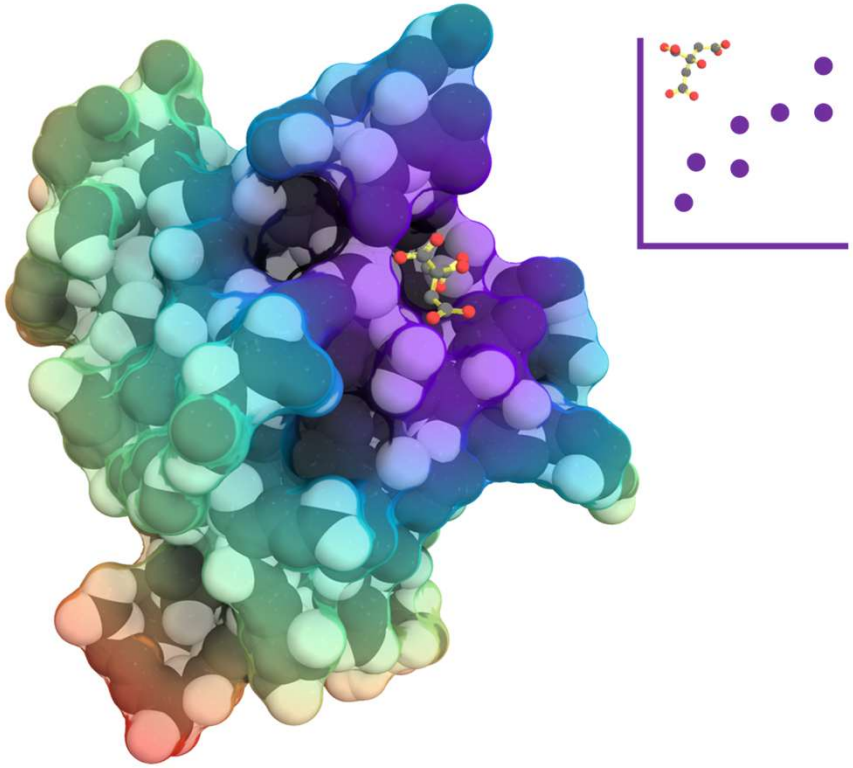
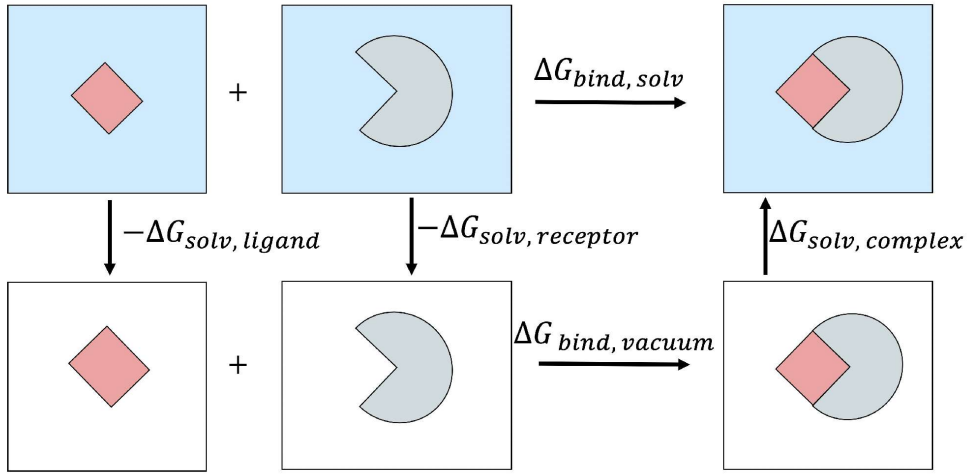
MM-GBSA with Gromacs

Naf Guo
2025

MM-GBSA

MM-GBSA/PBSA是一类构建热力学循环，计算配体与蛋白亲和力的方法，精度优于分子对接，劣于自由能微扰





Forouzesh, N.; Mishra, N. Molecules 2021, 26, 2383.

