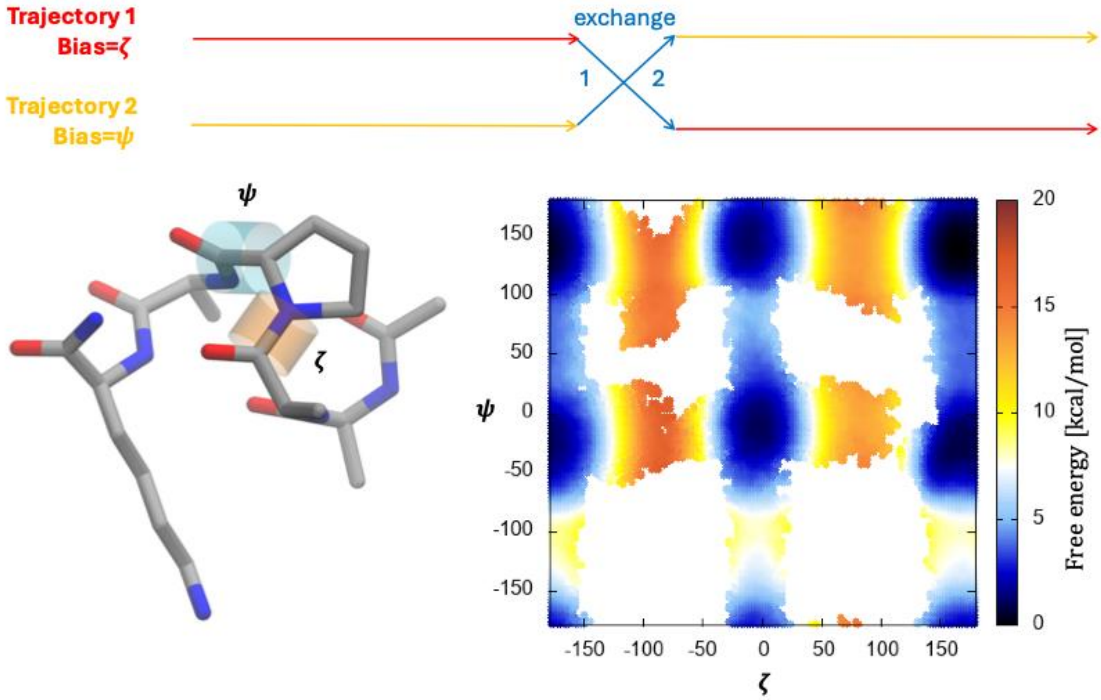
Setting Up and Analyzing Bias-Exchange Metadynamics Simulations: the case of cis-trans isomerization of Peptidyl-Prolyl Peptides

Fabrizio Marinelli and Vanessa Ariadna Leone Alvarez

Department of Biophysics and Data Science Institute, Medical College of Wisconsin,
Milwaukee, Wisconsin 53226-3548, United States

Correspondence should be addressed to: Fabrizio Marinelli (fmarinelli@mcw.edu) or Vanessa Ariadna Leone Alvarez (vleone@mcw.edu)



This tutorial offers a comprehensive protocol and practical examples for setting up and analyzing adaptively biased replica simulations of cis-trans isomerization in a proline-containing peptide in explicit solvent, with a particular emphasis on bias-exchange metadynamics. It begins with the basic setup and analysis of MD simulations for a solvated peptidyl-prolyl peptide, and progresses to the configuration of metadynamics and bias-exchange metadynamics simulations targeting dihedral angles that describe conformational changes of the peptide and the cis-trans transition at the peptide bond. The tutorial also outlines a detailed analysis protocol for calculating multidimensional free energies using weighted histogram analysis, as well as mean forces (https://github.com/FCAM-NIH/FCAM). Additionally, it covers how to visualize and extract structures, and perform kinetic analysis using the free energy map, with the METAGUI3 plugin (https://github.com/metagui/metagui3). To follow this tutorial, you will need a Linux computer with GROMACS (https://www.gromacs.org), PLUMED2 (https://www.plumed.org), VMD (https://www.ks.uiuc.edu/Research/vmd), Xmgrace, METAGUI3, and the tool implementing the force-correction analysis method (FCAM, https://github.com/FCAM-NIH/FCAM) installed, along with Python (preferably within anaconda) for the latter (check https://github.com/FCAM-NIH/FCAM for details). Visualization and analysis tools like Xmgrace can easily be replaced with suitable alternatives, such as pyplot along with Python-based analysis libraries.

Molecular dynamics of peptidyl prolyl peptides.

The peptidyl bond formed by all the 20 natural amino acids, except proline, exist almost exclusively in the *trans* conformation. Indeed, *cis* and *trans* form of proline peptides are nearly isoenergetic. Here we will expore the *cis/trans* population of a proline containing peptide. In particular, we will study the *cis-trans* isomerization of two 5mer, Ac-Ala-Ala-Pro-Ala-Lys-NH2, and compare against experimental values (Reimer et al., 1998).

1. Setup of Ac-Ala-Ala-Pro-Ala-Lys-NH2 peptide

The *cis* population of this small peptide was determined experimentally at pH 6.0 by Reimer *et al.* (Reimer *et al*., 1998); at this pH the lysine of the peptide in water solution is charged (Lys side chain pKa is 8.95). We will use GROMACS (www.gromacs.org, Abraham et al. 2015) software to setup and equilibrate both systems. This is a free package to perform molecular dynamics primary of biological molecules (protein, lipids and nucleic acids). It also includes a large variety of tools for set up (system topology, water box building, etc) and analyze simulations.

- Within practical1/ folder create a setup folder and the peptide conformation from the provided-files/ folder:

```
mkdir setup
cp provided-files/ace-AAPAK-ct2.pdb setup/.
```

- Use VMD software (https://www.ks.uiuc.edu/Research/vmd, Humphrey et al. 1996) to visualize the structural file copied to the setup/ folder

```
cd setup
vmd ace-AAPAK-ct2.pdb
```

As a default the peptide will be displayed as ‘Lines’ connecting the atoms. You can change the molecule visualization scheme, color, etc in the ‘Graphical Representations’ windows prompted after clicking ‘Representation’ in the ‘Graphics’ menu.

- To load a recent version of GROMACS for instance by typing:

```
source GROMACS-bin-dir/GMXRC
```

- We will use the pdb2gmx utility to setup the system (type *gmx pdb2gmx -h* to get information our all the options). Run this program typing:

```
gmx pdb2gmx -f ace-AAPAK-ct2.pdb -p AAPAK.top -ter -lys -water tip3p -o AAPAK.gro
```

Options:
-f indicate the input pdb
-p indicate the name of the topology file that will be generated by pdb2gmx
-ter if added, the termini (here acetylation and amidation) can be included interactively
-lys the protonation of lysine (only ionizable residue in the two peptides simulated in this tutorial) can be specified interactively
-water, model of water that will be used in our simulation (we will use CHARMM force field, so we choose tip3p water model)
-o indicate the final structure in gromacs format (.gro); change accordingly

- Select ‘CHARMM27’ forcefield (option 8)
Then select ‘protonated (charged +1)’ lysine (option 1)
When asked for the start terminus, select ‘None’ (option 2) because we have already added the acetylation on the structure with the residue ACE
Select ‘CT2’ (option 2) for the end terminus, that represent an amidated C-terminus

Note that the utility pdb2gmx indicates that you have net total charge of 1 (*Total charge 1.000 e*), we will need to add at least a negatively charged ion afterwards to neutralize the charge of the system.

The tool pdb2gmx should generated three files: the structural final .gro; the topology .top and an .itp file that contains positional restrains definition (on heavy atoms)

- You can use VMD to visualize the final structural file

```
vmd AAPAK.gro
```

As you can see while creating the topology of the peptide pdb2gmx has added the protons bound to the heavy atoms present in the initial structure.

- The topology file can be open with any text editor or reader, for example you can type

```
less AAPAK.top
```

After few commented lines indicated with ‘;’, the first line that you can find in the file is

```
#include "charmm27.ff/forcefield.itp"
```

This indicates that the force field used is CHARMM27, and the parameters of the present topology have been derived from this force field.

The subsequent command indicates the name of the protein (simply ‘Protein’ because we do not have any chain specified for the peptide) and that non-bonded interactions for neighboring atoms are excluded 3 bonds away.

```
[ moleculetype ]
; Name            nrexcl
Protein             3
```

Next section section defines the atoms of the system:

```
[ atoms ]
;   nr       type  resnr residue  atom   cgnr     charge       mass  typeB    chargeB      massB
; residue   1 ACE rtp ACE  q  0.0
     1        CT3      1    ACE    CH3      1      -0.27     12.011   ; qtot -0.27
     2         HA      1    ACE   HH31      2       0.09      1.008   ; qtot -0.18
     3         HA      1    ACE   HH32      3       0.09      1.008   ; qtot -0.09
```

*nr: Atom number; type: Atom type; resnr: **Amino acid** number that the atom belongs; residue: **amino acid** residue name; atom: atom name; cgnr: charge group number (sometimes several atoms are grouped as the same charged particle to speed up the calculation; this is not the case here); qtot: running sum of the molecule charge. Neglect TypeB, ChargeB and massB since are needed for free energy calculations not used in this course.*

The bond section specifies which atoms are covalently bound, and that this bond is represented by a function tabulated as 1 (harmonic potential of equation 4.31 of Gromacs-5.0.7 manual)

```
[ bonds ]
```

```
;  ai    aj funct           c0            c1            c2            c3
    1     2     1
    1     3     1
```

In [pairs] section 1-4 interactions between atoms are specified.

```
[ pairs ]
;  ai    aj funct           c0            c1            c2            c3
    1     8     1
    1     9     1
```

The angles between three atoms are represented with an Urey-Bradley potential (function number 5) as described in Section 4.2.8 of Gromacs Manual.

```
[ angles ]
;  ai    aj    ak funct           c0            c1            c2            c3
    2     1     3     5
    2     1     4     5
```

The proper periodic dihedral between 4 atoms is described with the equation 4.60 of Gromacs Manual.

```
[ dihedrals ]
;  ai    aj    ak    al funct           c0            c1            c2            c3            c4            c5
    2     1     5     6     9
    2     1     5     7     9
```

Instead improper dihedrals in the second [dihedral] section are represented with the harmonic potential shown in equation 4.59 of Gromacs Manual.

```
[ dihedrals ]
;  ai    aj    ak    al funct           c0            c1            c2            c3
    5     1     7     6     2
    7     5     9     8     2
```

