MM/GBSA tutorials for SARS-CoV-2 Mpro in complex with inhibitor N3 by MolAICal and NAMD

Qifeng Bai
Email: molaical@yeah.net
Homepage: https://molaical.github.io
School of Basic Medical Sciences
Lanzhou University
Lanzhou, Gansu 730000, P. R. China

Update on 2021-11-22

1. Introduction

In this tutorial, the MolAICal (https://molaical.github.io) is used to calculate the MM/GBSA between ligand N3 and SARS-CoV-2 Mpro based on molecular dynamical (MD) simulated results by NAMD. This tutorial is just a demo. To save running and storage space, only 25 frames of MD simulated trajectories of SARS-CoV-2 Mpro in complex with N3 are selected for this tutorial. This tutorial can **not only** be used to calculate MM/GBSA of protein-ligand **but also** MM/GBSA of protein-peptide, protein-protein, DNA-ligand, DNA-protein, protein-DNA, protein-RNA, and any complex based on the run MD simulations. It just replaces the protein and ligand of this tutorial with the appointed objects. For example, **protein** in this tutorial is substituted for **DNA**, and **ligand** in this tutorial is substituted for **peptide** **based on the same running command arguments of this tutorial**.

2. Materials

2.1. Software requirement

1) MolAICal : https://molaical.github.io
2) NAMD: https://www.ks.uiuc.edu/Research/namd/
(**Notice:** This tutorial can be run by **NAMD 2.x, 3.x, or a higher version.** For example, the command "**namd3**" substitutes for the command "**namd2**" in this tutorial **if you use NAMD 3.x version**. For a higher version of NAMD, users can use a similar replacement way as the previous example.)
3) Carma: https://github.com/glykos/carma or https://utopia.duth.gr/~glykos/Carma.html

2.2. Example files

1) All the necessary tutorial files are downloaded from:
https://github.com/MolAICal/tutorials/tree/master/004-MMGBSA

3. Procedure

Part I. Problem solution

Problem: some users meet the positive value problem when calculating MM/GBSA:
https://www.ks.uiuc.edu/Research/namd/mailing_list/namd-l.2020-2021/1295.html

Possible solutions: it must check whether the ligand is always in the same periodic boundary with the receptor along the entire simulated time. If not, please use the VMD PBC to wrap the ligand into the receptor along simulated time. More detailed:
https://www.ks.uiuc.edu/Research/vmd/script_library/scripts/pbcwrap/
https://www.ks.uiuc.edu/Research/vmd/plugins/pbctools/

I supply two commands run on the VMD Tk Console:

```
%> pbc wrap -centersel protein -center com -compound chain -all
%> pbc unwrap -sel "not (water or ions)" -all
```

Running above one or two commands can wrap the ligand into the receptor along the simulated time. Saving wrap or unwrap trajectories into DCD file using the below command:

```
%> set all [atomselect top all]
%> animate write dcd pro.dcd sel $all beg 0 end 24 waitfor all
```

Here, all atoms are selected. Users can select the wanted part. Our tutorial trajectories only have 25 frames. So the beginning is 0 and the end is 24. "pro.dcd" is just a name. And then starting the follow steps.

Part II. MM/GBSA calculation based on NAMD

3.1. MM/GBSA calculation

Firstly, I assume the equilibrium trajectory named "mpro.dcd" has been run by NAMD. Receptor and ligand have been put into the same periodic boundary in "mpro.dcd". Users can replace "mpro.dcd" with their own trajectory. Go to the tutorial directory:

```
#> cd 004-MMGBSA
```

3.1.1. Extracting trajectory of protein in complex with ligand

```
#> vmd -dispdev text -psf "mpro.psf" -e stripDCD.vmd -args protein,or,resname,LIG "mpro.dcd" "complex" mpro.psf mpro.pdb
```

-args: it is the usage like the command "atomselect" in Tk Console of VMD software such as "atomselect top protein or resname LIG". Here, comma "," represents blank space " ". The script file "stripDCD.vmd" can be found in the directory "scripts" of MolAICal software.

It will generate complex.psf, complex.pdb and complex.dcd. Turning on the parameters of "GBIS" and "sasa". Open "complex.conf" and modify the appropriate parameters of red fonts as below:

```
----------------------------------------------------------------------------------------------------------------
structure               complex.psf
coordinates             complex.pdb
outputName                complex

paraTypeCharmm            on
parameters                par_all36_prot.prm
parameters                par_all36_cgenff.prm
parameters                ligand.str
parameters                toppar_water_ions.str

coorfile open dcd complex.dcd
----------------------------------------------------------------------
```

Our tutorial is run by CPU. You can run it on GPU. Running NAMD command in Linux operating system as below:

```
#> namd2 +p3 complex.conf >& complex.log &
```