https://valdes-tresanco-ms.github.io/gmx_MMPBSA/dev/

Install UniGBSA

https://github.com/dptech-corp/Uni-GBSA

https://github.com/dptech-corp/Uni-GBSA

我们首先用conda安装一个包unigbsa

```
conda create -n gbsa -c conda-forge acpype openmpi mpi4py "gmx_mmpbsa>=1.5.6"
```

```
conda activate gbsa
```

```
pip install unigbsa lickit
```

Uni-GBSA: An Automatic Workflow to Perform MM/GB(PB)SA Calculations for Virtual Screening

[Briefings in Bioinformatics]

Background

Calculating the binding free energy of a ligand to a protein receptor is a crucial goal in drug discovery. Molecular mechanics/Generalized-Born (Poisson-Boltzmann) surface area (MM/GB(PB)SA), which balances accuracy and efficiency, is one of the most widely used methods for evaluating ligand binding free energies in virtual screening. Uni-GBSA is an automatic workflow to perform MM/GB(PB)SA calculations. It includes several functions, including but not limited to topology preparation, structure optimization, binding free energy calculation, and parameter scanning for MM/GB(PB)SA calculations. Additionally, it has a batch mode that allows the evaluation of thousands of molecules against one protein target simultaneously, enabling its application in virtual screening.

Maohua Yang, Zonghua Bo, Tao Xu, Bo Xu, Dongdong Wang, Hang Zheng, Uni-GBSA: an open-source and web-based automatic workflow to perform MM/GB(PB)SA calculations for virtual screening, Briefings in Bioinformatics, Volume 24, Issue 4, July 2023, bbad218,

Perform Calculation with one snapshot

https://github.com/dptech-corp/Uni-GBSA

我们就使用之前跑蛋白配体复合物的体系，注意，这里使用的蛋白质和配体都是做完了准备的

首先激活安装了UniGBSA的conda环境

```
conda activate gbsa
```

对做完了准备的蛋白质和配体结构，计算结合自由能，并进行能量分解

```
unigbsa-pipeline -i protein.pdb -l LAB.mol2 -o BindingEnergy.csv --decomp
```

计算以后对我们输入的这一帧，ΔG为 -52.8839 kcal/mol，一定要注意，这些计算的绝对值意义很小，我们一般的用法是，对一个蛋白质的一系列配体，做完对接以后，对得到的结合构象，均使用GBSA计算自由能，并排序
GBSA的结果越负越好，但是也要注意，**GBSA与实验值的相关性也比较一般，不能作为金标准来看，比对接打分好，比FEP烂**

```
(gbsa) root@bohrium-11312-1251688:~/gbsa# unigbsa-pipeline -i protein.pdb -l LAB.mol2 -o BindingEnergy.csv --decomp
03/01/2025 22:34:28 PM - INFO - Build protein topology.
03/01/2025 22:34:37 PM - INFO - Build ligand topology: LAB.
03/01/2025 22:34:48 PM - INFO - Running energy minimization: LAB.
03/01/2025 22:35:02 PM - INFO - Run the MMPB(GB)SA.
03/01/2025 22:36:55 PM - INFO - Clean the results.
================================================================================
Results: Energy.csv Dec.csv
Frames    mode    delta_G(kcal/mole)
     1      gb              -52.8839
```

Energy Decomposition

https://github.com/dptech-corp/Uni-GBSA

刚才进行计算的时候，我们打开了decomp选项，对结果进行了能量分解

```
unigbsa-pipeline -i protein.pdb -l LAB.mol2 -o BindingEnergy.csv --decomp
```

计算以后会出现一个以配体名字命名的文件夹，里面有一个**Dec.csv**文件，记录了蛋白质的不同残基对配体结合的贡献，这就是为什么我们称之为能量分解，把结合自由能分拆到每个氨基酸残基上

```
(gbsa) root@bohrium-11312-1251688:~/gbsa# ls
BindingEnergy.csv  LAB.  LAB.mol2  protein.pdb
(gbsa) root@bohrium-11312-1251688:~/gbsa# vim BindingEnergy.csv
(gbsa) root@bohrium-11312-1251688:~/gbsa# cd LAB.
(gbsa) root@bohrium-11312-1251688:~/gbsa/LAB.# ls
Dec.csv  Energy.csv  LAB..mol  complex.pdb  complex.top  complex_reres.pdb  index.ndx
```