Under [cmap] section is indicated the CMAP treatment of backbone dihedral (cross-term for the phi and psi backbone dihedral), a particular term of CHARMM forcefield.

```
[ cmap ]
;  ai    aj    ak    al    am funct
    5     7     9    15    17     1
   15    17    19    25    27     1
```

In this section, if we specify the usage of positional restraints in the run, the atoms and force constants to be used are read from the file posre.itp

```

; Include Position restraint file
#ifdef POSRES
#include "posre.itp"
#endif
```

Here it is indicated that we use TIP3P model for the waters

```
#include "charmm27.ff/tip3p.itp"
```

The following section defines the positional restraint to be applied to the oxygen of each water molecule if we indicate that we will restrain waters during the run.

```
#ifdef POSRES_WATER
; Position restraint for each water oxygen
[ position_restraints ]
;  i funct       fcx        fcy        fcz
   1    1       1000       1000       1000
#endif
```

Indicate where to search for the ions parameters.

```
#include "charmm27.ff/ions.itp"
```

Under the system session we can specify the name of the system to be simulated

```
 [ system ]
; Name
Protein
```

In the molecules section it is specified the molecules to be simulated, here we only have the oligopeptide.

```
[ molecules ]
; Compound        #mols
Protein             1
```

We will define the size and shape of the simulation box using the editconf tool (check different options typing *gmx editconf -h*). We can use four box types: triclinic, cubic, dodecahedron and octahedron. We also have to specify the minimum distance between the oligopeptide molecule and the box boundaries in nanometers. We will simulate under periodic boundary conditions, thus we have to choose the distance of the edge of the box in a manner that the peptide never interacts with its periodic image. A box distance of 1.2 nm indicates that there are at least 2.4 nm between two peptide periodic images; which should be enough of the cutoff of 1.2 nm (the one recommended for CHARMM force field) that we are using in the simulations of this course.
- In the command line type:

```
gmx editconf -f AAPAK.gro -o AAPAK_box.gro -bt cubic -d 1.2
```

If you open the output .gro file (less AAPAK_box.gro) you can see that at the last line is specified the new box size Now that you have the box defined you can add waters, you will use the solvate tool for this purpose (type *gmx solvate -h* for available options). Enter the following command:

```
gmx solvate -cp AAPAK_box.gro -cs -o AAPAK_solv.gro -p AAPAK.top
```

-cp specifies the input structure file; we used the default value for -cs, thus the water structure is obtained from spc216.gro inside Gromacs library. Since both SPC and TIP3P water models contains 3 atoms, a short equilibration with the parameter of the latter will yield to a TIP3P compliant conformation. Option -o indicates the output structural file with the solvated oligopeptide and -p the topology to be read and then written after the addition of waters.

In the topology file (*less AAPAK.top*) a line has been added for all the solvent molecules under the [molecules] section.

```
[ molecules ]
; Compound        #mols
Protein            1
SOL             1419
```

Explore the newly generated .gro file (vmd *AAPAK_solv.gro*). Note that instead of displaying a single peptide, you now have a box of water solvating it.

As we mentioned before our system has a net charge of +1, and therefore we have to add a counter ion to neutralize this charge. Also we would like to perform our simulations at a similar ionic strength as the one used in the reference experiments. Reimer *et al*, have performed they measurements in sodium phosphate buffer set at pH 6.0. Using Hendelson-Hasselbach equation for the second equilibrium with pKa2=7.21 (and neglecting the equilibria of pKa1 = 2.16 and pKa3 = 12.32) we can estimate that [H2PO4-] is 18.8mM and [HPO42-] is 1.2mM, plus 20mM of Na+, the total ionic strength is 21.8mM. We will add NaCl (0.0218 M NaCl) to reproduce this ionic strength in the experiments.

Ions can be added easily in Gromacs using the genion tool, that substitutes water molecules in the box by the chosen ion molecules. The input of the genion is a file with the .tpr extension, that combines the information of the system structure (.gro file) with the topology (.top file) and simulation parameters (.mdp file that contains number of steps, time step, thermostat, etc). We will, then, first generate a .tpr file using the grompp utility and a .mdp file.

- Copy the provided .mdp file to the setup folder.

```
cp ../provided-files/dummy.mdp .
```

- Run grompp (check grompp options by typing grompp -h) utility to obtain a .tpr file

```
gmx grompp -f dummy.mdp -c AAPAK_solv.gro -p AAPAK.top -o genion.tpr -maxwarn 1
```

After running this command, you should have a file named genion.tpr that you can use as an input for genion tool.

- Type the following command to add ions to your system:

```
gmx genion -s genion.tpr -o AAPAK_solv_ions.gro -p AAPAK.top -pname NA -nname CL -conc 0.0218
-neutral
```

-s option specifies the input .tpr file; -o the output structure file (.gro); -p the topology file (.top); -pname and -nname the positively and negatively ions to be added (by default they are NA – sodium – and CL – chloride); -conc the salt concentration (in M units); -neutral indicates that on top of the salt included to obtain a desired ionic strength counter-ions will be added to neutralize the net charge of the system.

- When prompted, select SOL molecules (option 13) to be substituted by added ions.

The [molecule] section of the newly generated topology file (less AAPAK.top) indicates that three water molecules were substituted by 1 sodium and 2 chloride ions.

```
[ molecules ]
```

```
; Compound        #mols
Protein            1
SOL         1416
NA                1
CL                2
```

- Inspect where the ions were placed in the final structure file (vmd AAPAK_solv_ions.gro)

2. Energy minimization of solvated peptide

At this point you have a water solvated, neutralized system that you can simulate. First you will perform some steps of energy minimization to reduce possible geometrical problems of the initial structure. To run any minimization/simulation in Gromacs we need a .tpr file, similar to the one we used to run the genion tool. Again we can use the grompp utility to obtain this file, but we will feed a different .mdp file, where we specify the options of the run to be performed.

- Copy the provided min.mdp file to the setup folder

```
cp ../provided-files/min.mdp .
```

The options chosen in the min.mdp file are explained as comments on the file (less min.mdp). You can check all possible mdp options in the http://manual.gromacs.org/online/mdp_opt.html website.
- Generate a .tpr file to run the minimization using the grompp tool:

```
gmx grompp -f min.mdp -c AAPAK_solv_ions.gro -p AAPAK.top -o min.tpr
```

- The minimization will be run in a different folder. For this purpose create a min/ folder in practical1/ folder, copy the min.tpr generated before and then move to the new folder:

```
mkdir ../min
cp min.tpr ../min/.
cd ../min
```

- Run the minimization typing the following command:

```
gmx mdrun -deffnm min
```

-deffnm indicates that all the input and output files are named min. After the minimization is finished you should get the following files:

```
min.log – text log file
min.edr – energy file
min.trr – trajectory file
min.gro – final minimized structure
```

After finished mdrun should indicate if the minimization algorithm converged, in how many steps and the final potential energy:

```
Steepest Descents converged to Fmax < 1000 in 143 steps
Potential Energy  = -6.3610281e+04
```

We can monitor *a posteriori* how the potential energy has change along the minimization run.

- For that we will extract the potential energy as a function of time from the binary .edr file using the energy utility:

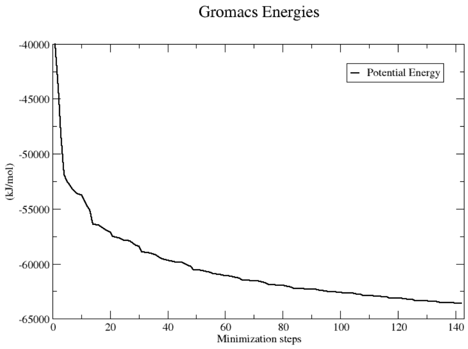
```
gmx energy -f min.edr -o upot.xvg
```

- When prompted select ‘Potential’ (Option 11) and then enter a blank line to end the interactive input.

Now we can plot the obtained upot.xvg file for example with xmgrace by typing:

```
xmgrace upot.xvg
```

The plot obtained (similar to the one displayed below) may show that the energy minimization algorithm has indeed lowered the potential energy until reaching a plateau. Now our system may not present geometric problems, thus we can use the final minimized structure to initiate the equilibration of the oligopeptide in water.



3. Equilibration of solvated peptide

Before performing the production runs we need to equilibrate the system, mainly we need that the solvent rearrange around the water. A common strategy is to perform molecular dynamics applying positional restraints to the solute (here the peptide), and release these restraints gradually.
We will run our simulations with NPT ensemble; the temperature is coupled by rescaling the velocity with a stochastic term (v-rescale keyword) and the pressure will be coupled with the Berendsen barostat (Berendsen keyword).

- Create a folder to run the equilibration, copy the final equilibrated structure from the min/ folder and the topology and restraints file from the setup/ folder:

```
mkdir ../eq
cp min.gro ../eq/.
cd ../eq
cp ../setup/AAPAK.top .
cp ../setup/posre.itp .
```

- Create the .tpr file needed to run the equilibration. For that purpose, first copy the eq1.mdp file from the provided-files/ folder and then run the grompp tool. An explanation for every option on the eq1.mdp has been added as a comment, if you need more information go to the http://manual.gromacs.org/online/mdp_opt.html website.

```
cp ../provided-files/eq1.mdp .
gmx grompp -f eq1.mdp -c min.gro -p AAPAK.top -o eq1.tpr
```

- Using the mdrun program run the equilibration run:

```
nohup gmx mdrun -deffnm eq1 >& out &
```

After the simulation finished you should have these files:

eq1.log – text log file
eq1.edr – energy file
eq1.trr – trajectory file
eq1.gro – final minimized structure

- First generate a plot of the potential energy or other terms in function of the simulation time using the energy tool of Gromacs:

```
gmx energy -f eq1.edr -o ene_eq.xvg
```

For example, you can select Potential (Option 11), Total-Energy (Option 13), Volume (Option 20) and then introduce a blank line.

- Plot the resulting file with xmgrace:

```
xmgrace -nxy ene_eq.xvg
```

We can then analyze how the coordinate of the system has change along the equilibration step. If you visualize the eq1.trr with vmd you will observe that the water molecules at the edge of the box seems to be broken. This is not really the case, but since we are working with periodic conditions, the atoms of the molecules at the edge of the water could be on the periodic image.

