 B
alchOutFile             forward-shift.fepout
alchOutFreq             10

alchVdwLambdaEnd        1.0
alchElecLambdaStart     0.5
alchVdWShiftCoeff       6.0
alchDecouple            yes

alchEquilSteps          100
set numSteps            500

runFEP 0.0 1.0 0.0625 $numSteps
```

The parameters utilized for the soft-core potential can be understood as follows. Outgoing atoms will see their electrostatic interactions with the environment decoupled from $\lambda = 0$ to $1 -$ `alchElecLambdaStart` $= 0.5$, while the interactions involving incoming atoms are progressively coupled from $\lambda = 1 -$ `alchElecLambdaStart` to 1.

At the same time, van der Waals interactions involving vanishing atoms are progressively decoupled

from $\lambda = 1 - \texttt{alchVdwLambdaEnd}$, *i.e.* 0 to 1, while the interactions of appearing atoms with the environment become coupled from $\lambda = 1$ to $1 - \texttt{alchVdwLambdaEnd}$, *i.e.* 0.

Run the new forward and backward simulations using, e.g., the commands `namd2 forward-shift.namd > forward-shift.log`
`namd2 backward-shift.namd > backward-shift.log`
in a terminal window and analyze the resulting output.

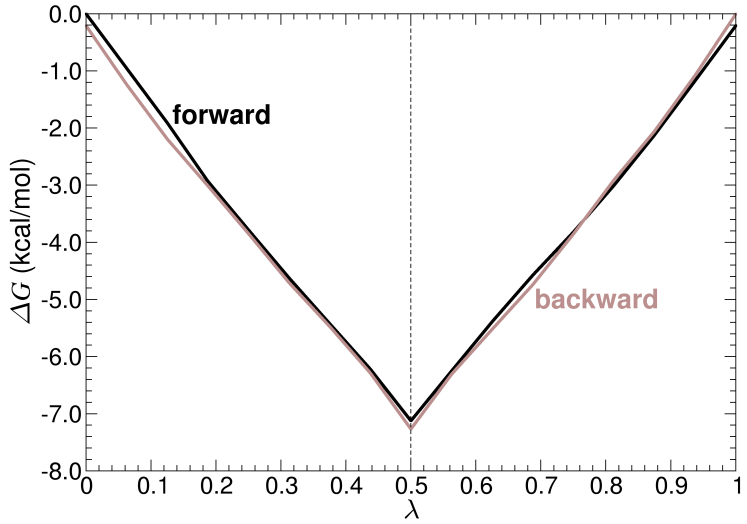


Figure 3: Result of an ethane → ethane *zero-sum* mutation including a soft-core potential correction to circumvent the "end-point catastrophes" highlighted in Figure 2. Using an identically modest sampling strategy, the expected naught free-energy change is nicely recovered. The essentially overlapping profiles obtained from the forward and the backward transformations ought to be noted.

Figure 3 shows a typical result for this new simulation. The effect of the soft-core potential is magnified in the total free energy change, now close to zero, and the symmetry of the profile with respect to $\lambda = 0.5$, which altogether suggests that the calculation has converged. However, greater sampling will improve convergence further. If desired, try running the simulation `forward-shift-long.namd`, which multiplies the length by a factor of 10, to see if it improves the result (NOTE: this simulation may take 30 minutes or more!).

Probing the convergence properties of the simulation

Convergence of the free energy calculation can be assessed by monitoring the time-evolution of $\Delta G(\lambda)$ for every individual λ-state and the overlap of configurational ensembles embodied in their density of states, $P_0[\Delta U(\mathbf{x})]$ and $P_1[\Delta U(\mathbf{x})]$, where $\Delta U(\mathbf{x}) = U_1(\mathbf{x}) - U_0(\mathbf{x})$ denotes the difference in the potential energy between the target and the reference states. Since a soft-core potential [12, 13] has been introduced to avoid the so-called "end-point catastrophes", the potential energy no longer varies linearly with the coupling parameter λ. It is, therefore, necessary that the reverse transformation be carried out explicitly to access $\Delta G_{\lambda+\delta\lambda\rightarrow\lambda}$, as is proposed in the above NAMD scripts.

2. Charging a spherical ion

In the second example of this tutorial, charging of a naked Lennard-Jones particle into a sodium ion is considered in an aqueous environment.

2.1. System setup

The system consists of a sodium ion immersed in a bath of water molecules. In the framework of the dual-topology paradigm, charging a Lennard-Jones particle is tantamount to shrinking a naked spherical particle, while growing concomitantly the ion. In this particular example, the single-topology approach would have the benefit of perturbing only the electrostatic component of the nonbonded potential, thus, avoiding perturbing the Lennard-Jones terms and improving convergence. We will see that it is possible to emulate such a single-topology paradigm within the dual-topology approach implemented in NAMD, simply by choosing appropriate parameters for the soft-core potential and the coupling of the particle with its environment.

2.1.1. Generating the PSF file

The first step in the generation of the PSF file is the definition of the sodium ion, which uses the `SOD` atom type. Using `psfgen`, load the standard CHARMM topology. The initial Cartesian coordinates are set to $\{0, 0, 0\}$. The ion can then be hydrated simply by employing the `Solvate` plugin of VMD, yielding

a new pair of files (PDB and PSF). It is recommended that the primary cell be large enough to minimize the self-interaction of the cation between adjacent boxes. A box length of 30 Å, *viz.* approximately 823 water molecules, represents a reasonable compromise. The distance separating the cation in the primary and the adjacent cells being equal to 30 Å, the interaction energy reduces to $q^2/4\pi\epsilon_0\epsilon_1 r$ = 0.1 kcal/mol with an ideal macroscopic permittivity of 78.4 for bulk water.

The following `psfgen` script can be used to generate the initial psf and pdb files for the ion and the corresponding uncharged particle.

```
topology ../common/top_all22_prot.inp

segment SOD {
  residue 1 SOD
}

writepsf setup.psf
writepdb setup.pdb
```

The particles can then be solvated using the `Solvate` plugin of VMD, for example, by running the following in the TkConsole:

```
solvate setup.psf setup.pdb -t 15 -o solvate
```

2.1.2. *Preparing the alchFile*

The `alchFile` is a replica of the PDB file, wherein the B column has been altered to indicate which atoms are vanishing or appearing.