	Frame	resid	mode	Inte	Van der	Electr	Polar Solv	Non-Pola	TOTAL
0	1	R:B:LEU:16	gb	0	-2.174	-0	0.002	-0.28452	-2.49552
1	1	R:B:GLN:59	gb	0	-0.949	0.13	0.073	-0.18217	-0.92417
2	1	R:B:ARG:183	gb	0	-2.631	0.6	-0.535	-0.05989	-2.62389
3	1	R:B:GLY:15	gb	0	-0.7	-0.1	0.312	-0.05875	-0.58375
4	1	R:B:TYR:69	gb	0	-1.666	-0.6	0.316	-0.18677	-2.12377
5	1	L::MOL:1	gb	0	-23.25	-6.7	4.792	-3.76031	-28.9223
6	1	R:B:PRO:32	gb	0	-1.592	0.07	-0.056	-0.20972	-1.78672
7	1	R:B:ARG:210	gb	0	-3.944	-1.1	0.427	-0.43224	-5.00224
8	1	R:B:LYS:213	gb	0	-0.393	-2.6	2.166	0	-0.83
9	1	R:B:ILE:34	gb	0	-1.311	0.01	-0.001	-0.09199	-1.38999
10	1	R:B:ASP:157	gb	0	-0.729	-1.1	1.299	-0.09003	-0.62803
11	1	R:B:ARG:206	gb	0	-1.175	0.65	-0.456	-0.02915	-1.01215