- To 'correct' this visualization issue we will use the Gromacs trjconv tool (check options typing *trjconv -h*):

```
gmx trjconv -f eq1.trr -s eq1.tpr -pbc mol -o eq1.xtc
```

- If you now visualize the output trajectory (eq1.xtc) generated by trjconv you will see that water molecules are 'wrapped' inside the box. Check the position of the three added ions (residues NA and CL) during the simulation.

```
vmd -gro min.gro -xtc eq1.xtc
```

4. Non-restrained molecular dynamics simulated of solvated peptide

After the waters have equilibrated around the peptide we can perform a non-restrained molecular dynamic. Here, instead of using Berendsen barostat we are going to use Parrinello-Rahman one, that it is able to reproduce the canonical ensemble.

- Copy the appropriate .mdp file and prepare the input for the molecular dynamics simulation using grompp tool. Observe the different keyword chosen for eq1.mdp and prod1.mdp. In this case we are going to read the velocities from the trajectory file of the equilibration run, thus we will not initialize them at the beginning of the run.

```
cp ../provided-files/prod1.mdp .
gmx grompp -f prod1.mdp -c eq1.gro -p AAPAK.top -o prod1.tpr

Start prod1 run.
nohup gmx mdrun -deffnm prod1 >& out_prod1 &
```

After the equilibration it is finished we can check is the peptide conformation is converged to a particular structure or if it oscillates between different ones.

- For that purpose you can compute the root mean square deviation (RMSD) of it along this first run using the g_rms tool:

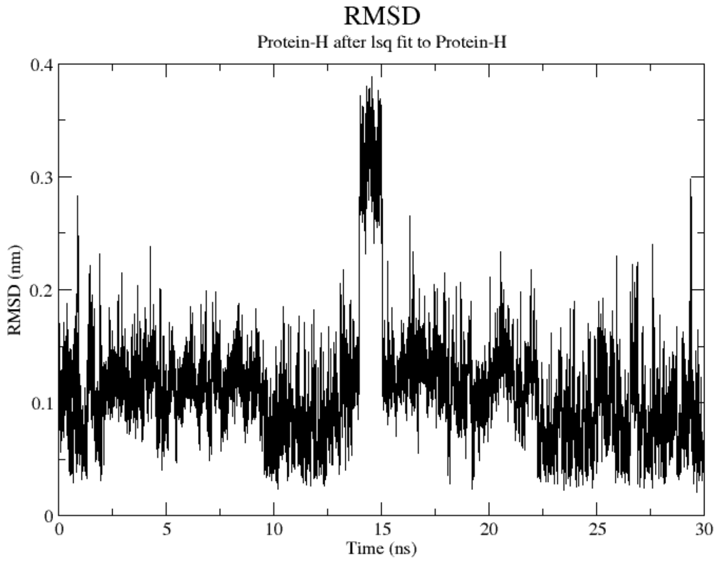
```
gmx rms -s prod1.tpr -f prod1.trr -o rmsd.xvg
```

Choose both for the fitting and the RMSD calculation the protein without the hydrogen atoms (Protein-H, option 2).

- You can plot the final results in the .xvg file

```
xmgrace rmsd.xvg
```

For example in the following plot from a simulation of 30ns of the peptide, most of the time the system samples a conformation at 0.1nm from the initial structure (we used the structural data of prod1.tpr file as a reference in the RMSD calculation), but during a short time at 15ns it adopted a conformation at 0.3-0.4nm of RMSD.



The analysis of a molecular dynamic simulation depends on the hypothesis that you want to test. For example in this case we want to explore the cis-trans isomerization, thus we can measure a dihedral angle that describe this transition.

- Before visualizing the system we will process it with trjconv as we did before for the equilibration run, but also centering it in the protein:

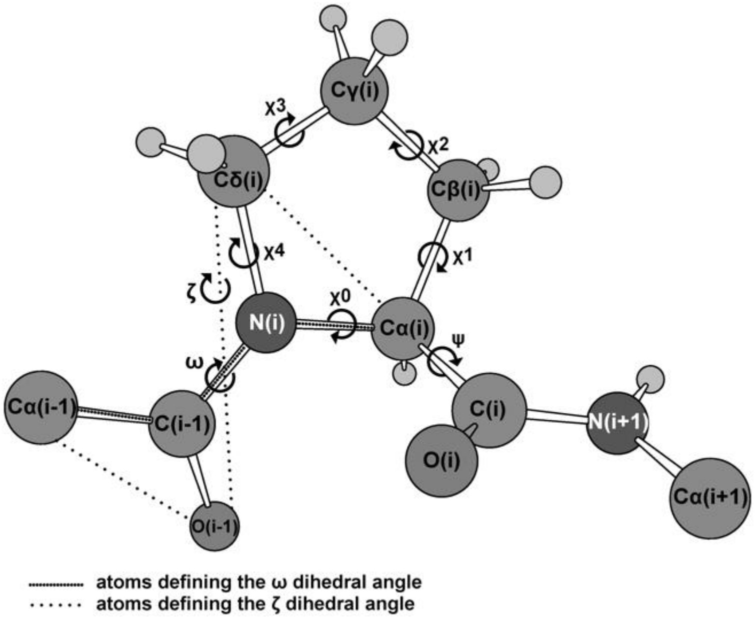
```
gmx trjconv -f prod1.trr -s eq1.tpr -center -pbc mol -o prod1.xtc
```

Select to center the protein (Option 1) and output the whole system (Option 0)

Open the final trajectory with VMD

```
vmd -gro eq1.gro -xtc prod1.xtc
```

Select ‘Mouse’ → ‘Label’ → ‘Dihedral’ from ‘VMD main’ window. Then click in the 4-atoms defined the ζ (improper angle analog to the proper omega torsion angle) dihedral as shown in the picture below (Leone et al. PLoS Comput Biol. 2009).



To see the variation of the dihedral value along the trajectory go to the ‘Labels’ option in the ‘Graphics’ menu. From the scrolling menu select ‘Dihedral’. Select the measured dihedral and within the tab ‘graph’ click on the ‘graph’ button. You should obtain a window with the plotted value of the dihedral along the trajectory frames and you can inspect is the value is changing along the simulation.

- Send a second simulation starting from the prod1 results. The options are identical than prod1, but we will simulate by 100ns overnight, thus we can use this long simulation to initiate the free-energy calculations.

```
cp ../provided-files/prod2.mdp .
gmx grompp -f prod2.mdp -c prod1.gro -e prod1.edr -p AAPAK.top -o prod2.tpr
nohup gmx mdrun -deffnm prod2 >& out_prod2 &
```

Free energy simulations of peptidyl prolyl *cis-trans* isomerization

1. Estimating metadynamics parameter from non-biased molecular dynamics simulation

Before starting the metadynamics simulation we should analyze the collective variables to bias along the unbiases molecular dynamics ran previously. For that purpose, measure the omega-analog angle (ζ) and the ψ angle during the simulation using the provided vmd tcl script measure_dihed.tcl.

- To obtain a trajectory without broken molecules on eq/ folder from the practical1 type:

```
gmx_mpi trjconv -f prod2.trr -s eq1.tpr -center -pbc mol -o prod2.xtc
```

Select option Protein (1) to center the trajectory and System (0) as output

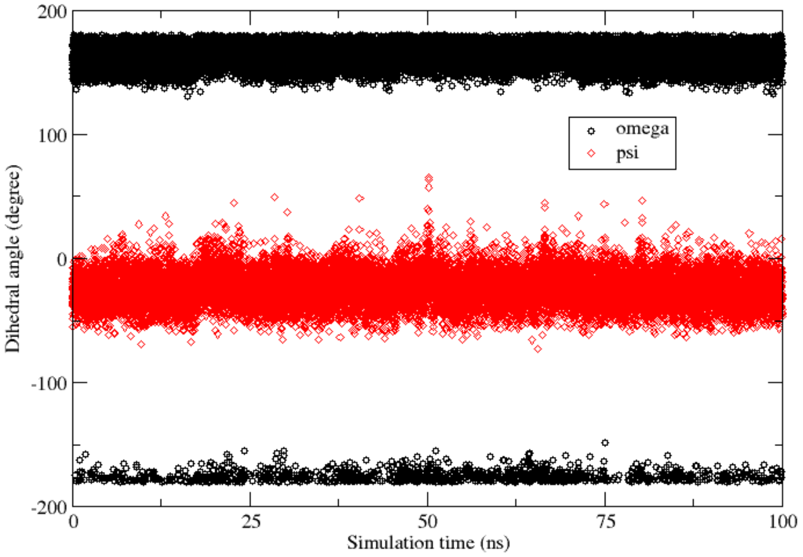
- Measure the dihedral angles along the trajectory with the provided measure_dihed.tcl script:

```
cp ../provided-files/measure_dihed.tcl .
vmd -dispdev text -e measure_dihed.tcl
```

After running the script you should obtain a text file named angles.dat, plot it with xmgrace

```
xmgrace -nxy angles.dat
```

As shown in the figure below, since the barriers for the cis-trans isomerization are high (15-20 kcal/mol) you will observe that in the 100ns molecular dynamics simulation the peptidyl prolyl bond is stuck on the initial trans conformation. Thus we need an enhanced sampling technique to explore this transition and to obtain the relative population between these two conformations.



We will estimate the width of the Gaussian from the standard deviation of the torsion angles. Since the angles are periodic we need a special equation to compute the average value and the standard deviation (average formula from Fisher Statistical Analysis of Circular Data, Cambridge University Press,1993. ISBN 0-521-35018-2 and standard deviation from Mardia 1972 Statistics of Directional Data).

- To compute the average and standard deviation of the ζ angle in radians type:

```
awk -v col=2 '{i++;sine+=sin($col*(3.1415926/180));cosine+=cos($col*(3.1415926/180))}END{sin_av2=(sine/i)**2;cos_av2=(cosine/i)**2;R=sqrt(sin_av2 + cos_av2);av=(atan2(sine,cosine));sigma=sqrt(-2*(log(R)));print av,(sigma)}' angles.dat
```

- To compute the average and standard deviation of the ψ angle in radians type:

```
awk -v col=3 '{i++;sine+=sin($col*(3.1415926/180));cosine+=cos($col*(3.1415926/180))}END{sin_av2=(sine/i)**2;cos_av2=(cosine/i)**2;R=sqrt(sin_av2 + cos_av2);av=(atan2(sine,cosine));sigma=sqrt(-2*(log(R)));print av,(sigma)}' angles.dat
```

- Obtain the atom numbers of each torsion angle by running the provided tcl script (get_serial.tcl)

```
cp ../provided-files/get_serial.tcl .
vmd -dispdev text -e get_serial.tcl
```

Keep a record of all these numbers, that we will use afterwards.

