Replica Exchange Molecular Dynamics (REMD) A Tutorial

Anji Babu Kapakayala
Department of Chemistry
Indian Institute of Technology Kanpur, India

March 4, 2019

This tutorial comes in three parts. In the first, the theory behind REMD simulations will be briefly described. Then we will look at how to perform t-REMD on alanine dipeptide in vacuum condition with 4 replica between 300 K and 1000 K using gromacs package patched with plumed. the AMBER14SB force field will be used and each replica will run for 2 ns. Finally, we will analyze the output files and construct free energy. This tutorial assumes you are comfortable with basic usage of Gromacs, Plumed and Linux commands.

1 Theory

Conformational sampling to simulate protein folding, drug binding processes using molecular dynamics is hampered by the slow barrier crossing conformational transitions. To overcome this, several enhanced sampling methods have been devised. Among these, global tempering approaches enhances the sampling of all the degrees of freedom of the system. Parallel tempering replica exchange molecular dynamics is a widely used global tempering method, where several copies of the system are simulated at different temperatures simultaneously and independently, while exchange of coordinates between two adjacent replicas is attempted after certain time intervals based on metropolis criteria. In this manner, high temperature replicas deliver new conformations to the lower temperature replicas which are otherwise very rarely sampled at low-temperatures.

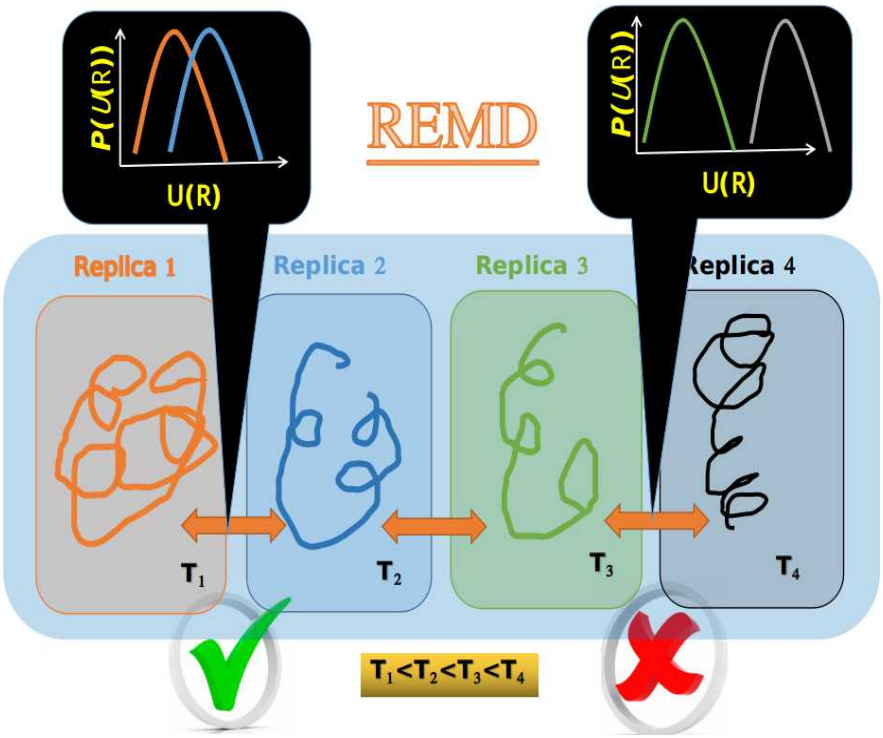
In REMD, the exchange of adjacent replica will be attempted based on metropolis criterion with following acceptance probability:

$$p(i \rightarrow j) = min1, e^{-\Delta} \tag{1}$$

Where,

$$\Delta = (\beta_i - \beta_j)(U(R_i) - U(R_j) \tag{2}$$

where $U(R_i)$ and $U(R_j)$ are the potential energies of replica i and j at temperatures β_i and β_j respectively, where $\beta = \frac{1}{k_B T}$.



Schematic of Replica Exchange Molecular Dynamics(REMD) Simulation.

Things to be noted before starting REMD Simulation:

There are number of inter-connected issues to be considered while setting up the simulation.

1. What range of temperatures do we need to span?
2. How many replicas do we need?
3. What exchange probability is needed?