```
ATOM      1 SOD  SOD     1       0.000   0.000   0.000  1.00  1.00       SOD
...
```

If the transformation consists in charging the naked Lennard-Jones sphere, atom `SOD` ought to be grown — *i.e.* 1. As a safety check, it is recommended to consider both forward, *i.e.* charge creation, and backward, *i.e.* charge annihilation, transformations.

In the present system, since `SOD` does not coexist with any other perturbed particle, no spurious extra term in the PSF file ought to be removed utilizing `alchemify`.

2.2. Running the free energy calculation

An interesting feature of the NAMD implementation of the soft-core potential [12] lies in the possibility to decouple at a different pace the van der Waals and the electrostatic interactions of the perturbed system with its environment, as depicted in Figure 4. It is evident from the latter that the electrostatic component can be modified independently from the van der Waals counterpart, which is precisely what reversible charging of the Lennard-Jones particle requires.

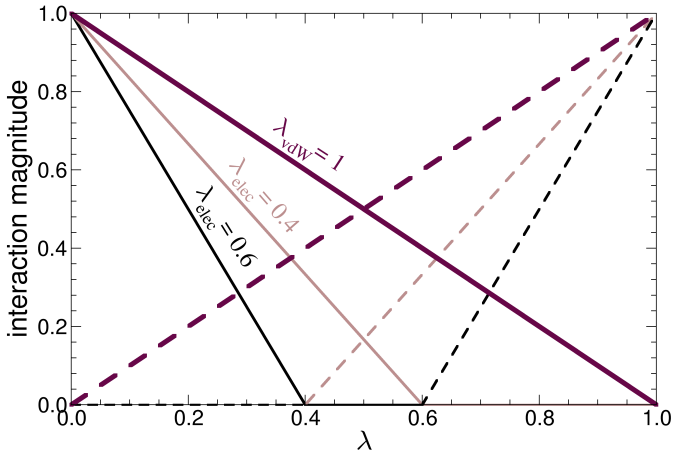


Figure 4: Decoupling of electrostatic and van der Waals interactions within the NAMD implementation of the soft-core potential. Two variables define how the perturbed system is coupled or decoupled from its environment, *viz.* λ_{elec} (`alchElecLambdaStart`) and λ_{vdW} (`alchVdwLambdaEnd`). In the present scenario, one value is considered for λ_{vdW}, namely 1.0, which means that the van der Waals interactions for outgoing and incoming particles will be, respectively, scaled down from 1.0 $-$ λ_{vdW} = 0.0 to 1.0, and scaled up from 0.0 to λ_{vdW}. If λ_{elec} = 0.4, electrostatic interactions for outgoing and incoming particles are, respectively, scaled down from 0.0 to 1.0 - λ_{elec} = 0.6, and scaled up from λ_{elec} to 1.0.

By setting both `alchElecLambdaStart` and `alchVdwLambdaEnd` to 0.0, the electrostatic interactions will be scaled down from 0.0 to 1.0, or, symmetrically, scaled up from 0.0 to 1.0, while van der Waals interactions remain unchanged.

```
# FEP PARAMETERS

source                  ../tools/fep.tcl
```

```
alch                    on
alchType                FEP
alchFile                solvate.fep
alchCol                 B
alchOutFile             forward.fepout
alchOutFreq             10

alchVdwLambdaEnd        0.0
alchElecLambdaStart     0.0
alchVdWShiftCoeff       5.0
alchDecouple            on