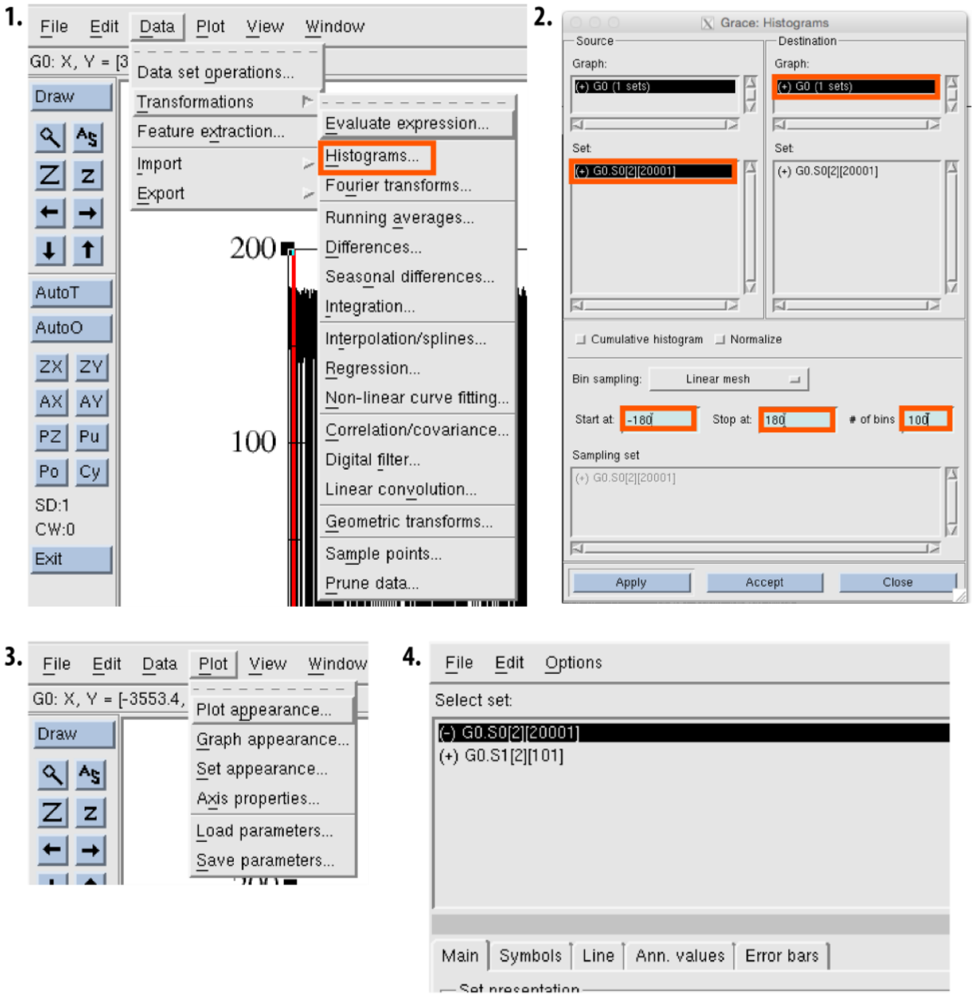
2. Compute a histogram for the ζ torsion angle

We will compute a histogram of the ζ torsion angle with xmgrace.

- Open the angle.dat file with xmgrace

```
xmgrace angles.dat
```

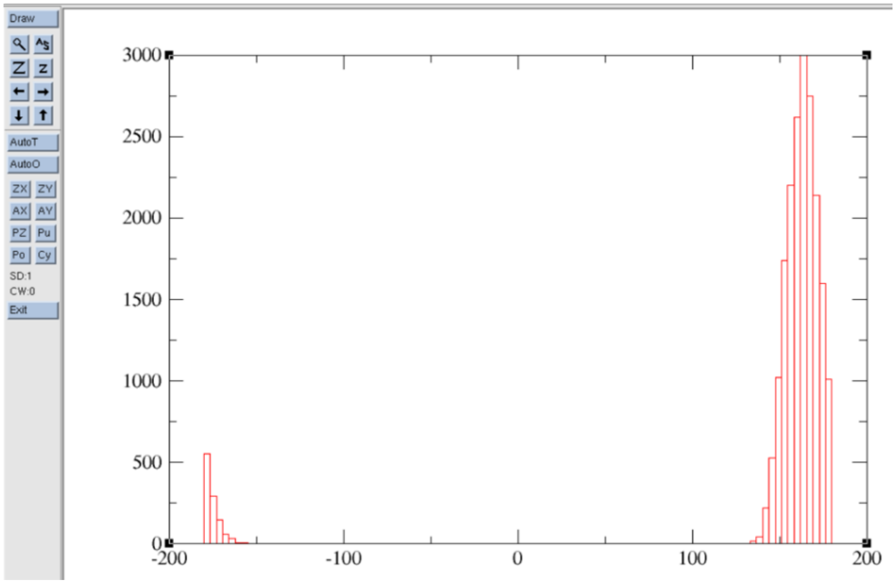
As shown in the panel 1. of the figure below, you can create a histogram from the plotted data by simply selecting ‘Histogram’ in the ‘Data’→ ‘Transformation’ menu.



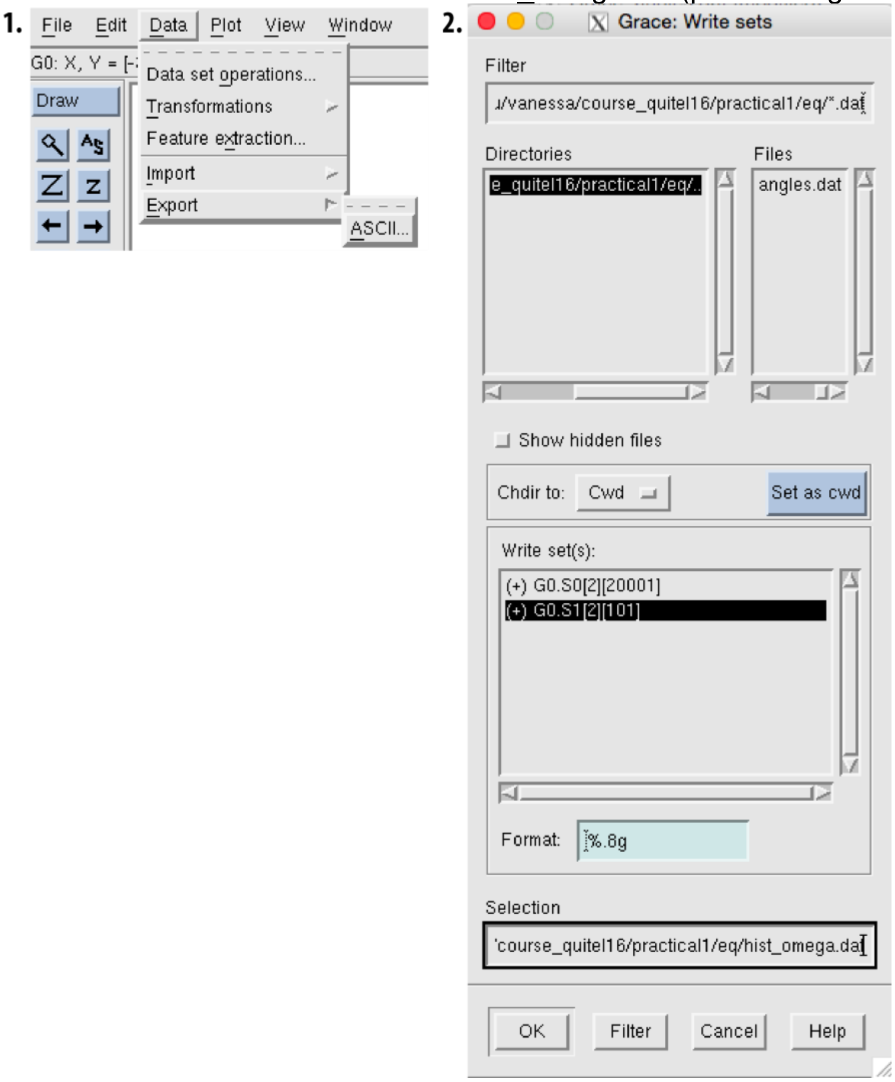
Source Graph: (+) G0 (1 sets)
Source Set: (+) G0.S0[2][20001]
Destination Graph: (+) G0 (1 sets)
Destination Set: (+) G0.S0[2][20001]
□ Cumulative histogram □ Normalize
Bin sampling: Linear mesh
Start at: -180 Stop at: 180 # of bins: 100
Sampling set: (+) G0.S0[2][20001]

Select set:
(-) G0.S0[2][20001]
(+) G0.S1[2][101]

Within the ‘Histogram’ window (panel 2. of figure above) select the histogram to start at -180 and finish at +180, and to make a total of 100 bins. Select the plotted data set to use as the histogram input data and the graph G.0 (**but not the set!**) as the output. Click the ‘Accept’ button when you entered all these options. After finishing the histogram we will hide the raw data, by clicking ‘Plot’→’Set appearance’ (panel 3. of the figure above), R-click on G0.SO set and then select ‘hide’(panel 4. of figure above). Afterwards click on the ‘As’ (autoscale) button on the action menu of xmgrace (under ‘Draw’). You should obtain an histogram as the one shown in the figure below.



Save the generated histogram by clicking in ‘Data’→’Export’→’ASCII’ (panel 1. Figure below). On the ‘write sets’ windows save the file as hist_omega.dat (panel 2. Figure below)



Filter: ı/vanessa/course_quitel16/practical1/eq/*.dat

Directories: e_quitel16/practical1/eq/..

Files: angles.dat

☐ Show hidden files

Chdir to: Cwd | Set as cwd

Write set(s):

(+) G0.S0[2][20001]

(+) G0.S1[2][101]

Format: %.8g

Selection: 'course_quitel16/practical1/eq/hist_omega.dat

OK | Filter | Cancel | Help

We can compute the free energy from the previously computed histogram using the equation:
G=-RTlnP

- For this purpose, we will use the following awk script:

```
awk '{if($2>0)print $1,-2.493*log($2)}' hist_omega.dat > free_omega.dat
```

- Plot the resulting free energy with xmgrace

```
xmgrace free_omega.dat
```

3. Bidimensional metadynamics simulation

We will use PLUMED2 (Tribello et al. 2014, The PLUMED consortium 2019) coupled with GROMACS to bias both ζ and ψ torsion angle. PLUMED is a plugin to perform large variety of enhanced sampling simulations that can be run with several widely used molecular dynamics engines (i.e. GROMACS, NAMD, AMBER, LAMMPS).