除了total energy也就是总能量以外，对每个残基还给出了不同类型能量的数值，我们可以看出每个残基与配体结合的主要驱动力来源，当然，也就是做个参考

Energy Decomposition Plotting

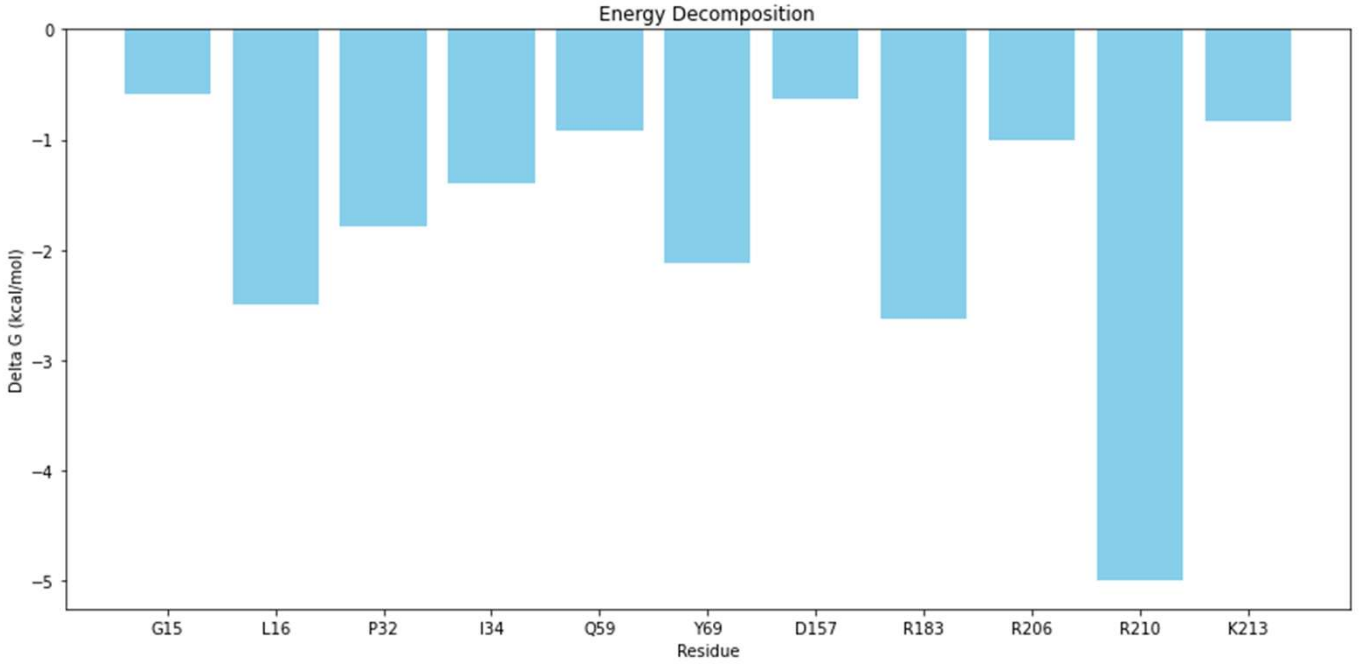
https://github.com/dptech-corp/Uni-GBSA

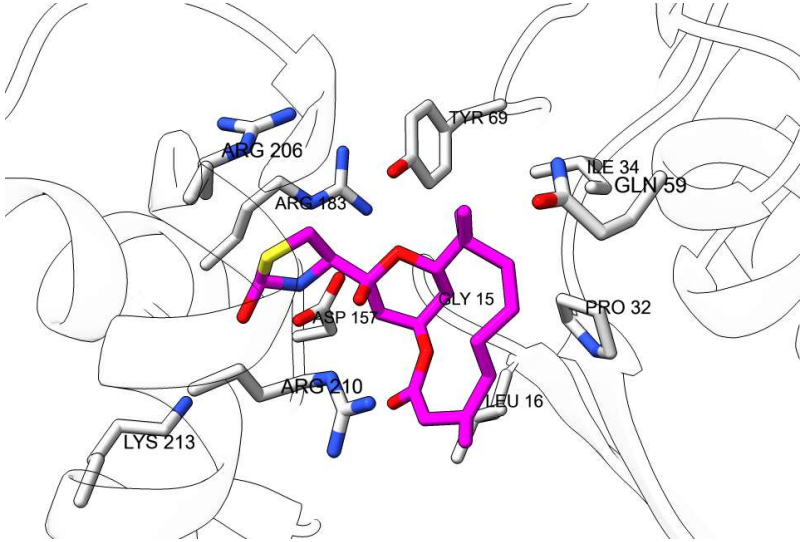
对Dec.csv的内容，我们可以用excel或者python脚本作图作图

conda install pandas matplotlib

我提供了一个python脚本，以jupyter notebook的形式给出

https://github.com/Sept-naf/gromacs-tutorials/blob/main/plot/plot_Dec.ipynb





Perform Calculation with trajectories

https://github.com/dptech-corp/Uni-GBSA

假如你已经跑了一段轨迹，想对轨迹进行GBSA计算，可以想象，最简单的做法就是对轨迹里的每一帧均进行GBSA计算，取平均

```
conda activate gbsa
```

对跑完了的轨迹，我们需要一个tpr文件，所有的拓扑文件，轨迹文件

```
(gbsa) root@bohrium-11312-1251688:~/gbsa/analysis_traj# ls
MOL_GMX.itp  md.xtc         posre_Protein_chain_B.itp  topol.top                  topol_Protein_chain_M.itp
md.tpr       posre_MOL.itp  posre_Protein_chain_M.itp  topol_Protein_chain_B.itp
```