alchEquilSteps          100
set numSteps            500

set numMinSteps         100

runFEPmin 0.0 1.0 0.0625 $numSteps $numMinSteps $temp
```

This sampling strategy involves 16 equally spaced windows with $\delta\lambda$ = 0.0625. Each window features 500 steps of MD sampling, among which 100 steps are equilibration. Assuming a time step of 2 fs for integrating the equations of motion, the total simulation time amounts to 16 ps. As will be shown in what follows, this time scale is appropriate for reproducing the expected charging free energy.

Standard NAMD configuration parameters will be used for this alchemical transformation. Langevin dynamics will be employed to maintain constant the temperature at 300 K. The Langevin piston will enforce a constant pressure of 1 bar. Long-range electrostatic forces will be handled by means of the PME algorithms. All chemical bonds will be frozen to their equilibrium value and a time step of 2 fs will be used. To impose isotropic fluctuations of the periodic box dimensions, `flexibleCell` will be set to `no`. The initial periodic cell, prior to equilibration, will be defined as a cube with an edge length of 30.0 Å. Because the system needs to be minimized first, the command `runFEPmin` is used instead of `runFEP`.

As before, run both the forward and backward simulations using the provided `*.namd` files and analyze them to get the charging free energy.

2.3. Results

The free energy profile delineating the alchemical transformation is shown in Figure 5. This profile can be obtained following the same protocol described in the first example of this tutorial.

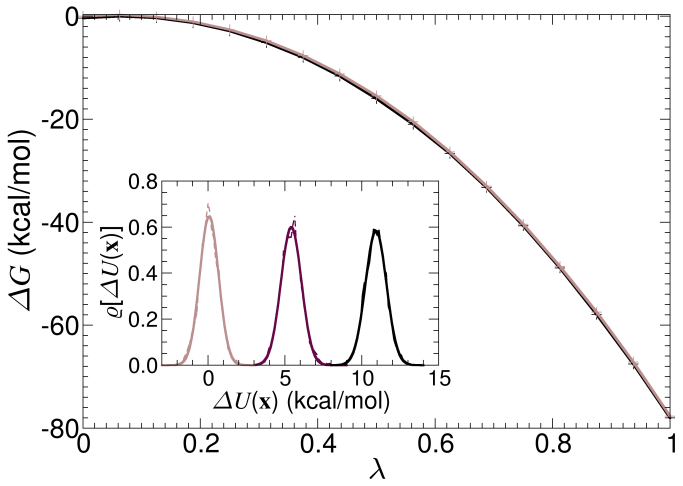


Figure 5: Free energy change for the charging of a naked Lennard-Jones particle into a sodium ion. Inset: Distribution of $\Delta U(x)$ at different values of λ (indicated by arrows on the free energy curve), with Gaussian fits.

A glance at Figure 5 reveals that the free energy of charging the spherical particle in water is considerable, amounting to -77.6 kcal/mol. As is, this estimate is incomplete, because it ignores the size-dependence of the system — *i.e.* ΔG is expected to vary with the size of the primary cell. [15] In the case of an Ewald summation-like simulation, with a cubic periodic simulation box, the size-dependence correction is equal to $+1/2\ \xi_{\mathrm{Ewald}}\ (q_1^2 - q_0^2)$, where q_0 and q_1 are the charges for the reference and the target states — *i.e.* +1 and 0, and $\xi_{\mathrm{Ewald}} = -2.837/L$, where L is the length of the cell.

A periodic cell length of L = 29.1 Å yields a size-dependence correction of -16.2 kcal/mol. Added to the raw charging free energy, the corrected estimate amounts to -93.8 kcal/mol, in reasonable agreement with the value of -96.8 kcal/mol obtained by Hummer *et al.*, [15] using a different set of Lennard-Jones parameters for sodium — *viz.* R_{ii}^* = 1.425 Å and ε_{ii} = 0.04793 kcal/mol, and with a box of 256 SPC water molecules. As a basis of comparison, the complete free energy of ionic hydration measured experimentally by Marcus is equal to -87.2 kcal/mol. [16] At the theoretical level, Straatsma and Berendsen, [17] using a simulation box containing 216 water molecules, estimated this free energy to -121.4 kcal/mol. It is apparent from these different results that the charging free energy is very sensitive to the force field parameters utilized.

The density of states for a selection of intermediate λ-states are depicted in the inset of Figure 5. From the onset, it is apparent that these distributions obey a normal law, thereby suggesting that second-order perturbation theory is sufficient to describe with a reasonable accuracy the charging process. [7]

Improving the agreement with experiment

Replacing the standard CHARMM Lennard-Jones parameters for sodium by those chosen by Hummer *et al.*, [**?**] verify that the charging free energy coincides with the estimate reached by these authors.

Recovering predictions from the Born model

Granted that water is a homogeneous dipolar environment, the charging free energy writes $\Delta G = -\beta/2\ q^2\ \langle V^2 \rangle_0$, where V is the electrostatic potential created by the solvent on the charge, q. Increasing q to $+2$, show that the charging free energy varies quadratically with the charge, as predicted by the Born model — *i.e.* $\Delta G = (\epsilon - 1)/\epsilon \times q^2/2a$, where a is the radius of the spherical particle and ϵ, the macroscopic permittivity of the environment.

3. Mutation of tyrosine into alanine

The free energy change involved in the point mutation of the N- and C-terminally blocked Ala–Tyr–Ala tripeptide can be estimated using the thermodynamic cycle of Figure 6. [18] The quantities $\Delta G^1_{\text{alch.}}$ and $\Delta G^2_{\text{alch.}}$ are obtained through two FEP simulations of the mutation, one *in vacuo* and the other in bulk water. This rather rudimentary case represents an affordable example of how point mutations in more complex protein systems may be studied.

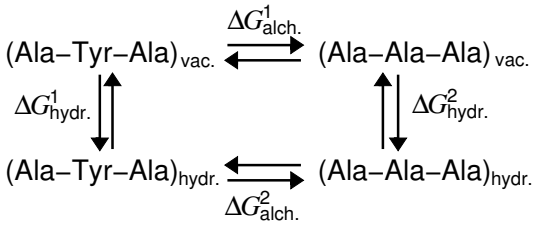


Figure 6: Thermodynamic cycle used in the Ala–Tyr–Ala $\rightarrow$ $(\text{Ala})_3$ alchemical transformation. The vertical arrows correspond to the hydration of the wild-type tripeptide and its mutant. The horizontal arrows correspond to the point mutation in bulk water and *in vacuo*, so that: $\Delta G^2_{\text{alch.}} - \Delta G^1_{\text{alch.}} = \Delta G^2_{\text{hydr.}} - \Delta G^1_{\text{hydr.}}$

3.1. System setup

Two systems have to be prepared: the isolated tripeptide, and the same solvated in explicit water. The latter can be built based on the former, using the `solvate` plugin of VMD. The files required to get

started with this setup can be found in the `03.Mutating-tyrosine-into-alanine` subdirectory of the archive.

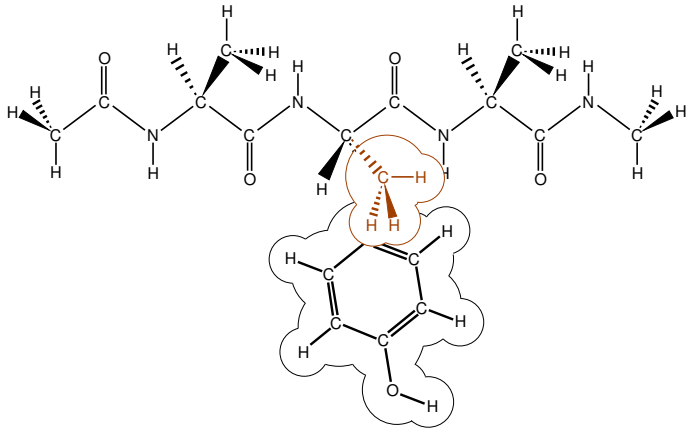


Figure 7: Dual topology hybrid molecule used for the Ala–Tyr–Ala → $(Ala)_3$ alchemical transformation. The initial state, *viz.* $\lambda = 0$ (black), and the final state *viz.* $\lambda = 1$ (brown), are defined concurrently. Apart from the two side chains, the chemical groups of the tripeptide are common to the two topologies.

3.1.1. *Hybrid CHARMM topology*

The hybrid topology for the tripeptide is depicted in Figure 7. The topology file `tyr2ala.top` is based on the standard CHARMM topologies for alanine and tyrosine. A topology file containing hybrid amino acids for all point mutations (except those involving proline) are available with the VMD plugin Mutator, available with VMD 1.8.5 and later. Recent releases of VMD include a database of hybrid topologies compatible with the CMAP corrections [19] of the CHARMM force field. The Mutator plugin may be used to prepare hybrid protein topologies and coordinates suitable for alchemical FEP. The manual procedure that we follow here, however, is much more flexible, should one want to include specific patches in the structure.

```
* Topology for tyrosine-to-alanine transformation
27 1    ! Version: pretend we are CHARMM27b1

RESI Y2A          0.00
GROUP
ATOM N    NH1    -0.47
ATOM HN   H       0.31
ATOM CA   CT1     0.07
ATOM HA   HB      0.09
GROUP
ATOM CBA  CT2    -0.18
ATOM HB1A HA      0.09
ATOM HB2A HA      0.09
GROUP
ATOM CGA  CA      0.00
GROUP
```

```
ATOM CD1A CA     -0.115
ATOM HD1A HP      0.115
GROUP
ATOM CE1A CA     -0.115
ATOM HE1A HP      0.115
GROUP
ATOM CZA  CA      0.11
ATOM OHA  OH1    -0.54
ATOM HHA  H       0.43
GROUP
ATOM CD2A CA     -0.115
ATOM HD2A HP      0.115
GROUP
ATOM CE2A CA     -0.115
ATOM HE2A HP      0.115
GROUP
ATOM CBB  CT3    -0.27
ATOM HB1B HA      0.09
ATOM HB2B HA      0.09
ATOM HB3B HA      0.09
GROUP
ATOM C    C       0.51
ATOM O    O      -0.51
BOND N    HN
BOND N    CA
BOND C    CA
BOND C    +N
BOND CA   HA
BOND O    C
BOND CBA  CA
BOND CGA  CBA
BOND CD2A CGA
BOND CE1A CD1A
BOND CZA  CE2A
BOND OHA  CZA
BOND CBA  HB1A
BOND CBA  HB2A
BOND CD1A HD1A
BOND CD2A HD2A
BOND CE1A HE1A
BOND CE2A HE2A
BOND OHA  HHA
BOND CD1A CGA
BOND CE1A CZA
BOND CE2A CD2A
BOND CBB  CA
BOND CBB  HB1B
BOND CBB  HB2B
BOND CBB  HB3B
IMPR N    -C    CA    HN
IMPR C    CA    +N    O
END
```

3.1.2. Generating the PSF file

The PSF file is generated using `psfgen`. The first and third residues are standard alanine residues, so we should invoke both the topology file from CHARMM27 and the custom hybrid topology.