- In the practical2/ folder create metad/ folder and copy the system topology and final conformation and energy files obtained from the 100ns molecular dynamics simulation that you run before.

```
mkdir metad
cd metad/
cp ../../practical1/eq/prod2.gro .
cp ../../practical1/eq/prod2.edr .
cp ../../practical1/eq/AAPAK.top .
```

As you did to set up the molecular dynamics simulation you need a .tpr file to run the metadynamics. Here we will use the same parameters that we used for the unbiased simulation. We also need an input for PLUMED, that should indicate the collective variables parameters and the biasing method that you will use.

- Copy the .mdp file and the plumed input file, plumed.dat, from the provided-files/ folder:

```
cp ../provided-files/metad.mdp .
cp ../provided-files/plumed.dat .
```

Explore the options of the plumed input (less plumed.dat), the explanation of each option has been added with comments.

- Check if the atom numbers that define the two torsion angles are correct for the system that you are simulating and in case change them:

```
omega: TORSION ATOMS=19,26,28,31
psi: TORSION ATOMS=27,31,39,41
```

- In the metad line change the SIGMA values to the standard deviation of each torsion angle that you computed in the unbiased simulation

```
metad: METAD ARG=omega,psi PACE=1000 HEIGHT=0.2 SIGMA=0.17,0.24 FILE=HILLS
```

The following lines instruct plumed to write collective variable and applied forces every 500 steps

```
PRINT STRIDE=500 ARG=omega,psi,metad.bias FILE=COLVAR
DUMPFORCES ARG=omega,psi STRIDE=500 FILE=forces.dat
```

- Run the grompp tool to generate a .tpr file for the metadynamics run:

```
gmx_mpi grompp -f metad.mdp -c prod2.gro -e prod2.edr -p AAPAK.top -o metad.tpr
```

- Run the metadynamics simulation using mdrun and adding a flag to read the plumed input:

```
gmx_mpi mdrun -deffnm metad -plumed plumed.dat >& meta.out &
```

The calculation should write the Gaussians into HILLS files and the values of the collective variable to the files COLVAR along with the applied biasing forces in forces.dat. Check if these files are being generated.

4. Bias-exchange metadynamics with two unidimensional replicas

While the plain bidimensional metadynamics is running we will set up a bias-exchange metadynamics, with two replicas, and in each of this replicas we will bias one of the torsion angles. In this case a bias-exchange approach may not give a great advantage to our simulation In more complex systems, this technique could be very useful where more than two variables are slow, or you do not know which is the most representative collective variable of the simulated transition. On the third protocol you will observe that the WHAM analysis of the unidimensionally biased replicas should give a similar result than the bidimensional metadynamics.

- Create a folder to run the bias exchange metadynamics and copy all the files needed to setup the simulation:

```
mkdir metad_be
cd metad_be
cp ../../practical1/eq/prod2.gro .
cp ../../practical1/eq/prod2.edr .
cp ../../practical1/eq/AAPAK.top .
cp ../provided-files/metad.mdp .
cp ../provided-files/plumed_be-common.dat .
```

Create two folders to run each bias exchange walker and copy respective plumed file in each:

```
mkdir 0
mkdir 1
cp ../provided-files/plumed.0.dat 0/plumed.dat
cp ../provided-files/plumed.1.dat 1/plumed.dat
```

For running the bias exchange with two replicas we need 3 PLUMED files; one common (here we named it plumed_be-common.dat) and two other files for each metadynamics replica (plumed.0.dat and plumed.1.dat). If you open the common file (less plumed_be-common.dat) you can observe that it contains the definition of both torsion angles, change the atom indices to match the ones of the system you are simulating. The RANDOM_EXCHANGES command indicates to PLUMED to exchange the replicas randomly, in this example we only have two replicas, but if you have more you need to include this command.

```
# randomize the exchange (not between consecutive indeces, here just 2 replicas, so it is not relevant)
RANDOM_EXCHANGES
# set up the two torsion collective variables
# omega (use cv1 so it will be compatible with the METAGUI analysis plugin)
cv1: TORSION ATOMS=19,26,28,31
# psi (use cv1 so it will be compatible with the METAGUI analysis plugin)
cv2: TORSION ATOMS=27,31,39,41
```

The 0/plumed.dat and 1/plumed.dat are very similar to the plumed file used in the plain metadynamics example, but the bias is applied to only one of the colvars. Check the difference between the two files (vimdiff 0/plumed.dat 1/plumed.dat).

- Run the grompp tool to generate a .tpr file for the bias-exchange metadynamics run (you could also use the .tpr that you have generated for the bidimensional metadynamics run):

```
gmx_mpi grompp -f metad.mdp -c prod2.gro -e prod2.edr -p AAPAK.top -o metad-be.tpr
```

- copy the.tpr file and the respective plumed file in each replica folder:

```
cp metad-be.tpr 0/
cp metad-be.tpr 1/
```

- To run the bias-exchange metadynamics type:

```
mpirun -np 4 gmx_mpi mdrun -deffnm metad-be -plumed plumed.dat -multidir ? -replex 2000 >& meta-be.out &
```

The option -replex indicates the frequency of the attempted exchanges (here 2000 time steps) and -multidir ? indicates folders containing each bias exchange walker or replica (here 2).

The calculation should write the Gaussians into HILLS.0, HILLS.1 files, the values of the collective variable to the files COLVAR.0, COLVAR.1 and the applied forces into forces.0.dat and forces.1.dat. Check if these files are being generated.

Analysis of free energy simulations of peptidyl prolyl *cis-trans* isomerization

1. Analysis of bi-dimensional metadynamics simulations

- Within the practical3/ folder create a folder to analyze the metadynamics simulation that you performed before and within that create a folder to analyze its convergence. Then copy the HILLS file to that folder.

```
cd practical3/
mkdir metad-analysis/
mkdir metad-analysis/convergence
cd metad-analysis/convergence
cp ../../../practical2/metad/HILLS .
```

You can compute the bidimensional free energy profile summing the Gaussians deposited during the simulation and stored in the HILLS file. However, before, you need to know if these free energy profiles are converged. For this purpose, we will compute the free energy at different simulation time; when convergence is achieved the plot should not change even if you continue the simulation. Since it is easier to visualize the convergence with a unidimensional plot, than with a bidimensional one, we will compute the two free energy plots for each variable, integrating the other variable.

- To compute the free energy plot in function of omega we will use the sum_hills tool:

```
plumed sum_hills --hills HILLS --idw omega --kt 2.49 --stride 1000 --outfile fes_omega
```

--kt specifies the kT used to integrate the ψ variable (300K), the --stride indicates that the whole HILLS file will be divided every 1000 HILLS.

You should obtain several files named fes_omega0.dat, fes_omega1.dat, etc. Each file is a free energy plot summing incremental number of deposited Gaussians. Plot the files to explore if the metadynamics is converged on the ζ angle (xmgrace fes_omega*.dat). If it is converged, you should observe that the last free energy profiles are very similar.

- In an analogous manner check the convergence of the free energy profile along ψ torsion angle:

```
plumed sum_hills --hills HILLS --idw psi --kt 2.49 --stride 1000 --outfile fes_psi
```

- Once we determined that the free energy profile is reasonably converged calculate bi-dimensional free energy energy profiles every 1000 HILLS.

```
cd ../
mkdir fes
cd fes
cp ../../..//practical2/metad/HILLS .
plumed sum_hills --hills HILLS --stride 1000 --outfile fes_2D
```

- As for the 1D profiles you should obtain several files named fes_2D0.dat, fes_2D1.dat, etc. Each file is a free energy plot summing incremental number of deposited Gaussians. From the 1D profiles determine from which file the free energy start to converge. Convergence is achieved once the profiles change mostly by a shift constant. In this case after approximately 30000 hills the free energy is converged, that would be files fes_2D30.dat, fes_2D31.dat etc.

- You can plot one of the 2D profiles with gnuplot:

```
gnuplot
```

```
gnuplot> set pm3d
gnuplot> splot ‘fes_2D40.dat’ u 1:2:3 ls 0
```

- The best estimate of the free energy with metadynamics is given by the average of the free energy after convergence (Marinelli et al. PloS Comp Biol 2009). Here it is a simple script (that you will find also in your metad/ folder; average_fes.sh) to calculate the average free energy (time average of the bias potential after convergence).

- Take 2D profiles from 30 and align them by subtracting the average value of the free energy for each file:

```
for i in {30..62} ; do ref=`awk '{if($1!="#!"&&NF>0){i++;av+=$3}}END{print av/i}' fes_2D${i}.dat`; awk -v
ref=${ref} '{if($1!="#!"&&NF>0){print $1,$2,$3-ref,$4,$5}else{print $0}}' fes_2D${i}.dat >
fes_2D${i}_ref.dat ; done
```

- Just enumerate files for paste:

```
files=`for i in {30..62} ; do echo fes_2D${i}_ref.dat ;done`
```

- Paste all files and calculate average 2D profile:

```
paste $files | awk 'BEGIN{j=0;free=0}{if($1!="#!"&&NF>0){for(i=3;i<=NF;i+=5){j++;free+=$i};print
$1,$2,free/j,$4,$5;free=0;j=0}if(NF==0) print " "}' > fes_2D_ave.dat
```

- Average is now calculated, to visualize the profile better subtract the value of the minimum so that the profile goes from 0 to the maximum value. To do this first calculate the minimum:

```
min=`awk '{if(NF>0){i++;if(i==1) min=$3;else{if($3<min) min=$3}}}END{print min}' fes_2D_ave.dat`
```

- Then subtract minimum to free energy:

```
awk -v min=$min '{if(NF>0){print $1,$2,$3-min,$4,$5}else{print $0}}' fes_2D_ave.dat > pippo; mv pippo
fes_2D_ave.dat
```

- The final free energy is now ready to be visualized with gnuplot:

```
gnuplot
```

- In the gnuplot windows enter the following command

```
gnuplot> set pm3d
gnuplot> splot ‘fes_2D_ave.dat’ u 1:2:3 ls 0
```

Compare the energy of the different minima and transition states of the free energy surface. Which is the lower trans minimum? Which is the lower cis minimum? Which is the lowest transition state between the two?