Where the symbol "&" assigns the command to run in the background on the Linux operating system. If NAMD runs on Windows operating system, the symbol "&" must be omitted. For instance:

```
#> namd2 +p3 complex.conf > complex.log
```

3.1.2. Extracting trajectory of protein only.

```
#> vmd -dispdev text -psf "mpro.psf" -e stripDCD.vmd -args protein "mpro.dcd" "protein" mpro.psf mpro.pdb
```

It will generate protein.psf, protein.pdb and protein.dcd. Open "protein.conf" and modify the appropriate parameters in the similar way of "complex.conf"

Our tutorial is run by CPU. You can run it on GPU. Running NAMD command in Linux operating system as below:

```
#> namd2 +p3 protein.conf >& protein.log &
```

3.1.3. Extracting trajectory of ligand only.

```
#> vmd -dispdev text -psf "mpro.psf" -e stripDCD.vmd -args resname,LIG "mpro.dcd" "ligand" mpro.psf mpro.pdb
```

It will generate ligand.psf, ligand.pdb and ligand.dcd. Open "ligand.conf" and modify the appropriate parameters in the similar way of "complex.conf"

Our tutorial is run by CPU. You can run it on GPU. Running NAMD command in Linux operating system as below:

```
#> namd2 +p3 ligand.conf >& ligand.log &
```

3.1.4. Calculating MM/GBSA by MolAICal

```
#> molaical.exe -mmgbsa -c complex.log -r protein.log -l ligand.log
```

The output contains the binding free energy △G as below:

```
------------------------------------------------------------------------
delta E(internal): -0
delta E(electrostatic) + deltaG(sol): 7.7029
delta E(VDW): -44.4361
delta G binding: -36.7332    +/-    3.4202 (kcal/mol)
------------------------------------------------------------------------
```

Part III. Decomposing the free energy contributions of a per-residue

3.2. Decomposing the free energy

If the users want to use MolAICal to decompose the free energy contributions of a per-residue, it can draw lessons from the above steps. For example, the residue M165 of SARS-CoV-2 Mpro is reported to interact with ligand N3. So the residue M165 is chosen as an example. First of all, change the console into the “004-MMGBSA\Decompose” folder:

```
#> cd 004-MMGBSA\Decompose
```

3.2.1. Extracting trajectory of an appointed residue in complex with ligand

```
#> vmd -dispdev text -psf "../mpro.psf" -e ../stripDCD.vmd -args protein,and,resid,165,or,resname,LIG "../mpro.dcd" "res-lig" ../mpro.psf ../mpro.pdb
```

-args: it is the usage like the command “atomselect” of VMD software such as "atomselect top protein and resid 165 or resname LIG". Here, comma "," represents blank space " ". The script file “stripDCD.vmd” can be found in the directory “scripts” of MolAICal software.

It will generate res-lig.psf, res-lig.pdb and res-lig.dcd. Open “res-lig.conf” and modify the appropriate parameters in the similar way of “complex.conf”

Our tutorial is run by CPU. You can run it on GPU. Running NAMD command in Linux operating system as below:

```
#> namd2 +p3 res-lig.conf >& res-lig.log &
```

3.2.2. Extracting trajectory of an appointed residue only

```
#> vmd -dispdev text -psf "../mpro.psf" -e ../stripDCD.vmd -args protein,and,resid,165 "../mpro.dcd" "res" ../mpro.psf ../mpro.pdb
```

It will generate res.psf, res.pdb and res.dcd. Open “res.conf” and modify the appropriate parameters in the similar way of “complex.conf”

Our tutorial is run by CPU. You can run it on GPU. Running NAMD command in Linux operating system as below:

```
#> namd2 +p3 res.conf >& res.log &
```

3.2.3. Extracting trajectory of ligand only

```
#> vmd -dispdev text -psf "../mpro.psf" -e ../stripDCD.vmd -args resname,LIG "../mpro.dcd" "lig" ../mpro.psf ../mpro.pdb
```

It will generate lig.psf, lig.pdb and lig.dcd. Open “lig.conf” and modify the appropriate parameters in the similar way of “complex.conf”

Our tutorial is run by CPU. You can run it on GPU. Running NAMD command in Linux operating system as below:

```
#> namd2 +p3 lig.conf >& lig.log &
```

3.2.4. Calculating free energy contributions of a per-residue by MolAICal

```
#> molaical.exe -mmgbsa -c res-lig.log -r res.log -l lig.log
```

The output contains the binding free energy of a per-residue △G as below:

```
------------------------------------------------------------------------
delta E(internal): 0
delta E(electrostatic) + deltaG(sol): -0.4691
delta E(VDW): -4.4319
delta G binding: -4.901    +/-    1.0524 (kcal/mol)
------------------------------------------------------------------------
```

Note: the users can use the above similar method to calculate the free energy contribution of pairwise and per-residue by MolAICal.