```
topology ../common/top_all22_prot.inp
topology tyr2ala.top

# Build the topology of both segments
segment Y2A {
  pdb tyr2ala.pdb
```

```
  first ACE
  last CT3
}
# The sequence of this segment is Ala-Y2A-Ala

# Read coordinates from pdb files
coordpdb tyr2ala.pdb Y2A

writepsf y2a.psf
writepdb y2a.pdb
```

Running the script with the command `psfgen setup.pgn` creates a PDB and a PSF file.

3.1.3. Preparing the alchFile

As in the case of the ethane-to-ethane transformation, the appearing and vanishing groups are defined by a `alchFile`. The `alchFile` for the tyrosine to alanine transformation, (`y2a.fep`), is easily prepared by editing a copy of `y2a.pdb`. The modified part should read:

```
...
ATOM     17  N   Y2A     2       5.841  -1.926  -3.336  1.00  0.00      YTOA N
ATOM     18  HN  Y2A     2       5.362  -2.371  -4.106  1.00  0.00      YTOA H
ATOM     19  CA  Y2A     2       7.291  -1.926  -3.336  1.00  0.00      YTOA C
ATOM     20  HA  Y2A     2       7.655  -0.898  -3.336  1.00  0.00      YTOA H
ATOM     21  CBA Y2A     2       7.842  -2.640  -2.100  1.00 -1.00      YTOA C
ATOM     22 HB1A Y2A     2       7.014  -2.994  -1.485  1.00 -1.00      YTOA H
ATOM     23 HB2A Y2A     2       8.452  -3.487  -2.411  1.00 -1.00      YTOA H
ATOM     24  CGA Y2A     2       8.687  -1.679  -1.298  1.00 -1.00      YTOA C
ATOM     25 CD1A Y2A     2       8.856  -0.360  -1.739  1.00 -1.00      YTOA C
ATOM     26 HD1A Y2A     2       8.377  -0.028  -2.660  1.00 -1.00      YTOA H
ATOM     27 CE1A Y2A     2       9.640   0.531  -0.996  1.00 -1.00      YTOA C
ATOM     28 HE1A Y2A     2       9.771   1.557  -1.339  1.00 -1.00      YTOA H
ATOM     29  CZA Y2A     2      10.254   0.104   0.187  1.00 -1.00      YTOA C
ATOM     30  OHA Y2A     2      11.016   0.969   0.909  1.00 -1.00      YTOA O
ATOM     31  HHA Y2A     2      11.063   1.844   0.516  1.00 -1.00      YTOA H
ATOM     32 CD2A Y2A     2       9.302  -2.106  -0.115  1.00 -1.00      YTOA C
ATOM     33 HD2A Y2A     2       9.170  -3.132   0.227  1.00 -1.00      YTOA H
ATOM     34 CE2A Y2A     2      10.086  -1.215   0.627  1.00 -1.00      YTOA C
ATOM     35 HE2A Y2A     2      10.564  -1.547   1.548  1.00 -1.00      YTOA H
ATOM     36  CBB Y2A     2       7.842  -2.640  -2.100  1.00  1.00      YTOA C
ATOM     37 HB1B Y2A     2       7.014  -2.994  -1.485  1.00  1.00      YTOA H
ATOM     38 HB2B Y2A     2       8.452  -3.487  -2.411  1.00  1.00      YTOA H
ATOM     39 HB3B Y2A     2       8.687  -1.679  -1.298  1.00  1.00      YTOA C
ATOM     40  C   Y2A     2       7.842  -2.640  -4.572  1.00  0.00      YTOA C
ATOM     41  O   Y2A     2       7.078  -3.122  -5.407  1.00  0.00      YTOA O
...
```

Visual inspection in VMD

Now, let us visualize the system containing the hybrid amino-acid. Run VMD with the following command: `vmd y2a.psf -pdb y2a.fep`. In the Graphics/Representations menu, set the coloring

method to `Beta`. The appearing alanine side chain should be colored blue and the tyrosine side chain should be red, while the backbone and the two unperturbed alanine residues should be green. Compare the result with Figure 7.

3.1.4. *Cleaning up the in vacuo structure file (Optional if using NAMD 2.7b2 or later)*

Again, removal of the spurious bonded force-field terms and addition of a non–bonded exclusion list can be performed using `alchemify`. It should be invoked using the following command line:
`alchemify y2a.psf y2a-alch.psf y2a.fep`

The resulting file `y2a-alch.psf` will be used when performing the *in vacuo* FEP calculations.

3.1.5. *Preparing the hydrated system*

Load the isolated tripeptide in VMD: `vmd y2a.psf y2a.pdb`. Open the Solvate interface (Extensions/Modeling/Add Solvation Box). Let's define a cubic, $26\times26\times26$ Å^3 water box: Uncheck the "Use Molecule Dimensions" box and the "Waterbox Only", set the minimum value of x, y and z to -13 and their maximum to $+13$. Running `solvate` creates the files `solvate.psf` and `solvate.pdb`.

A new `alchFile` containing the solvated structure has to be prepared, by copying `solvate.pdb` to `solvate.fep` and editing the latter manually to add FEP flags in the B column.

When running `solvate`, information about non–bonded exclusions is lost, so if using a version of NAMD earlier than 2.7b2, `solvate.psf` should be treated with `alchemify` again:
`alchemify solvate.psf tyr2ala_hydrated.psf solvate.fep`

3.2. Running the free energy calculations

3.2.1. *In vacuo simulation*

Prepare a NAMD configuration file for an MD run at a constant temperature of 300 K (Langevin dynamics with a damping coefficient of 10 ps^{-1}), with cutoff electrostatics and no particular boundary

conditions. Set the `rigidBonds` option to `all` and choose a time step of 0.5 fs. Save the trajectory to a DCD file for later inspection, with a frequency of *e.g.* 2000 time steps (1 ps). An example configuration file is provided.

3.2.2. *Solvated system*

Run MD at a constant temperature of 300 K (again with a damping coefficient of 1 ps^{-1}) and pressure of 1 bar, using PME electrostatics. Set the `rigidBonds` option to `all` and choose a time step of 1 fs. To impose isotropic fluctuations of the periodic box dimensions, set the `flexibleCell` variable to `no`. Set periodic boundary conditions with an initial $30\times30\times30$ $Å^3$ cubic periodic box. Save the trajectory to a DCD file for later inspection, with a frequency of *e.g.* 5000 time steps (5 ps). Before moving on to production simulations, the solvated system has to be equilibrated at the target temperature and pressure. First, perform an energy minimization step to remove possible steric clashes, then run a short MD simulation in the NPT ensemble.

3.2.3. *Sampling strategy*

The *in vacuo* transformation requires relatively long sampling times, because there are no solvent fluctuations that could couple to conformational fluctuations of the peptide (see Results). Fortunately, for such an extremely small system, a 0.5-nanosecond trajectory can be generated very quickly on one single processor. Here, use will be made of 20 contiguous windows, involving 25 ps of MD sampling — among which 2 ps of equilibration:

```
# FEP PARAMETERS

source                  ../../tools/fep.tcl

alch                    on
alchType                FEP
alchFile                y2a.fep
alchCol                 B
alchOutFile             forward.fepout
alchOutFreq             10

alchVdwLambdaEnd        1.0
alchElecLambdaStart     0.5
alchVdWShiftCoeff       4.0
alchDecouple            off
```

```
alchEquilSteps          4000
set numSteps            50000

set numMinSteps         1000

runFEPmin 0.0 1.0 0.05 $numSteps $numMinSteps $temp
```

A similar strategy is used for the solvated system, albeit with a somewhat larger time step. In each window, the system is equilibrated over `alchEquilSteps` MD steps, *viz.* here 100 steps, prior to 400 steps of data collection, making a total of 0.5 ps of MD sampling. Altogether, the alchemical transformation is carried out over 10 ps, and the backward transformation over the same time. Use the following FEP section:

```
# FEP PARAMETERS

source                  ../../tools/fep.tcl

alch                    on
alchType                FEP
alchFile                solvate.fep
alchCol                 B
alchOutFile             forward-off.fepout
alchOutFreq             10

alchVdwLambdaEnd        1.0
alchElecLambdaStart     0.5
alchVdWShiftCoeff       4.0
alchDecouple            off

alchEquilSteps          100
set numSteps            500

set numMinSteps         100

runFEPmin 0.0 1.0 0.05 $numSteps $numMinSteps $temp
```

3.3. Results

The $\Delta G(\lambda)$ curves for both simulations, as well as an alternate method, are shown in Figure 8. Using an adapted protocol for each of the two mutations, the free energy difference for the hydrated state is $+11.7$ kcal/mol, and $+4.0$ kcal/mol for the isolated state; your values may be slightly different due to

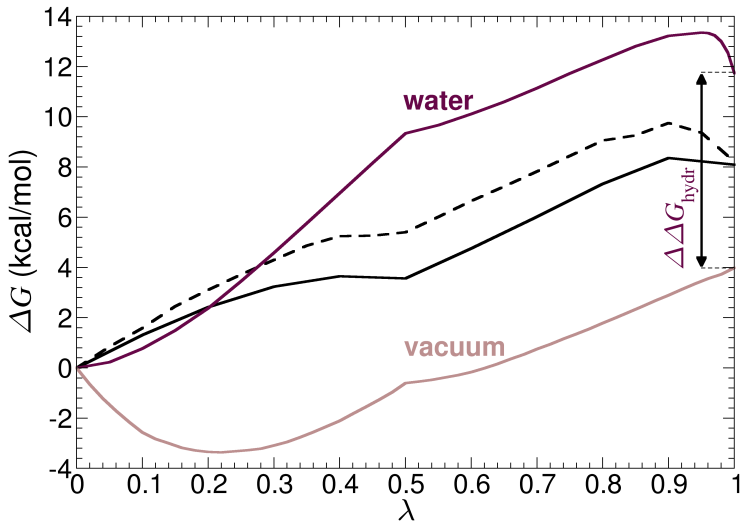


Figure 8: Results for the Tyr → Ala mutations in water and in vacuum, in the Ala–Tyr–Ala blocked tripeptide. Difference between the corresponding free-energy changes yields the relative hydration free energy of Ala–Tyr–Ala with respect to $(Ala)_3$. The transformation in the gas phase accounts for intramolecular interactions of the perturbed moieties, *i.e.* `decouple off`. As a basis of comparison and a consistency check, the results of decoupling-recoupling simulations, *i.e.* forward (dark solid line) and backward (dark dashed line) transformations with `decouple on`, in water are supplied. Because this option ignores intra-perturbed interactions, it also obviates the need for separate gas-phase simulations.

the very short simulation times used here. Using the thermodynamic cycle of Figure 6, one may write:

$$\Delta\Delta G = \Delta G^2_{\rm alch.} - \Delta G^1_{\rm alch.} = \Delta G^2_{\rm hydr.} - \Delta G^1_{\rm hydr.}$$

The net solvation free energy change $\Delta\Delta G$ for the Ala–Tyr–Ala → $(Ala)_3$ transformation found to be $+7.7$ kcal/mol. Alternately, a single decoupling/recoupling simulation indicates a hydration free energy difference of $+8.1$ kcal/mol, in agreement with the double annihilation scheme. This result may be related to the differential hydration free energy of side–chain analogues, *i.e.* the difference in the hydration free energy of methane and *p*–cresol, that is, respectively, 1.9 + 6.1 = +8.0 kcal/mol [20, 21]. Interestingly enough, Scheraga and coworkers have estimated the side–chain contribution for this mutation to be equal to +8.5 kcal/mol [22].

This very close agreement with experimental determinations based on side–chain analogues, as well as other computational estimates, may be in part coincidental or due to fortuitous cancellation of errors. Indeed, some deviation could be expected due to environment effects — *viz.* the mutation of a residue embedded in a small peptide chain *versus* that of an isolated, prototypical organic molecule[23] — and, to a lesser extent, the limited accuracy of empirical force fields. The first explanation may be related

to the concept of "self solvation" of the side chain. Here, the tyrosyl fragment is not only solvated predominantly by the aqueous environment, but also, to a certain degree, by the peptide chain, which, under certain circumstances, can form hydrogen bonds with the hydroxyl group.

Moreover, it should be noted that even for a small and quickly relaxing system such as the hybrid tripeptide, convergence of the FEP equation requires a significant time. In some cases, very short runs may give better results than moderately longer ones, because the former provide a local sampling around the starting configuration, while the latter start exploring nearby conformations, yet are not long enough to fully sample them.

In general, whether or not intramolecular interactions ought to be perturbed — *i.e.* `alchDecouple` set to `off` or `on`, respectively — requires careful attention. As has been seen here, ignoring perturbed intramolecular interactions is computationally advantageous in the sense that it obviates the need for the gas-phase simulation depicted in Figure 7. This choice is fully justified in the case of rigid, or sufficiently small molecules. If, however, the system of interest can assume multiple conformations, setting `alchDecouple` to `on` may no longer be appropriate. This is due to the fact that on account of the environment, specifically its permittivity, the conformational space explored in the low-pressure gaseous phase is likely to be different from that in an aqueous medium.

In all cases, visualizing MD trajectories is strongly advisable if one wishes to understand the behavior of the system and to solve possible sampling issues. Looking at the present tyrosine-to-alanine trajectories, it appears that the main conformational degree of freedom that has to be sampled is the rotation of the tyrosine hydroxyl group. Convergence is actually faster for the solvated system than for the tripeptide in vacuum, because fluctuations of the solvent help the tyrosine side chain pass the rotational barriers, which does not happen frequently in vacuum.

4. Binding of a potassium ion to 18–crown–6

The main thrust of this section is to provide the theoretical framework for measuring standard binding free energies, resorting to the simple example of a potassium ion associated to a crown ether, namely 18–crown–6. 18–crown–6:K^+ has been investigated previously by means of molecular simulations. In their pioneering work, Dang and Kollman evaluated the potential of mean force delineating the reversible sep-

aration of the ion from the ionophore [24, 25]. Alchemical FEP, however, has proven over the years [7] to constitute a method of choice for measuring *in silico* host:guest binding free energies.

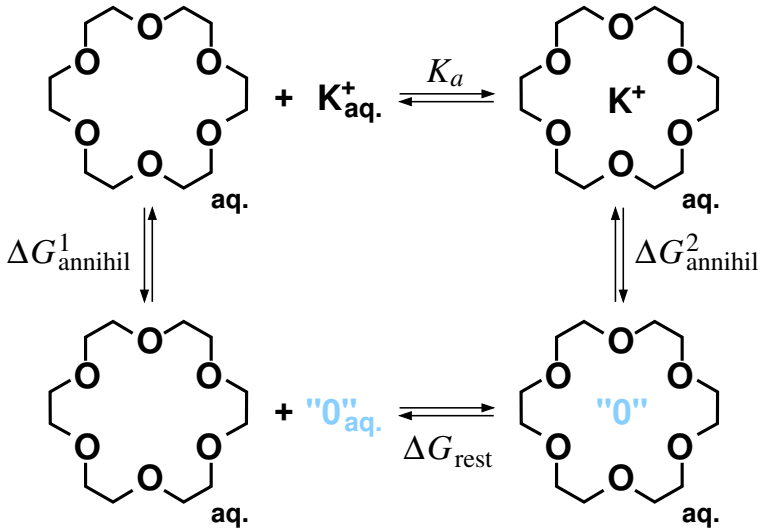


Figure 9: Thermodynamic cycle delineating the reversible association of a potassium ion to the 18–crown–6 crown ether in an aqueous environment. Binding of the ion to the ionophore is characterized by the association constant K_a and can be measured directly by means of a potential of mean force–like calculation. Alternatively, FEP can be employed to annihilate, or create the potassium ion both in the free and in the bound states. It follows from the latter that $\Delta G_{\text{binding}} = -1/\beta \ln K_a = \Delta G^1_{\text{annihil}} - \Delta G^2_{\text{annihil}}$. To prevent the cation from moving astray when it is only weakly coupled to the crown ether, *i.e.* at the very end of the annihilation transformation, or at the beginning of the creation transformation, positional restraints ought to be enforced, contributing ΔG_{rest} to the overall binding free energy.

Here, the standard binding affinity of a potassium ion towards 18–crown–6 will be determined using this approach and following the thermodynamic cycle of Figure 9, wherein the alkaline ion undergoes a double annihilation [26], in its free and bound states.

4.1. System setup

Double annihilation implies that the ion will be annihilated both in the free and in the bound state. In other words, two different molecular systems have to be built, *viz.* . a potassium ion in a water bath and the complex 18–crown–6:K^+, also in a water bath.

4.1.1. Building the host:guest complex

The topology of 18–crown–6 is supplied in `18crown6.top`. In addition, the initial coordinates of the ionophore are provided in the form of a PDB file, `18crown6.pdb`. The first step of the setup consists in building the PSF file for the host, employing VMD and `psfgen`. Proceed similarly for the guest, *i.e.* the potassium ion.

Once the PSF files for the host and the guest are available, merge the two species into a single PSF and PDB file by issuing the following commands in the VMD TKConsole:

```
readpsf 18crown6.psf
readpsf potassium.psf
coordpdb 18crown6.pdb
coordpdb potassium.pdb
writepsf complex.psf
writepdb complex.pdb
```

Next, open in VMD the files `complex.psf` and `complex.pdb`, and center the alkaline ion with respect to the center of mass of 18–crown–6, utilizing the command `$sel moveby` $\{x\ y\ z\}$, where `$sel` is the selection corresponding to the cation, and x, y and z, a vector for translating it to the centroid of the crown ether. Write out a new pdb file for the entire system.

4.1.2. Preparing the hydrated system

The size-dependence correction imposed by the long-range nature of charge–dipole interactions is expected to cancel out if the dimensions of the simulation cell are identical for the two vertical legs of the thermodynamic cycle of Figure 9. In both cases, a box of dimension $28 \times 28 \times 28$ Å^3 appears to constitute a reasonable compromise in terms of cost-effectiveness.

Employing `solvate` as done previously creates the files `solvate.psf` and `solvate.pdb` for the two different setups.

In each case, a new `alchFile` containing the solvated structure has to be prepared, by cloning `solvate.pdb` to `solvate.fep` and editing the latter manually to add FEP flags in the B column for the vanishing alkaline ion.

4.2. Running the free energy calculations

4.2.1. Definition of the restraining potential

As usual, equilibration of the molecular systems will be carried out as a preamble to the free-energy calculations. In the case of the hydrated 18–crown–6:K^+ complex, this thermalization step will be utilized to appreciate how strongly bound is the guest in its dedicated binding site.

To this end, after energy minimization to remove possible steric clashes, equilibrium MD will be run for both systems at a constant temperature of 300 K with a damping coefficient of 1 ps^{-1}, and pressure of 1 bar, using PME electrostatics. The `rigidBonds` option will be set to `all` to increase the integration time step of 2 fs. To impose isotropic fluctuations of the periodic box dimensions, the `flexibleCell` variable will be set to `no`. The trajectory will be saved in a DCD file with a frequency of *e.g.* 100 time steps. Check that the solvated system is equilibrated by monitoring the target temperature and pressure.

To measure how the position of the alkaline ion fluctuates in the ionophore, the trajectory of the 18–crown–6:K^+ complex can be loaded and the coordinates of the system aligned with respect to the heavy atoms of the crown ether. The alignment and calculation of the distance separating the center of mass of the latter from the ion every frame can be performed using the following simple TCL script:

```
set outfile [open COM-ion.dat w]

set nf [molinfo top get numframes]

set all [atomselect top "all"]

set crown [atomselect top "resname 18C6 and noh"]
set crown0 [atomselect top "resname 18C6 and noh" frame 0]

set ion [atomselect top "segname POT"]

for { set i 1 } { $i <= $nf } { incr i } {
   $all frame $i
   $crown frame $i
   $ion frame $i
   $all move [measure fit $crown $crown0]
   $crown update
   $ion update
   set COM [measure center $ion weight mass]
   set dist [expr sqrt(pow([lindex $COM 0],2)+pow([lindex $COM 1],2)+pow([lindex $COM 2],2))]
   puts $outfile [format "%8d %8f" $i $dist]
}
```