Temperatures should be distributed across all the replicas in a geometric progression which means keep the exchange rate constant across the temperature range. Depending on the number of processors available and the range of

temperature to sample, choose a exponential distribution:

$$T_i = T_0 * e^{ci} \tag{3}$$

Where, c is the desire acceptance ratio and T_0 is the starting temperature, these two parameters can be tuned to obtain reasonable temperature intervals. The exponential allows the increase in temperature intervals. As distribution of total energy increases with temperature and thus exchange rate increases. In the case of larger systems you may have to use large range of temperatures in such scenario it is wise to use below link to get temperature range: Temperature Generator for REMD:(http://folding.bmc.uu.se/remd/).

The literature suggest that an exchange acceptance probability is around 0.2 (i.e 20%) is a good idea. You will have to experiment with the number of replicas you want to use to span the desired temperature range with 0.2 exchange probability.

In this tutorial we will use only 4 replicas with temperatures 300.00, 366.42, 546.63, 996.03. These temperatures gives reasonable exchange probabilities as we are dealing with alanine di-peptide in vacuum conditions. Where as you might need to use more number of replicas when you are looking for larger systems in explicit solvent.

2 Setting up REMD Simulation:

Note: To run the REMD simulations smoothly we will have to install **gromacs** with mpi version and should be patched with **plumed**(optional).

In this part, we will build the initial structures using gmx2pdb tool and perform REMD Simulation after short minimization and equilibration steps.

Preparation of Starting Structure:

The necessary steps for preparing the starting structure is dependent of whether you want to use explit solvent or vacuum or implicit solvent model. As in this tutorial we will be preparing alanine di-peptide in vacuum conditions, the preparation of initial structure are limited to generation of gromacs topology (**topol.top**) and coordinates (**conf.gro**) files.

- **Topology Generation:**

  The below command will generate the topology file default is **topol.top**, Coordinate file default is **conf.gro** and **porse.itp** file (which is used for positions restraints). During this step you will have to choose the force field as **AMBER14SB** and then choose **None.** for solvent model for

vacuum simulations.

```
gmx pdb2gmx -f ala-dipeptide.pdb -ter -ignh
```

- **Energy Minimization:**

```
; Minimization mdp For Vacuum
;            condition
integrator      = steep
emtol           = 1000.0
emstep          = 0.01
nsteps          = 50000
nstlist         = 10
cutoff-scheme   = group
ns_type         = grid
couombtype      = cutoff
rcoulomb        = 2.0
rvdw            = 2.0
pbc             = no
rlist           = 2.0
```

min.mdp:Input file for gromacs minimization in vacuum condition.

The above input(**min.mdp**) is to run 5000 steps energy minimization using steepest descent algorithm for vacuum condition. (refer gromacs manual for the more details.) Now, run these gromacs commands to perform the minimization run:

```
gmx grompp -f min.mdp -p topol.top -c conf.gro -o min.tpr
gmx mdrun -v -deffnm min
```

- **Equilibration of All the Replica:**

  In principle, we will have to equilibrate the initial structure at every temperature before starting the REMD simulation. As we are running vacuum simulation this step can be skipped.

  You can use this **nvt.mdp** file as input for the equilibration runs.

```
title          =Alanine dipeptide in vacuum
cpp                = /lib/cpp
integrator         = md
dt                 = 0.002
nsteps             = 1000000
nstxout            = 100
nstvout            = 100
nstfout            = 100
nstlog             = 100
nstenergy          = 100
nstxtcout          = 100
xtc_grps           = System
energygrps         = Protein
nstlist            = 10
ns_type            = grid
coulombtype        = cutoff
rvdw               = 2.0
rlist              = 2.0
rcoulomb           = 2.0
cutoff-scheme      =group
comm-mode          =Angular
pbc                = no
tcoupl             = v-rescale
tc-grps            = Protein
tau_t              = 0.1
ref_t              = 300
gen_vel            = yes
gen_temp           = 300
gen_seed           = 173529
constraints        = all-bonds
```

nvt.mdp:Input file for gromacs equilibration in vacuum condition.

You need to prepare 4 nvt.mdp files (nvt0.mdp, nvt1.mdp, nvt2.mdp and nvt3.mdp) for the respective temperature replica. In the above **nvt.mdp**

file modify **ref_t** tag for different temperatures. And copy the **mini.gro** from the minimization run and use this command to run Equilibration:

```
gmx grompp -f nvt.mdp -p topol.top -c mini.gro -o nvt.tpr
gmx mdrun -v -deffnm nvt
```

- **REMD Simulation:**

We will be running REMD simulation using 4 replica with temperature range 300.0, 366.42, 546.63, 996.03. We have to prepare the input files as said above.