Error estimate on the free energy
A simple estimate of the error on the free energy can be calculated comparing the average profiles of two halves of the simulation (files fes_2D30.dat… fes_2D46.dat):

- Just enumerate files in the first half for paste:

```
filesfh=`for i in {30..46} ; do echo fes_2D${i}_ref.dat ;done`
```

- Paste all files and calculate average 2D profile of the first half of the simulation:

```
paste $filesfh | awk 'BEGIN{j=0;free=0}{if($1!="#!"&&NF>0){for(i=3;i<=NF;i+=5){j++;free+=$i};print
$1,$2,free/j,$4,$5;free=0;j=0}if(NF==0) print " "}' > fes_2D_ave_FH.dat
```

- Do the same for the second half of the simulation (files fes_2D46.dat… fes_2D62.dat)

```
filessh=`for i in {47..62} ; do echo fes_2D${i}_ref.dat ;done`
```

```
paste $filessh | awk 'BEGIN{j=0;free=0}{if($1!="#!"&&NF>0){for(i=3;i<=NF;i+=5){j++;free+=$i};print
$1,$2,free/j,$4,$5;free=0;j=0}if(NF==0) print " "}' > fes_2D_ave_SH.dat
```

- Now calculate error subtracting free energy of first half and second half:

```
paste fes_2D_ave_FH.dat fes_2D_ave_SH.dat | awk '{if(NF>0) print $1,$2,sqrt(($3-$8)*($3-$8));else
print $0}' > fes_2D_error.dat
```

- Now plot the error:

```
gnuplot
```

- In the gnuplot windows enter the following command

```
gnuplot> set pm3d
gnuplot> splot ‘fes_2D_error.dat’ u 1:2:3 ls 0
```

In alternative to the average 2D profile obtained from the bias potential, a robust estimate of the free energy can be also calculated using the force-correction analysis method (FCAM, Marinelli et al. JCTC 2021). To do so, download the python tool implementing FCAM via:

```
git clone https://github.com/FCAM-NIH/FCAM.git
```

copy and inspect the input file to calculate mean forces based on FCAM:

```
cp ../../provided-files/input_metad_fcam.dat .
```

The file contains information on the grid specification to calculate the mean forces and required files, such as the COLVAR files and the files containing the applied forces (which should be written in ouput the same pace):

```
COLVAR_FILE COLVAR
CV-CL 1 -3.14159265359  3.14159265359 144 PERIODIC
CV-CL 2 -3.14159265359  3.14159265359 144 PERIODIC
READ_BIAS_FORCE_TRJ forces_500.dat
```

Calculate the free energy gradient or negative of the mean forces:

```
python FCAM-FOLDER/calcf_vgauss.py -if input_metad_fcam.dat -units kj -temp 300 > job&
```

Integrate the mean forces to obtain the free energy, fes.out, over 10000000000 Kinetic Monte Carlo (KMC) steps:

```
python  FCAM-FOLDER/graf_fes_kmc.py  -ff  grad_on_eff_points.out  -units  kj  -temp  2300  -nsteps 10000000000  -ofesf fes.out > job &
```

The 2D free energy, fes.out, can be visualized using gnuplot:

```
gnuplot> spl ' fes.out' u 1:2:3 w p palette
```

Error analysis can be obtained by calculating mean forces and free energies for different simulation time intervals via the options -trfr1 and -trfr2 of the program calcf_vgauss.py. Additional details can be found in https://github.com/FCAM-NIH/FCAM.

2. Analysis of bias-exchange metadynamics simulations

- Again here you should first check the convergence of the unidimensional free energy plots:

```
cd practical3
mkdir be-analysis/
mkdir be-analysis/convergence
cd be-analysis/convergence
mkdir cv1
cd cv1
plumed sum_hills --hills ../../../../practical2/metad_be/0/HILLS.0 --stride 1000
xmgrace fes_*.dat
cd ..
mkdir cv2
cd cv2
plumed sum_hills --hills ../../../../practical2/metad_be/1/HILLS.1 --stride 1000
xmgrace fes_*.dat
cd ..
```

We will use the information contained in the two unidimensional replicas of the bias-exchange metadynamics to reconstruct a bi-dimensional free energy surface using first the weighted histogram analysis method implemented in METAGUI3 (Marinelli et al. PloS Comp Biol 2009, Giorgino at al. Computer Physics Communications 2017) and then the force correction analysis method (Marinelli at al. JTCT 2021). METAGUI3 is a plugin of the VMD visualization program. This tool can be use to first cluster the trajectories based on the biased collective variables and then assigns a free energy to each cluster applying a WHAM procedure to the 1D biasing potentials.

- To run the METAGUI we need that the COLVAR files (COLVAR.0 and COLVAR.1) are synchronized to the trajectory. Therefore, we are going to generate the COLVAR file with the coordinates in the trajectory file (after copying all needed files from the metad-be/ folder):

```
cd be-analysis
cp ../metad-be/* .
cp ../provided-files/plumed.dat .
plumed driver --plumed plumed.dat --mf_trr 0/metad-be.trr --timestep 0.002 --trajectory-stride 1000 > log0
mv COLVAR COLVAR.0
plumed driver --plumed plumed.dat --mf_trr 1/metad-be.trr --timestep 0.002 --trajectory-stride 1000 > log1
mv COLVAR COLVAR.1
```

- To make colvar files (COLVAR.0 and COLVAR.1) compatible with METAGUI3 plugin modify the header as shown below (use a text editor to open the colvar files, e.g. *vi COLVAR.0*):

Original COLVAR.0	COLVAR.0 compatible with METAGUI
#! FIELDS time omega psi metad.bias #! SET min_omega -pi #! SET max_omega pi #! SET min_psi -pi #! SET max_psi pi 0.000000 2.546200 -0.403785 0.000000	#! FIELDS time cv1 cv2 #! ACTIVE 1 1 A 0.000000 2.546200 -0.403785 0.000000

Original COLVAR.1	COLVAR.1 compatible with METAGUI
#! FIELDS time omega psi metad.bias #! SET min_omega -pi #! SET max_omega pi #! SET min_psi -pi #! SET max_psi pi 0.000000 2.546200 -0.403785 0.000000	#! FIELDS time cv1 cv2 #! ACTIVE 1 2 B 0.000000 2.546200 -0.403785 0.000000

Both COLVAR files must contain all biased variables (remember to indicate to print all of them in the plumed.dat) and they must be named cv1, cv2...cvN.
The line #! ACTIVE ${num_of_biased_cv} ${biased_cv_number} ${label} is needed; the ${num_of_biased_cv} indicates the number of CV biased **in the current replica (not over all the bias-exchange replicas)**. The ${label} is a letter that identify the replica, you can specify simply A for the first replica, B for the second, etc.

- Add the following header to the hills files (HILLS.0 and HILLS.1) to make them compatible with METAGUI plugin (use a text editor to open the hills files, e.g. *vi HILLS.0*):

HILLS.0
#! ACTIVE 1 1 A

HILLS.1
#! ACTIVE 1 2 B

The HILLS file do not need to be synchronized with the COLVAR and trajectory files, but the time needs to be consistent.
For visualization purposes, wrap trajectories within the box with the protein centered in the middle. This can be obtained with trjconv:

```
echo 1 0 | gmx trjconv -f 0/metad-be.trr -s ../../practical2/metad_be/metad-be0.tpr -pbc mol -center -o metad-be0.xtc
echo 1 0 | gmx trjconv -f 1/metad-be.trr -s ../../practical2/metad_be/metad-be0.tpr -pbc mol -center -o
```

```
metad-be1.xtc
```

It is also convenient to copy a gro file to load trajectories with VMD:

```
cp ../../practical1/min/min.gro
```

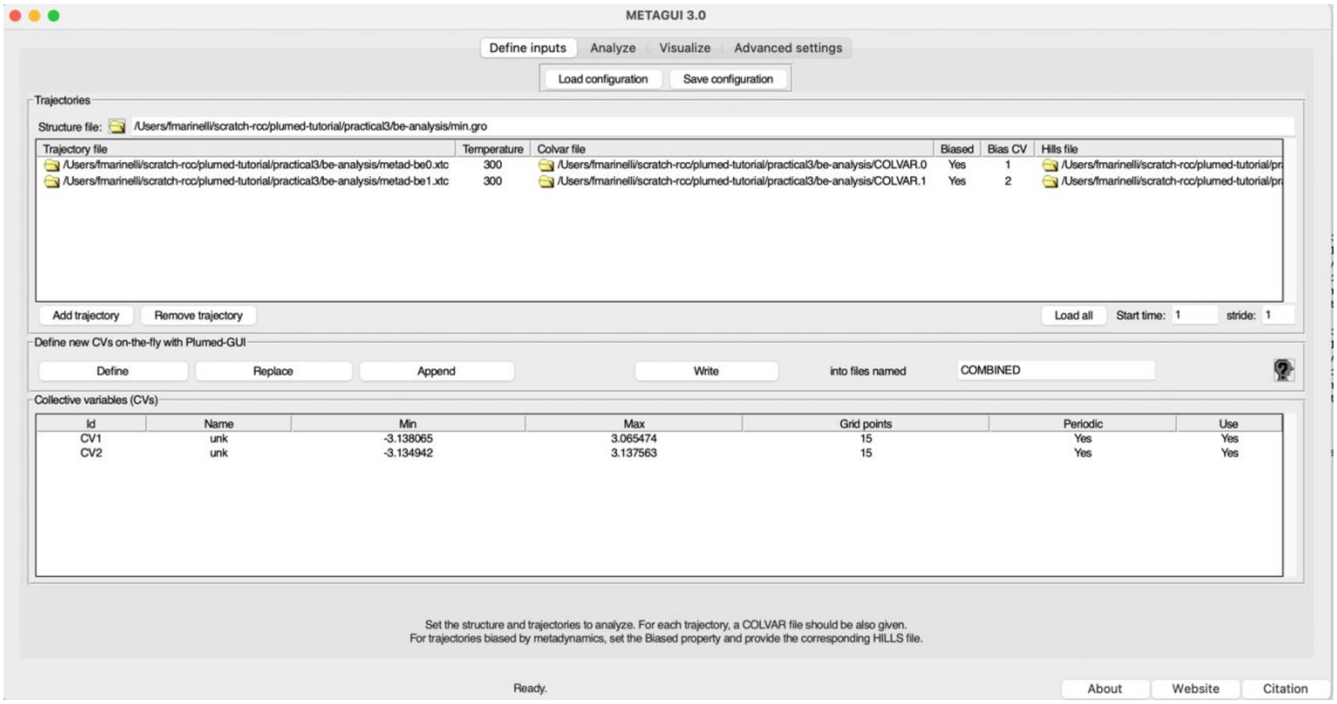
The metagui-input file (shown below) indicate all the options needed to be specified for the METAGUI plugin as the temperature (in energy units), where to find the COLVAR, HILLS, trajectory files and a reference .pdb file needed to load the trajectories. Also the range and the number of bins for each available variable are specified. You could also generate the input file by adding each of the option in the METAGUI windows on VMD.

```
# METAGUI_EXE METAGUI-FOLDER/metagui_util.x
KT 2.4943
CLUSTER_TYPE grid
FES_RANGE 100
GRO_FILE /Users/fmarinelli/scratch-rcc/plumed-tutorial/practical3/be-analysis/min.gro
COLVAR_FILE            /Users/fmarinelli/scratch-rcc/plumed-tutorial/practical3/be-analysis/COLVAR.0
/Users/fmarinelli/scratch-rcc/plumed-tutorial/practical3/be-analysis/metad-be0.xtc        300         1
/Users/fmarinelli/scratch-rcc/plumed-tutorial/practical3/be-analysis/0/HILLS.0
COLVAR_FILE            /Users/fmarinelli/scratch-rcc/plumed-tutorial/practical3/be-analysis/COLVAR.1
/Users/fmarinelli/scratch-rcc/plumed-tutorial/practical3/be-analysis/metad-be1.xtc        300         1
/Users/fmarinelli/scratch-rcc/plumed-tutorial/practical3/be-analysis/1/HILLS.1
TRAJ_SKIP 1
CVGRID  1 -3.138065 3.065474 15 unk PERIODIC
CVGRID  2 -3.134942 3.137563 15 unk PERIODIC
ACTIVE 2  1 2
T_CLUSTER 1
T_FILL 1
DELTA 1
GCORR 1
TR_N_EXP 5
SIEVING 1
NUMBER_STRUCTURES_SIEVING 5000
MEDOIDS_CLUSTERS 100
GROMACS_CUTOFF 1.0
```