Part IV. Entropy calculation

3.3. Entropy calculation by Carma and MolAICal

3.3.1. Download the Carma

Carma can run on Windows or Linux operating systems. Download new version Carma V2.01 from https://github.com/glykos/carma or http://utopia.duth.gr/~glykos/progs

I recommend the new latest version of Carma for entropy calculation. The previous version will show the "NaN" errors, etc. Please check the below issue report:
https://groups.google.com/g/carma-molecular-dynamics/c/KpyY5sEkrj4
Here, 100 frames are extracted from DCD trajectories for entropy calculation. The users can extract a suitable number of frames for their projects. More number of frames will cost more running time.

3.3.2. Calculate the complex entropy

```
#> cd 004-MMGBSA\entropy
```

Goto folder "com". This step is for removing rotations and translations.

```
#> carma -v -fit -force -atmid ALLID -segid A -segid C complex.dcd complex.psf
```

This step is for calculating entropy.

```
#> carma -v -cov -eigen -mass -force -temp 310 -atmid ALLID -segid A -segid C -last 100 carma.fitted.dcd complex.psf >& com.log &
```

3.3.3. Calculate the receptor entropy

Goto folder “rec”. This step is for removing rotations and translations.

```
#> carma -force -v -fit -atmid ALLID -segid A protein.dcd protein.psf
```

This step is for calculating entropy.

```
#> carma -v -cov -eigen -mass -force -temp 310 -atmid ALLID -segid A -last 100 carma.fitted.dcd protein.psf >& rec.log &
```

3.3.4. Calculate the ligand entropy

Goto folder “lig”. This step is for removing rotations and translations.

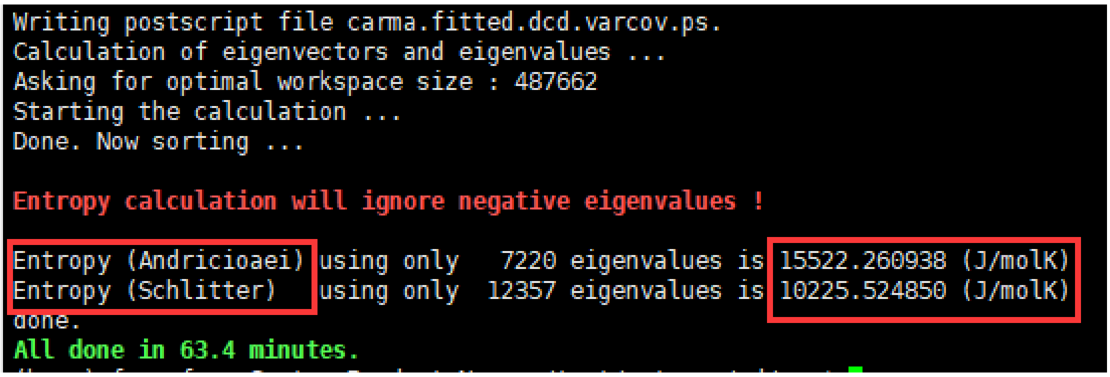
```
#> carma -v -fit -force -atmid ALLID -segid C ligand.dcd ligand.psf
```

This step is for calculating entropy.

```
#> carma -v -cov -eigen -mass -force -temp 310 -atmid ALLID -segid C -last 100 carma.fitted.dcd ligand.psf >& lig.log &
```

3.3.5. Calculate the entropy

Please check log files: “com.log”, “rec.log” and “lig.log”, and find the Andricioaei’s or Schlitter’s entropy values in the files of “com.log”, “rec.log” and “lig.log”. For example, it can find the Andricioaei’s or Schlitter’s entropy value in the content of “com.log” (see below figure):



```
Writing postscript file carma.fitted.dcd.varcov.ps.
Calculation of eigenvectors and eigenvalues ...
Asking for optimal workspace size : 487662
Starting the calculation ...
Done. Now sorting ...

Entropy calculation will ignore negative eigenvalues !

Entropy (Andricioaei) using only   7220 eigenvalues is 15522.260938 (J/molK)
Entropy (Schlitter)   using only  12357 eigenvalues is 10225.524850 (J/molK)
done.
All done in 63.4 minutes.
```

For Andricioaei entropy, run the below command:

```
#> molaical.exe -entropy -c 15522.260938 -r 15211.880284 -l 1454.409253 -t 310.0
```

It will show the result:

```
The entropy DELTA S = -84.70646297459386 (kcal/mol)
```

For Schlitter entropy, run the below command:

```
#> molaical.exe -entropy -c 10225.524850 -r 10028.714754 -l 1465.682932 -t 310.0
```

It will show the result:

```
The entropy DELTA S = -93.9502124300494 (kcal/mol)
```

Where: **-entropy:** means delta entropy calculation

-c: the entropy of complex

-r: the entropy of the receptor or first molecule

-l: the entropy of the ligand or second molecule

-t: the Kelvin temperature in MD simulation