Prepare 4 sets of inputs

[ala0.gro, nvt0.mdp] corresponds to ref_t=300.00

[ala1.gro, nvt1.mdp] corresponds to ref_t=366.42

[ala2.gro, nvt2.mdp] corresponds to ref_t=546.63

[ala3.gro, nvt3.mdp] corresponds to ref_t=996.03

For temperature-REMD we need to run a number of simulations that can communicate. This is done via **mdrun -multi** option in gromacs, and **-replex** tag also needs to be used to provide desired exchange frequency.

Construct 4 *.tpr files using above gromacs command.

we can also write a small shell script to do that using loops: **submit.sh**

```
for i in ‘0 1 2 3‘;do
gmx grompp -f nvt$i.mdp -p topol.top -c mini.gro -o remd$i.tpr
-maxwarn 10;done

mpirun -np 4 gmx_mpi mdrun -v -deffnm remd -multi 4 -replex
100
```

- **REMD with Plumed**:

In the case of using plumed you will have to give additional **-plumed** tag in the command and make sure gromacs is patched with plumed and you place plumed input file (**plumed.dat**) in the same working directory. Whereas the plumed input for alanine di-peptide phi psi as collective variables (**CVs**):

```
# set up two variables for Phi and Psi dihedral angles
phi: TORSION ATOMS=5,7,9,15
psi: TORSION ATOMS=7,9,15,17

# monitor the two variables PRINT STRIDE=10 ARG=phi,psi
FILE=COLVAR
```

And the submit script will look like: (**submit.sh**)

```
for i in '0 1 2 3';do
gmx grompp -f nvt$i.mdp -p topol.top -c mini.gro -o remd$i.tpr
-maxwarn 10;done

mpirun -np 4 gmx_mpi mdrun -v -deffnm remd -plumed
plumed.dat -multi 4 -replex 100
```

Where,
-np = No of processors used
-multi= Instruct the program to preform multi (4) runs
-replex = Instruct the system to attend an exchange at every 100 steps.

"You can find the replica exchange statistics such as exchange probabilities and the exchanges of replica involved in every 100 steps in the remd$i.log files"

3 Post Processing & Analysis:

- **Observing Replica Exchange Statistics:**

```
grep -A9 "average probabilities" *.log
```

```
remd_0.log:Repl  average probabilities:
remd_0.log-Repl     0    1    2    3
remd_0.log-Repl      .52  .19  .05
remd_0.log-Repl  number of exchanges:
remd_0.log-Repl     0    1    2    3
remd_0.log-Repl     2595  958  264
remd_0.log-Repl  average number of exchanges:
remd_0.log-Repl     0    1    2    3
remd_0.log-Repl      .52  .19  .05
remd_0.log-
--
remd_1.log:Repl  average probabilities:
remd_1.log-Repl     0    1    2    3
remd_1.log-Repl      .52  .19  .05
remd_1.log-Repl  number of exchanges:
remd_1.log-Repl     0    1    2    3
remd_1.log-Repl     2595  958  264
remd_1.log-Repl  average number of exchanges:
remd_1.log-Repl     0    1    2    3
remd_1.log-Repl      .52  .19  .05
remd_1.log-
--
remd_2.log:Repl  average probabilities:
remd_2.log-Repl     0    1    2    3
remd_2.log-Repl      .52  .19  .05
remd_2.log-Repl  number of exchanges:
remd_2.log-Repl     0    1    2    3
remd_2.log-Repl     2595  958  264
remd_2.log-Repl  average number of exchanges:
remd_2.log-Repl     0    1    2    3
remd_2.log-Repl      .52  .19  .05
remd_2.log-
--
remd_3.log:Repl  average probabilities:
remd_3.log-Repl     0    1    2    3
remd_3.log-Repl      .52  .19  .05
remd_3.log-Repl  number of exchanges:
remd_3.log-Repl     0    1    2    3
remd_3.log-Repl     2595  958  264
remd_3.log-Repl  average number of exchanges:
remd_3.log-Repl     0    1    2    3
remd_3.log-Repl      .52  .19  .05
remd_3.log-
```

Log::Replica exchange statistics extracted from log files.