- Open the vmd with the plugin

```
vmd -e run_metagui.tcl
```

On the menu ‘File’, click on ‘Load configuration’. Choose the metagui-input file. If everything has been loaded correctly you should have a window as the one shown below.



Trajectories

Structure file: /Users/fmarinelli/scratch-rcc/plumed-tutorial/practical3/be-analysis/min.gro

Trajectory file	Temperature	Colvar file	Biased	Bias CV	Hills file
/Users/fmarinelli/scratch-rcc/plumed-tutorial/practical3/be-analysis/metad-be0.xtc	300	/Users/fmarinelli/scratch-rcc/plumed-tutorial/practical3/be-analysis/COLVAR.0	Yes	1	/Users/fmarinelli/scratch-rcc/plumed-tutorial/pr
/Users/fmarinelli/scratch-rcc/plumed-tutorial/practical3/be-analysis/metad-be1.xtc	300	/Users/fmarinelli/scratch-rcc/plumed-tutorial/practical3/be-analysis/COLVAR.1	Yes	2	/Users/fmarinelli/scratch-rcc/plumed-tutorial/pr

Start time: 1 stride: 1

Define new CVs on-the-fly with Plumed-GUI

into files named COMBINED

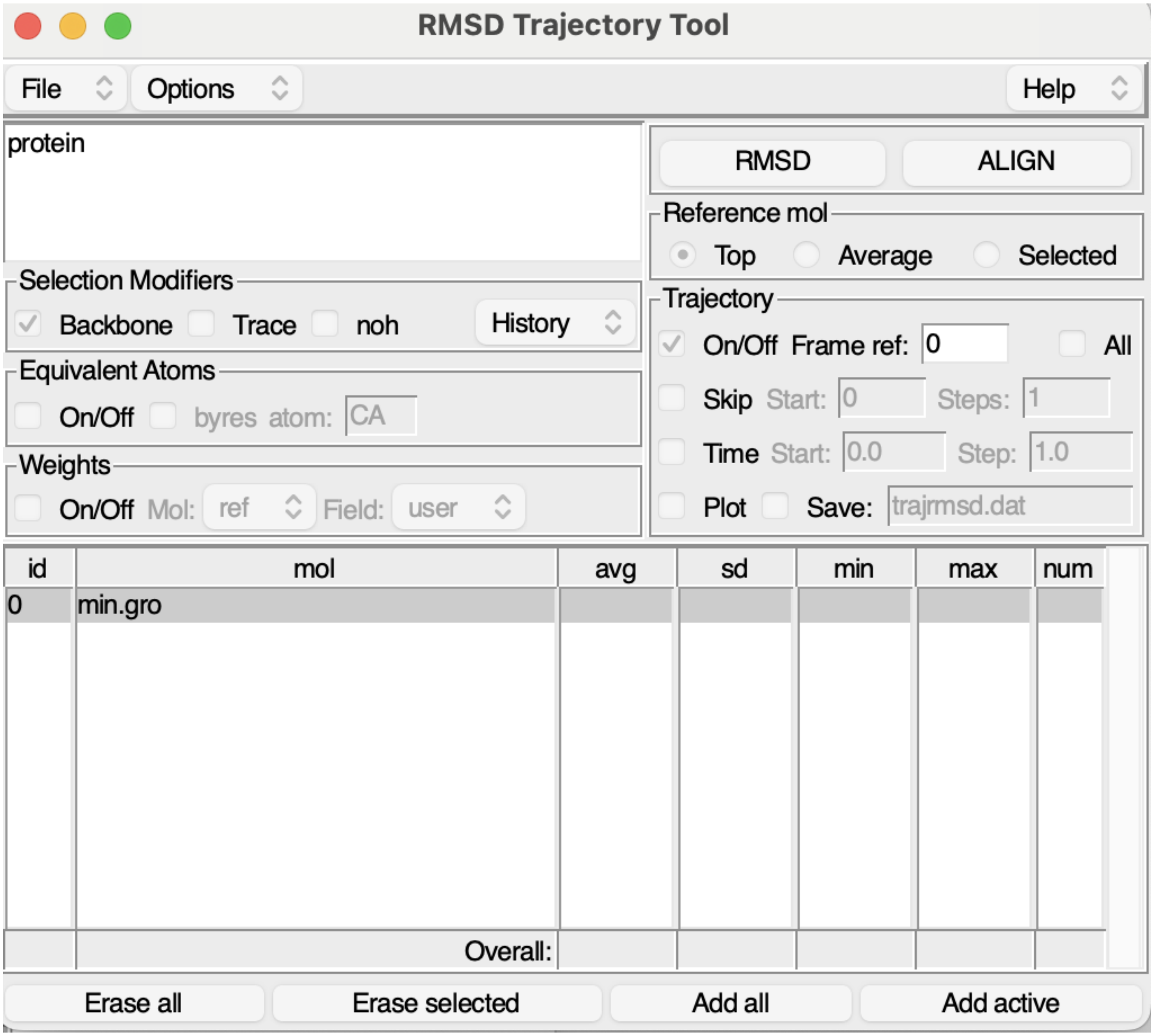
Collective variables (CVs)

Id	Name	Min	Max	Grid points	Periodic	Use
CV1	unk	-3.138065	3.065474	15	Yes	Yes
CV2	unk	-3.134942	3.137563	15	Yes	Yes

Set the structure and trajectories to analyze. For each trajectory, a COLVAR file should be also given.
For trajectories biased by metadynamics, set the Biased property and provide the corresponding HILLS file.

Ready.

- After the window has loaded the metagui-input click on ‘Load All’ button.
- We will align all the frames of the trajectories for visualization with VMD. On the ‘VMD main’ windows click on Extensions→ Analysis → RMSD trajectory tool. Select ‘backbone’ and then click on ‘Align’ as shown in the figure below.



protein

Selection Modifiers

☑ Backbone □ Trace □ noh

Equivalent Atoms

□ On/Off □ byres atom: CA

Weights

□ On/Off Mol: ref Field: user

Reference mol

● Top ○ Average ○ Selected

Trajectory

☑ On/Off Frame ref: 0 □ All

□ Skip Start: 0 Steps: 1

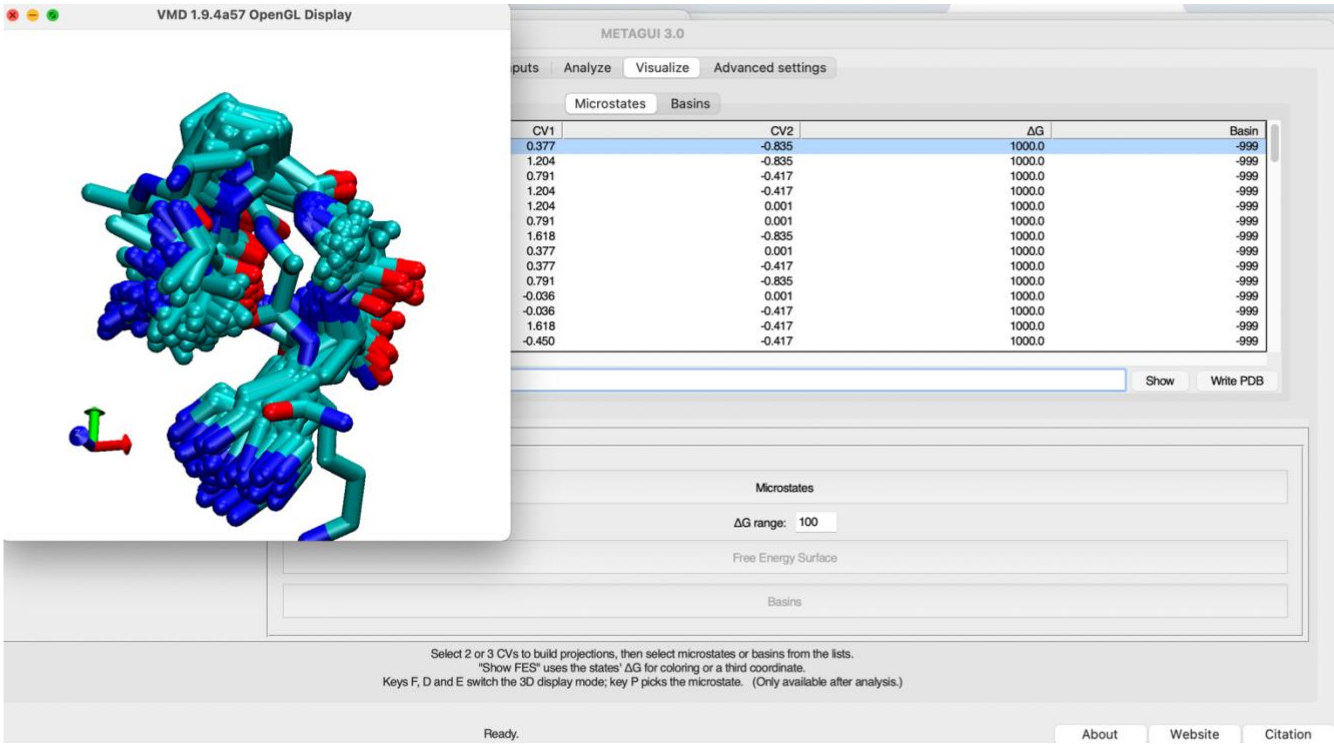
□ Time Start: 0.0 Step: 1.0

□ Plot □ Save: trajrmsd.dat

id	mol	avg	sd	min	max	num
0	min.gro					
	Overall:					

Go back to menu file, and click on analyze, microstates and then find microstates. This will divide the space of the omega and psi collective variables in bins with size 2pi/15 (the 15 has been selected by you on the grid box for both variables). As shown below you should have a list for all microstates on the ‘Visualize section.