我们还需要准备一个新的index.ndx文件，创建新组，把蛋白命名为receptor，把配体命名为ligand

```
gmx make_ndx -f md.tpr
```

```
 0 System              : 104760 atoms
 1 Protein             :  6443 atoms
 2 Protein-H           :  3224 atoms
 3 C-alpha             :   408 atoms
 4 Backbone            :  1224 atoms
 5 MainChain           :  1634 atoms
 6 MainChain+Cb        :  2014 atoms
 7 MainChain+H         :  2026 atoms
 8 SideChain           :  4417 atoms
 9 SideChain-H         :  1590 atoms
10 Prot-Masses         :  6443 atoms
11 non-Protein         : 98317 atoms
12 Other               :    56 atoms
13 LAB                 :    56 atoms
14 NA                  :   103 atoms
15 CL                  :    97 atoms
16 Water               : 98061 atoms
17 SOL                 : 98061 atoms
18 non-Water           :  6699 atoms
19 Ion                 :   200 atoms
20 Water_and_ions      : 98261 atoms
```

```
> 1

Copied index group 1 'Protein'

> name 21 receptor


> 13

Copied index group 13 'LAB'

> name 22 ligand


> q
```



```
 0 System              : 104760 atoms
 1 Protein             :  6443 atoms
 2 Protein-H           :  3224 atoms
 3 C-alpha             :   408 atoms
 4 Backbone            :  1224 atoms
 5 MainChain           :  1634 atoms
 6 MainChain+Cb        :  2014 atoms
 7 MainChain+H         :  2026 atoms
 8 SideChain           :  4417 atoms
 9 SideChain-H         :  1590 atoms
10 Prot-Masses         :  6443 atoms
11 non-Protein         : 98317 atoms
12 Other               :    56 atoms
13 LAB                 :    56 atoms
14 NA                  :   103 atoms
15 CL                  :    97 atoms
16 Water               : 98061 atoms
17 SOL                 : 98061 atoms
18 non-Water           :  6699 atoms
19 Ion                 :   200 atoms
20 Water_and_ions      : 98261 atoms
21 receptor            :  6443 atoms
22 ligand              :    56 atoms
```