- **Concatenate the trajectories:**

  By concatenating all the log files into single log file (**REMD.log**) and by using gromacs built-in tool **demux.pl** will generate the **replica_index.xvg** and **replica_temp.xvg** files. We need a trajectory with continuous coordinates despite the jumps in the ensemble space due to attempted exchanges. This trajectory can be generated using **gmx trjcat** tool with **-demux** tag and above index files.

  You can use the below commands to do this:

```
# Concatenate log files
cat *.log > REMD.log
demux.pl REMD.log

# De-multiplexing a REMD trajectory
gmx trjcat -f *.xtc -demux replica_index.xvg
```

- **Checking Potential Energy Overlap**

  As we have discussed above (referring Schematic of REMD) the metropolis criteria is dependent of the potential energy deference of the replicas ($\Delta U(\mathbf{R})$). Which means that to make sure the adjacent replicas to be sampled the continuous coordinates their potential energy distributions should overlap else the metropolis criteria will reject the exchange.

  And also make a note that checking distributions of potential energy overlap is the wise idea to check whether the taken temperature range is required enough or not. See the PE overlap of 4 replica:

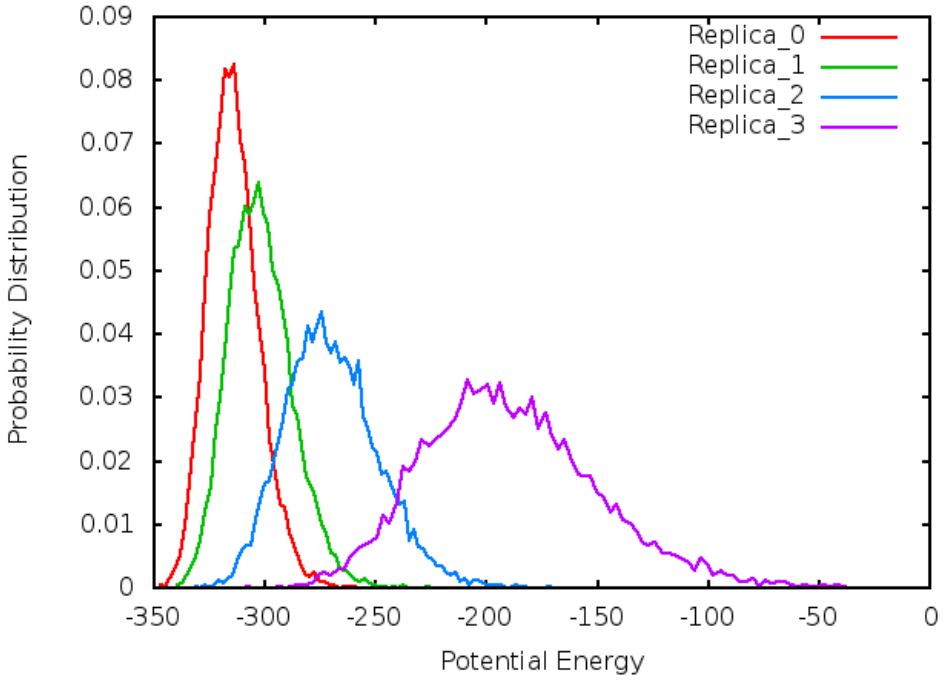


Fig:4:Overlap of distributions of potential energy of each replica.

You can get the potential energy of each replica with time using **gmx energy** tool and then you can use **xmgrace** or self written program to get the distribution of it.

Gromacs Commands:

```
# Choose potential energy on screen to print data into file PE.xvg

gmx energy -f remd.edr -s remd.tpr -o PE.xvg
```

- **Construction of FES:**

  In the case of using **PLUMED**, you can see the COLVAR$i files for thr respective temperature replica. you can plot the CV values with time to check their evaluation. **Gnuplot** commands to plot CV v/s Time of 300 K replica are as follows:

```
# Open Gnuplot on terminal
Terminal$ gnuplot


        plot "COLVAR0" u 1:2 w l lw 2 title"PHI"
        plot "COLVAR0" u 1:3 w l lw 2 title"PSI"
```