-To visualize the microstates, click on ‘show miscrostates on select’, select one of them and finally click on the ‘show’ button.



CV1	CV2	ΔG	Basin
0.377	-0.835	1000.0	-999
1.204	-0.835	1000.0	-999
0.791	-0.417	1000.0	-999
1.204	-0.417	1000.0	-999
1.204	0.001	1000.0	-999
0.791	0.001	1000.0	-999
1.618	-0.835	1000.0	-999
0.377	0.001	1000.0	-999
0.377	-0.417	1000.0	-999
0.791	-0.835	1000.0	-999
-0.036	0.001	1000.0	-999
-0.036	-0.417	1000.0	-999
1.618	-0.417	1000.0	-999
-0.450	-0.417	1000.0	-999

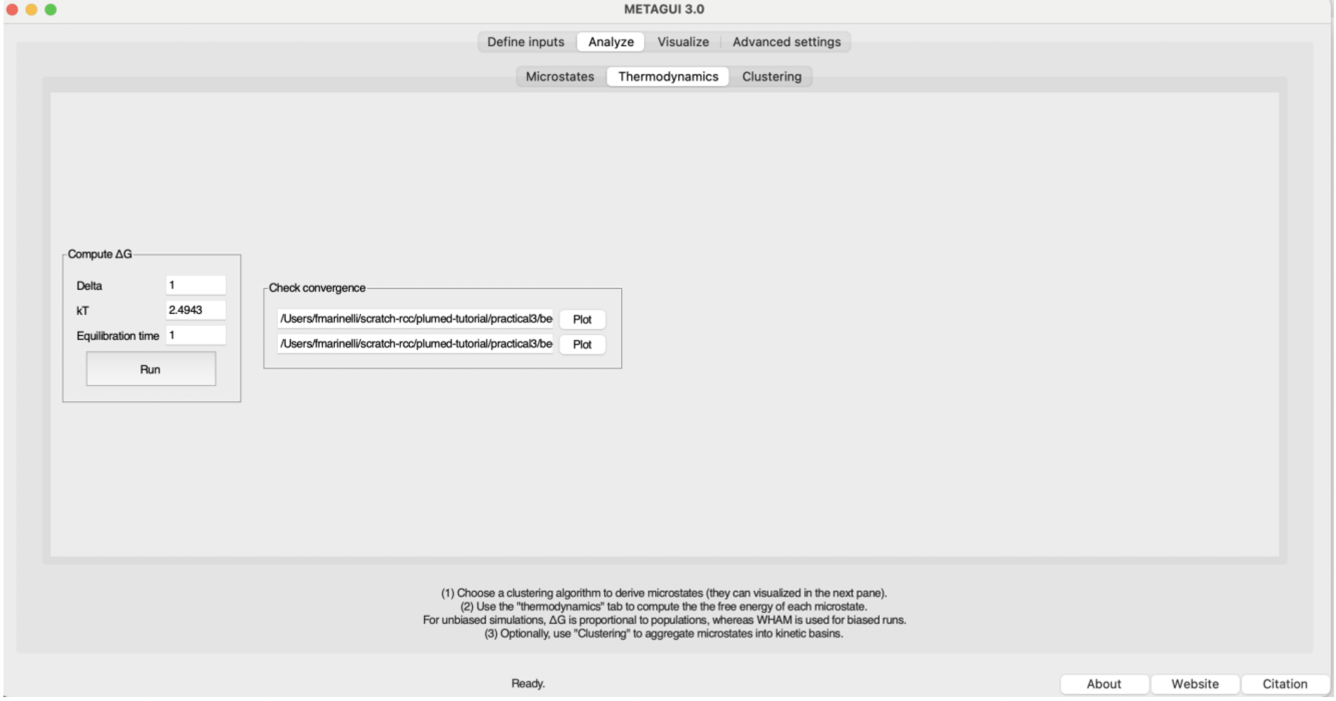
ΔG range: 100

Select 2 or 3 CVs to build projections, then select microstates or basins from the lists.
"Show FES" uses the states' ΔG for coloring or a third coordinate.
Keys F, D and E switch the 3D display mode; key P picks the microstate. (Only available after analysis.)

Ready.

In the VMD display window you should have all the structures that belong to the selected microstate.

- If you click on analyze, thermodynamics and then 'Run' the program will calculate the free energy for each microstate (in this case it is equivalent to the bi-dimensional profile in function of omega and psi torsion angles).



Delta 1
kT 2.4943
Equilibration time 1

/Users/fmarinelli/scratch-rcc/plumed-tutorial/practical3/be
/Users/fmarinelli/scratch-rcc/plumed-tutorial/practical3/be

(1) Choose a clustering algorithm to derive microstates (they can visualized in the next pane).
(2) Use the "thermodynamics" tab to compute the the free energy of each microstate.
For unbiased simulations, ΔG is proportional to populations, whereas WHAM is used for biased runs.
(3) Optionally, use "Clustering" to aggregate microstates into kinetic basins.

Ready.

When this calculation is finished you will observe that in the 'Free en' column you should have the free energy value and not '1000.' as before. The free energy computed by the WHAM analysis is based on the bias potential (accumulated Gaussians) after the specified equilibration time (in the metagui window is indicated in the 'Equil. Time for VG' box).

Before continuing the analysis with the metagui plugin you will compare the obtained free energy by the

WHAM analysis against the free energy computed from the bi-dimensional metadynamics simulation. All the information of the microstates are stored on the MICROSTATES file.

- Copy to the present folder the fes_2D_ave.dat file that you have generated during the bi-dimensional analysis

```
cp ../metad-analysis/fes/fes_2D_ave.dat .
```

- Open gnuplot

```
gnuplot
```

- In the gnuplot windows enter the following command

```
gnuplot> spl [-3.14159:3.14159][][0:80]'fes_2D_ave.dat' u 1:2:3 w p pt 7,"MICROSTATES" u 3:4:5 w p pt 7
```

Is the reconstructed free energy plot of the bias exchange similar to the one of the bi-dimensional metadynamics?

You should be able to visualize the free energy landscape by going to visualize, microstates and then clicking on Free energy surface. In this representation each microstate corresponds to a sphere and the bigger ones are the lowest free energy microstate.

To visualize the structures within each microstate, click on 'p' on your keyboard and then click on the microstate sphere of interest. To go back to the free energy landscape, click 'f' again.

2. Calculating the free energy of bias exchange metadynamics using mean forces

The 2D free energy from bias exchange metadynamics simulation can also be obtained using FCAM. To do so it is required an input file with grid specification, and location of COLVAR and applied force files having the following format:

```
COLVAR_FILE ../0/COLVAR.0
COLVAR_FILE ../1/COLVAR.1
CV-CL 1 -3.14159265359  3.14159265359 144 PERIODIC
CV-CL 2 -3.14159265359  3.14159265359 144 PERIODIC
READ_BIAS_FORCE_TRJ ../0/forces.0.dat
READ_BIAS_FORCE_TRJ ../1/forces.1.dat
```

The file can be copied locally from the provided files:

```
cp ../provided-files/input_metad_be_fcam.dat .
```

Similarly to the analysis of the 2D metadynamics mean forces and free enegy can be calculated using FCAM tools:

```
python FCAM-FOLDER/calcf_vgauss.py -if input_metad_be_fcam.dat -units kj -temp 300 > job&
```

```
python FCAM-FOLDER/graf_fes_kmc.py -ff grad_on_eff_points.out -units kj -temp 2300 -nsteps 10000000000 -ofesf fes.out > job &
```

The 2D free energy, fes.out, can be visualized using gnuplot:

```
gnuplot> spl ' fes.out' u 1:2:3 w p palette
```

Is the reconstructed free energy plot similar to the one obtained with WHAM? How does it compare with that obtained from 2D metadynamics?

Error calculations with FCAM can be obtained via block analysis through the options -trfr1 and -trfr2 of the program calcf_vgauss.py. Additional details (including analysis of bias exchange data for the same peptide) can be found in https://github.com/FCAM-NIH/FCAM.

References

Humphrey, W., Dalke, A. and Schulten, K., "VMD - Visual Molecular Dynamics", J. Molec. Graphics, 1996, vol. 14, pp. 33-38.

Reimer U, Scherer G, Drewello M, Kruber S, Schutkowski M & Fischer G (1998) J. Mol Biol 279:449-460

Vanessa Leone, Gianluca Lattanzi, Carla Molteni, Paolo Carloni. PLoS Comput Biol. (2009) Mar 13;5(3):e1000309.

Fabrizio Marinelli, Fabio Pietrucci, Alessandro Laio, Stefano Piana. (2009) PLoS Comput Biol 5(8): e1000452

G.A. Tribello, M. Bonomi, D. Branduardi, C. Camilloni, G. Bussi. (2014) Comp. Phys. Comm. 185, 604

M.J. Abraham, T. Murtola, R. Schulz, S. Páll, J.C. Smith, B. Hess, and E. Lindahl. (2015) *SoftwareX*, **1–2** 19–25

Toni Giorgino, Alessandro Laio, Alex Rodriguez (2017). Computer Physics Communications. 217:204-209

The PLUMED consortium. (2019) Nat. Methods 16, 670.

Fabrizio Marinelli, José D. Faraldo-Gómez (2021). J. Chem. Theory Comput. 2021, 17, 11, 6775–6788

Gromacs manual: http://manual.gromacs.org