```
close $outfile
```

The maximum fluctuation in the distance between the center of mass of the ionophore and the potassium ion will serve as the basis for a positional restraint introduced in the subsequent free-energy calculation, in which the guest is annihilated in its bound state.

4.2.2. Sampling strategy

An identical sampling strategy will be used for the annihilation of the potassium ion in the free and in the bound state. Here, use will be made of 32 equally spaced intermediate states:

```
# FEP PARAMETERS

source                  ../../tools/fep.tcl

alch                    on
alchType                FEP
alchFile                solvate.fep
alchCol                 B
alchOutFreq             10
alchOutFile             forward.fepout

alchElecLambdaStart     0.1
alchVdwLambdaEnd        1.0
alchVdwShiftCoeff       5.0
alchdecouple            on

alchEquilSteps          50
set numSteps            200

set dLambda             0.03125

runFEP 0.0 1.0 $dLambda $numSteps
```

Enforcing a positional restraint by means of an isotropic harmonic potential — *i.e.* an external potential that confines the ion within a sphere of given radius, centered about the center of mass of the ionophore — the COLVARS module of NAMD will be employed. To do so, the following two lines ought to be added in the NAMD configuration or input file:

```
colvars                 on
colvarsConfig           COMCOM.in
```

In addition, a separate, dedicated `colvarsConfig` file, will be written to instruct COLVARS of the positional restraint to be enforced:

```
colvarsTrajFrequency     100
colvarsRestartFrequency  100

colvar {
   name COMDistance

   width 0.1

   lowerboundary 0.0
   upperboundary 0.5

   lowerWallConstant  100.0
   upperWallConstant  100.0

   distance {
      group1 {
         atomnumbers {  43 }
      }
      group2 {
         atomnumbers {  1    4    5    8   11   12
                       15   18   19   22   25   26
                       29   32   33   36   39   40 }
      }
   }
}
```

Here, a confinement potential will be imposed to the ion to remain within a sphere of 1 Å diameter. This positional restraint corresponds to a loss of translational entropy equal to $-1/\beta \ln(c_0 \Delta v)$, where Δv is the effective volume sampled by the guest and c_0 is the usual standard concentration.

4.3. Results

The results of the two independent free-energy calculations are depicted in Figure 10. To probe micro-reversibility of the transformation, both forward, annihilation, and backward, creation simulations were run. The marginal hysteresis between either pair of free-energy profiles is suggestive of a low finite-length systematic error — albeit a closer examination of the associated probability distributions, $P_0[\Delta U(\mathbf{x})]$ and $P_1[\Delta U(\mathbf{x})]$, is necessary to ascertain that such is indeed the case [27, 7].

The difference between the net free-energy changes for the ion in its free and bound states yields the binding free energy, to which the contribution due to the positional restraint ought to be added. Confining

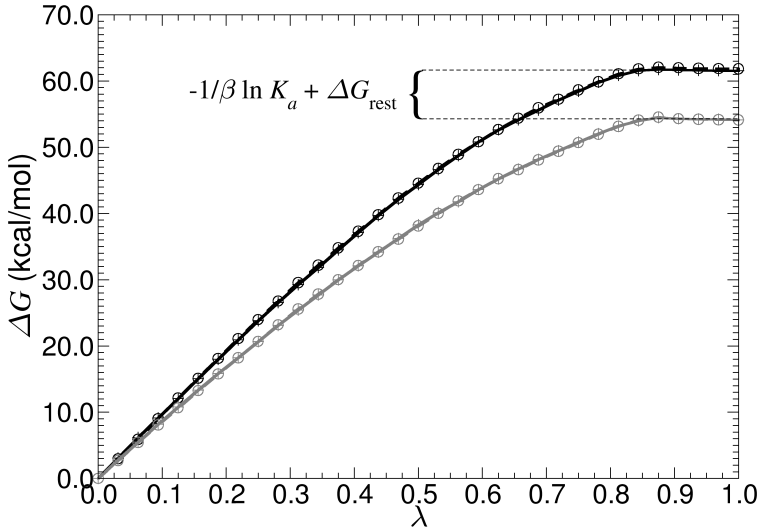


Figure 10: Free-energy change for the double annihilation of a potassium ion in its bound state, associated to 18–crown–6 (black solid line and pluses for the forward transformation, black dashed line and circles for the backward transformation), and in its free state, in a bulk aqueous environment (light solid line and pluses for the forward transformation, red dashed line and circles for the backward transformation). The difference at λ = 1, between the net free-energy changes, $\Delta G^1_{\mathrm{annihil}} - \Delta G^2_{\mathrm{annihil}}$, *viz.* 54.1 − 61.5 = -7.4 kcal/mol, corresponds to the binding free energy, *i.e.* $-1/\beta \ln K_a$, to which the contribution due to the confinement of the ion in the crown ether, *viz.* 4.8 kcal/mol, ought to be added. The resulting standard binding free energy of -2.6 kcal/mol is in good agreement with the experimental measurement of -2.91 kcal/mol of Michaux and Reisse [28].

the ion in a spherical volume enclosed in the ionophore to prevent it from escaping as host:guest coupling fades out is tantamount to a loss of translational entropy, which can be evaluated analytically. This entropic term corresponds to a free-energy contribution equal to $-1/\beta \ln(c_0 \Delta v)$, which in the case of a spherical volume element of 0.52 Å^3, amounts to about 4.8 kcal/mol.

Put together, the theoretical estimate for the binding free energy of a potassium ion associated to 18–crown–6 is equal to -2.6 kcal/mol, which appears to be in good agreement with the available experimental measurement of -2.91 kcal/mol [28]. The accuracy is clearly improved compared to estimates from the early 1990s by Dang and Kollman, based on potential of mean force calculations with the limited sampling times then feasible: their first calculations [24] yielded -3.5 kcal/mol, whereas the second publication [25] reports -2.0 ± 0.3 kcal/mol.

Role of confinement potentials

Introduction of a positional restraint by means of the COLVARS module can be substituted by the addition of pseudo bonds between the ionophore and the alkaline cation. These bonds, aimed at tethering the ion as its coupling to the crown ether vanishes, are declared in the NAMD configuration file by:

```
extraBonds     yes
extraBondsFile restraints.txt
```

The reader is invited to refer to the NAMD user's guide for the syntax employed for defining pseudo bonds in the external `extraBondsFile` file, and verify that following this route and subsequently measuring the free-energy cost incurred for fading out these pseudo bonds — *i.e.* by zeroing out reversibly the associated force constants, the aforementioned contribution due to the loss of translational entropy can be recovered.

Finite-length bias

For each individual λ-intermediate state, monitor the probability distribution functions $P_0[\Delta U(\mathbf{x})]$ and $P_1[\Delta U(\mathbf{x})]$, and verify that the systematic error due to the finite length of the free-energy calculation [7] is appreciably small.

References

[1] Dixit, S. B.; Chipot, C., Can absolute free energies of association be estimated from molecular mechanical simulations ? The biotin–streptavidin system revisited, *J. Phys. Chem. A* **2001**, *105*, 9795–9799.

[2] Phillips, J. C.; Braun, R.; Wang, W.; Gumbart, J.; Tajkhorshid, E.; Villa, E.; Chipot, C.; Skeel, L.; Schulten, K., Scalable molecular dynamics with NAMD, *J. Comput. Chem.* **2005**, *26*, 1781–1802.

[3] Bhandarkar, M.; Brunner, R.; Buelens, F.; Chipot, C.; Dalke, A.; Dixit, S.; Fiorin, G.; Freddolino, P.; Grayson, P.; Gullingsrud, J.; Gursoy, A.; Hardy, D.; Harrison, C.; Hénin, J.; Humphrey, W.; Hurwitz, D.; Krawetz, N.; Kumar, S.; Nelson, M.; Phillips, J.; Shinozaki, A.; Zheng, G.; Zhu, F. *NAMD user's guide, version 2.7*. Theoretical biophysics group, University of Illinois and Beckman Institute, Urbana, IL, July 2009.

[4] Born, M., Volumen und Hydratationswarme der Ionen, *Z. Phys.* **1920**, *1*, 45–48.

[5] Chipot, C.; Pearlman, D. A., Free energy calculations. The long and winding gilded road, *Mol. Sim.* **2002**, *28*, 1–12.

[6] Chipot, C. Free energy calculations in biological systems. How useful are they in practice? in *New algorithms for macromolecular simulation*, Leimkuhler, B.; Chipot, C.; Elber, R.; Laaksonen, A.; Mark, A. E.; Schlick, T.; Schütte, C.; Skeel, R., Eds., vol. 49. Springer Verlag, Berlin, 2005, pp. 183–209.

[7] Chipot, C.; Pohorille, A., Eds., *Free energy calculations. Theory and applications in chemistry and biology*, Springer Verlag, 2007.

[8] Gao, J.; Kuczera, K.; Tidor, B.; Karplus, M., Hidden thermodynamics of mutant proteins: A molecular dynamics analysis, *Science* **1989**, *244*, 1069–1072.

[9] Pearlman, D. A., A comparison of alternative approaches to free energy calculations, *J. Phys. Chem.* **1994**, *98*, 1487–1493.

[10] Humphrey, W.; Dalke, A.; Schulten, K., VMD: visual molecular dynamics, *J. Mol. Graph.* **1996**, *14*, 33–8, 27–8.

[11] Pearlman, D. A.; Kollman, P. A., The overlooked bond–stretching contribution in free energy perturbation calculations, *J. Chem. Phys.* **1991**, *94*, 4532–4545.

[12] Zacharias, M.; Straatsma, T. P.; McCammon, J. A., Separation-shifted scaling, a new scaling method for Lennard-Jones interactions in thermodynamic integration, *J. Chem. Phys.* **1994**, *100*, 9025–9031.

[13] Beutler, T. C.; Mark, A. E.; van Schaik, R. C.; Gerber, P. R.; van Gunsteren, W. F., Avoiding singularities and neumerical instabilities in free energy calculations based on molecular simulations, *Chem. Phys. Lett.* **1994**, *222*, 529–539.

[14] Pitera, J. W.; van Gunsteren, W. F., A comparison of non–bonded scaling approaches for free energy calculations, *Mol. Sim.* **2002**, *28*, 45–65.

[15] Hummer, G.; Pratt, L.; Garcia, A. E., Free energy of ionic hydration, *J. Phys. Chem.* **1996**, *100*, 1206–1215.

[16] Marcus, Y., Thermodynamics of solvation of ions: Part 5. Gibbs free energy of hydration at 298.15 K, *J. Chem. Soc. Faraday Trans.* **1991**, *87*, 2995–2999.

[17] Straatsma, T. P.; Berendsen, H. J. C., Free energy of ionic hydration: Analysis of a thermodynamic integration technique to evaluate free energy differences by molecular dynamics simulations, *J. Chem. Phys.* **1988**, *89*, 5876–5886.

[18] Kollman, P. A., Free energy calculations: Applications to chemical and biochemical phenomena, *Chem. Rev.* **1993**, *93*, 2395–2417.

[19] MacKerell, Alexander D; Feig, Michael; Brooks, Charles L, Extending the treatment of backbone energetics in protein force fields: limitations of gas-phase quantum mechanics in reproducing protein conformational distributions in molecular dynamics simulations, *J. Comput. Chem.* **2004**, *25*, 1400–1415.

[20] Dixit, S. B.; Bhasin, R.; Rajasekaran, E.; Jayaram, B., Solvation thermodynamics of amino acids. Assessment of the electrostatic contribution and force-field dependence, *J. Chem. Soc., Faraday T rans.* **1997**, *93*, 1105–1113.

[21] Shirts, M. R.; Pande, V. S., Solvation free energies of amino acid side chain analogs for common molecular mechanics water models, *J. Chem. Phys.* **2005**, *122*, 134508.

[22] Ooi, T.; Oobatake, M.; Némethy, G.; Scheraga, H. A., Accessible surface areas as a measure of the thermodynamic parameteres of hydration of peptides, *Proc. Natl. Acad. Sci. USA* **1987**, *84*, 3086–3090.

[23] König, G.; Boresch, S., Hydration free energies of amino acids: why side chain analog data are not enough, *J. Phys. Chem. B* **2009**, *113*, 8967–8974.

[24] Dang, Liem X.; Kollman, Peter A., Free energy of association of the 18-crown-6:K+ complex in water: a molecular dynamics simulation, *J. Am. Chem. Soc.* **1990**, *112*, 5716–5720.

[25] Dang, Liem X.; Kollman, Peter A., Free Energy of Association of the K+:18-Crown-6 Complex in Water: A New Molecular Dynamics Study, *J. Phys. Chem.* **1995**, *99*, 55–58.

[26] Gilson, M. K.; Given, J. A.; Bush, B. L.; McCammon, J. A., The statistical-thermodynamic basis for computation of binding affinities: A critical review, *Biophys. J.* **1997**, *72*, 1047–1069.

[27] Zuckerman, D.M.; Woolf, T.B., Theory of a systematic computational error in free energy differences, *Phys. Rev. Lett.* **2002**, *89*, 180602.

[28] Michaux, Gabriel; Reisse, Jacques, Solution thermodynamic studies. Part 6. Enthalpy-entropy compensation for the complexation reactions of some crown ethers with alkaline cations: a quantitative interpretation of the complexing properties of 18-crown-6, *J. Am. Chem. Soc.* **1982**, *104*, 6895–6899.