  Whereas, the 1D free energy surface along Phi and Psi can be constructed using plumed in-built tool **sum_hills** with following command:

```
# Construct FES using sum_hills


plumed sum_hills –histo COLVAR0 –idw phi –sigma 0.2 –kt 2.5
                –outhisto fes_phi.dat
plumed sum_hills –histo COLVAR0 –idw psi –sigma 0.2 –kt 2.5
                –outhisto fes_psi.dat
```

These command will produce the following figures:

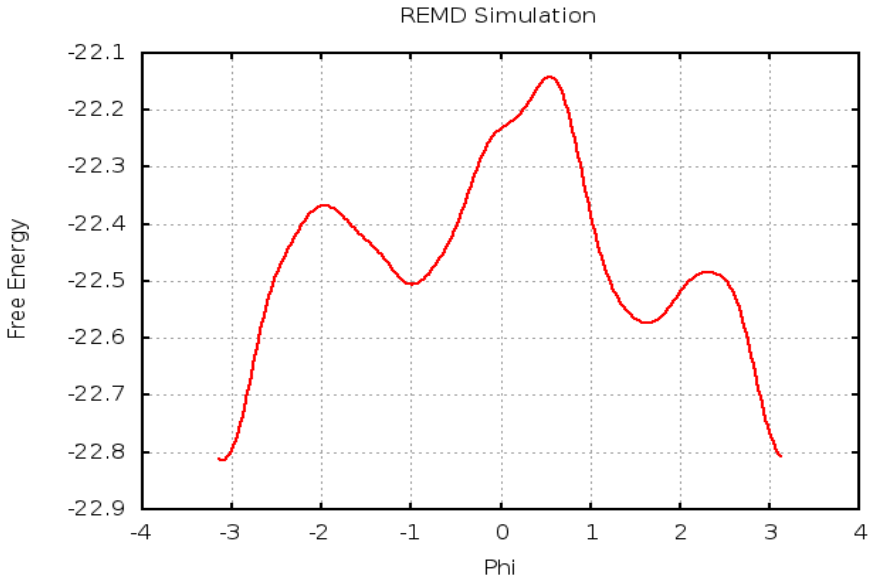


Fig:5:Free energy along Phi.

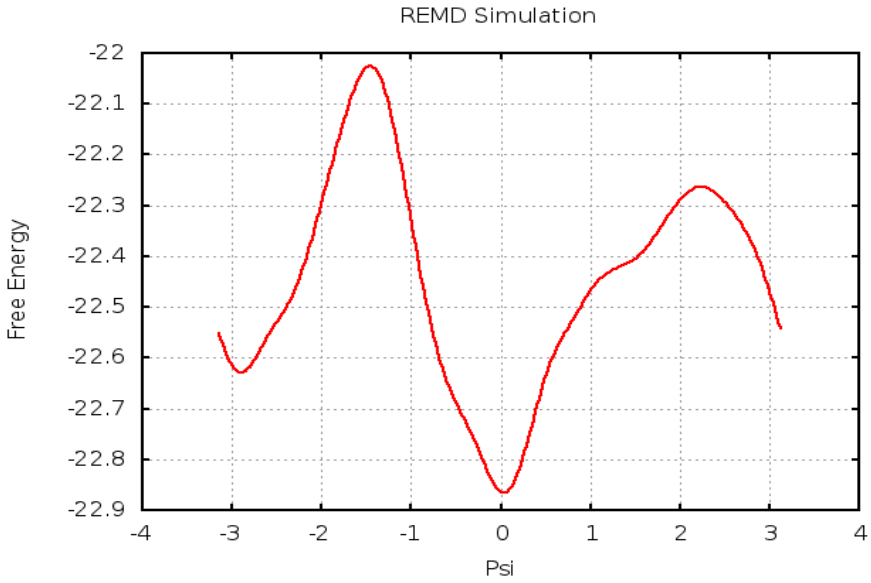


Fig:6:Free energy along Psi.

4 Summary:

To be conclude we have discussed the underlying theory of REMD and setting up the simulations along with using the post processing commands and tools. Have a look at our GitHub page to know more details about scripts and input files used in this tutorial(Nisanth Nair GitHub Organization).

5 References:

1. Sugita, Y.; Okamoto, Y. Chem. Phys. Lett. 1999, 314 (1–2), 141–151.
2. Plumed: [http://www.plumed.org/]
3. More about sum_hills: Click here
4. More Details: **REMD Tutorial on Github**.