Perform Calculation with trajectories

https://github.com/dptech-corp/Uni-GBSA

文件准备好了以后就可以跑了

```
unigbsa-traj -i md.tpr -p topol.top -ndx index.ndx -m gb,decomposition -t md.xtc
```

```
unigbsa-traj -i md.tpr -p topol.top -ndx index.ndx -m gb -t md.xtc
```

```
(gbsa) root@bohrium-11312-1251688:~/gbsa/analysis_traj# unigbsa-traj -i md.tpr -p topol.top -ndx index.ndx -m gb -t md.xtc
03/01/2025 23:56:03 PM - INFO - Run the MMPB(GB)SA.
03/01/2025 23:59:04 PM - INFO - Clean the results.
================================================================================
Results: Energy.csv Dec.csv
    Frames mode      complex     receptor     ligand      Internal  ...  Electrostatic  Polar Solvation  Non-Polar Solvation      Gas  Solvation      TOTAL
0        1   gb  2582.075442  2580.369922  52.512093 -1.634248e-13  ...       -10.0084          12.4858            -6.229572 -57.0628   6.256228 -50.806572
1        2   gb  2712.987647  2712.008636  50.140823  1.000000e-04  ...        -7.0035           9.3241            -6.122012 -52.3639   3.202088 -49.161812
2        3   gb  2706.258466  2707.409531  50.149687  1.207923e-13  ...        -7.3153          10.1109            -6.203052 -55.2087   3.907848 -51.300852
3        4   gb  2784.444030  2782.151928  50.456246 -1.000000e-04  ...        -7.4844          11.3542            -6.216144 -53.3022   5.138056 -48.164144
4        5   gb  2597.210037  2596.948776  48.745943  1.000000e-04  ...        -3.6858           8.5696            -5.913382 -51.1411   2.656218 -48.484882
5        6   gb  2674.617863  2669.323462  53.811232  1.000000e-04  ...        -8.6489          10.6960            -6.156131 -53.0567   4.539869 -48.516831
6        7   gb  2751.974021  2750.457545  51.334096  5.329071e-13  ...        -6.3045          10.9537            -6.059920 -54.7114   4.893780 -49.817620
7        8   gb  2660.007325  2653.522539  52.030299 -1.000000e-04  ...        -6.1770           9.9305            -5.874914 -49.6010   4.055586 -45.545414
8        9   gb  2469.123716  2470.360786  49.800820 -3.712586e-13  ...        -7.4023          10.3330            -5.908491 -55.4623   4.424509 -51.037791
9       10   gb  2678.615550  2666.953821  57.454491  1.000000e-04  ...        -8.7200          11.5002            -5.964362 -51.3288   5.535838 -45.792962
10      11   gb  2619.155628  2610.461458  55.383548 -1.000000e-04  ...        -8.5905          10.6867            -5.665878 -51.7101   5.020822 -46.689278
11      12   gb  2673.478512  2664.333696  51.718394  1.000000e-04  ...        -7.6121          10.5717            -5.523178 -47.6222   5.048522 -42.573678
12      13   gb  2795.942422  2781.074428  56.240071 -1.065814e-13  ...        -9.4676          11.9734            -6.053577 -47.2918   5.919823 -41.371977
13      14   gb  2694.388010  2684.707829  51.407442 -1.000000e-04  ...        -8.7906          12.2080            -5.719961 -48.2152   6.488039 -41.727161
14      15   gb  2673.771741  2672.078080  48.632720  3.552714e-13  ...        -5.6486           9.7977            -5.765859 -50.9709   4.031841 -46.939059
15      16   gb  2646.099636  2639.649932  51.145409  8.970602e-14  ...        -8.3269          11.0086            -5.879305 -49.8251   5.129295 -44.695805
16      17   gb  2605.929353  2599.669812  53.550965  1.000000e-04  ...        -9.7864          12.0418            -6.152624 -53.1806   5.889176 -47.291424
17      18   gb  2584.746257  2579.222758  51.804379 -1.634248e-13  ...        -8.6581          10.8557            -6.032780 -51.1038   4.822920 -46.280880
18      19   gb  2726.616612  2721.438047  52.121065 -8.313350e-13  ...        -7.8175          10.0621            -5.611100 -51.3935   4.451000 -46.942500
19      20   gb  2603.473080  2588.719338  59.192816  1.000000e-04  ...        -7.8290          10.5817            -5.984874 -49.0359   4.596826 -44.439074
20      21   gb  2556.767564  2550.167258  51.709027 -6.616929e-13  ...        -7.9406          10.5247            -5.732321 -49.9011   4.792379 -45.108721
```

Perform MD Simulation with UniGBSA

https://github.com/dptech-corp/Uni-GBSA

Uni-GBSA也可以用来直接跑MD，但是，出于学习的考虑，希望初学的时候少用这个功能

```
unigbsa-md -p protein.pdb -l LAB.mol2 -pf amber99sb -lf gaff2 -d 1.2 -conc 0.15 -o output -nsteps 1000
```

```
$ unigbsa-md -h
usage: unigbsa-md [-h] -p PROTEIN [-l LIGAND] [-pf PROTFORCE] [-lf {gaff,gaff2}] [-bt BOXTYPE] [-box BOX BOX BOX] [-d D] [-conc CONC] [-o OUTDIR]
[-nsteps NSTEP] [-nframe NFRAME]
              [-nt THREADS] [-verbose] [-v]

Run MD simulation for input file.

optional arguments:
  -h, --help       show this help message and exit
  -p PROTEIN       Protein file for the simulation.
  -l LIGAND        Ligand file or directory for the simulation.
  -pf PROTFORCE    Protein forcefield.
  -lf {gaff,gaff2}  Ligand forcefield: gaff or gaff2.
  -bt BOXTYPE      Simulation box type, default: triclinic
  -box BOX BOX BOX  Simulation box size.
  -d D             Distance between the solute and the box.
  -conc CONC       Specify salt concentration (mol/liter). default=0.15
  -o OUTDIR        The output directory.
  -nsteps NSTEP    Simulation steps. default:2500
  -nframe NFRAME   Number of frames to save for the xtc file. default:100
  -nt THREADS      Number of threads to run this simulation.
  -verbose         Keep all the files in the simulation.
  -v, --version    show program's version number and exit
```