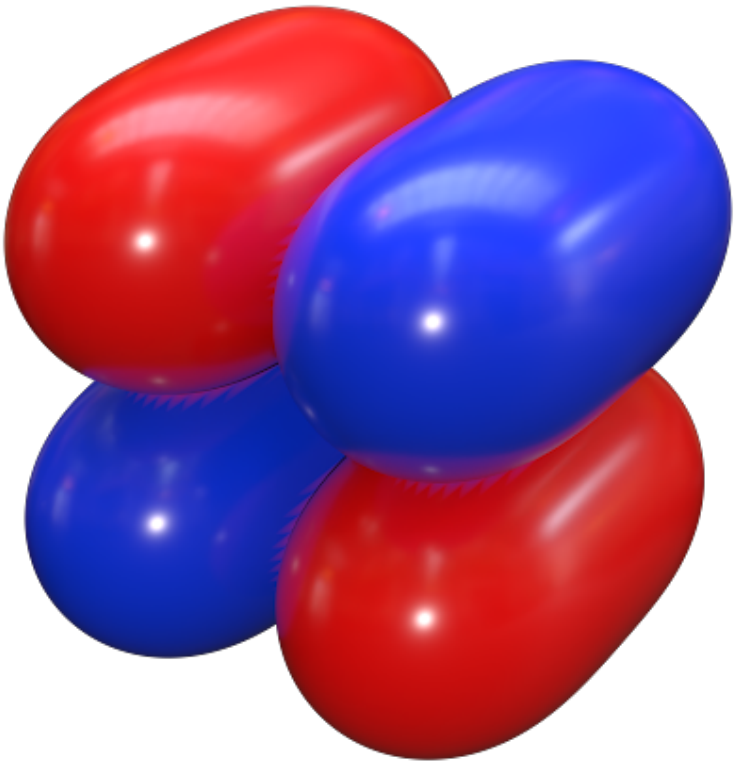
Sakura Science Molecular Simulation Workshop Series: Redocking at DFTB Level

February 23, 2025



1 Introduction

Molecular docking is generally known as a tool to estimate the binding energies and/or binding free energies between two molecules with specific active sites. This method is well-established for many types of ligand-receptor interactions covering a broad range of material designs, including drug discovery. Popular free molecular docking tools are:

- **Autodock Vina and AutoDock** Launched by the Scripps research institute, AutoDock is well-renowned for its power and flexibility. It is often praised for its impressive docking calculations and virtual screening. It is free to use, wisdom of crowds approach for enhanced accuracy, and extensive online user community for peer-based support. Both AutoDock and Autodck Vina can be found at `https://autodock.scripps.edu`.
- **Unidock** A powerful enhanced version of docking tools that utilize machine learning to reproduce docking score functions, including but not limited to the Vina score. The virtual screening performed by using this software is extremely quick since it utilizes the GPU. You can freely download the software at `https://github.com/dptech-corp/Uni-Dock`.
- **DOCK6** Popular docking software that utilizes its own scoring function. The scoring function is at similar accuracy with the AutoDock Vina, yet this software also utilizes search databases of ligands for compounds that mimic the inhibitory binding interactions of an experimentally validated inhibitor. Moreover, it offers the prediction of protein-protein and protein-DNA binding interactions. You can freely download the software at `https://dock.compbio.ucsf.edu/DOCK_6/dock6_manual.htm`.

The above-mentioned software, however, only offers a classical docking score function, where the scoring method was solely based on classical mechanics or empirical approaches. The quantum-mechanical (QM)-based docking method, nowadays, has become very popular, for example, **xtb-dock** (`https://xtb-docs.readthedocs.io/en/latest/xtb_docking.html`), using the automated interaction site screening (aISS) method. So far, the aISS method has been developed at the extended tight-binding (xTB) method, which is claimed to be the next generation of the tight-binding family. Despite it offering a relatively slow calculation speed compared to its predecessor, the xTB method is still sufficiently robust for moderate molecular system size (Up to 500 atoms).

As a brother to the xTB method, the density-functional tight-binding (DFTB) offers a better computational cost that can handle up to 5000 atoms by using a computer with 12 processors. Furthermore, nowadays, the periodic table baseline parameter set has been established for the DFTB method, extending its applicability to handle many types of materials, including rare-earth metals. Interestingly, this extension does not affect the original accuracy of the DFTB method to describe the biomolecular systems. In the present workshop, the performance of the DFTB method will be assessed for the protein-ligand complex of a certain enzymatic reaction.

To handle a large system size consisting of 3000 atoms, the divide-and-conquer (DC) scheme will be utilized. In the DC scheme, the whole system will be divided into several subsystems that can be delivered into several CPU cores. There are several ways to define the subsystems:

- Automatically generating subsystems by defining small boxes that slice the system into many pieces. This approach is the easiest one since the user does not need any scripting or chemical intuition to define the subsystems.
- Defining by using specific labeling for each atom in the systems. This approach is more difficult yet promising for a uniform system, for example, polymers.

The DC method has been published in some of our works [1–7].

2 Login to The Workshop Server

To perform all calculations described in this user manual, you need to access the workshop server by following this procedure:

1. Ensure that you are connected to the ITB VPN server by using the OpenVPN App. Further instructions are given separately during the workshop.
2. Open **Termius** app.
3. Click **NEW HOST**.
4. Fill in the address: 167.205.72.13
5. Typing the label as your preference, e.g., "Sakura Science Club (SSC)".
6. Type the username: ssc
7. Type the password: sakura
8. Click **Connect** (a green square button at the bottom part of the window).

3 Calculation Procedure

1. Copy all necessary files for this tutorial

```
cp -r /home/ssc/SSC_Workshop/Day1 .
```

2. Enter the `Day1` directory

```
cd Day1
```

```
ls
```

The following display will appear:

```
calc_score.py complex enzim ligan Modul structures
```

All adopted structures in this workshop are available in `structures` directory. You can download and visualize the following files:

- `enzim.xyz` ⇒ Adopted enzyme protein structure that consists of 3780 atoms with the total net charge of -1. Note that you need to ensure the total charge of your protein before performing the calculation at DFTB level.
- `ligan.xyz` ⇒ The ligand structure adopted in this workshop. The ligand structure consists of 38 atoms with a neutral net charge.
- `complex.xyz` ⇒ Protein-ligand complex consists of 3818 atoms with a total net charge of -1.

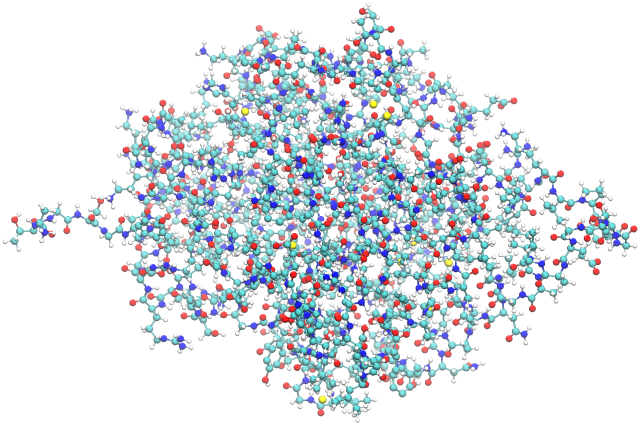


Figure 1: Adopted protein model in the present workshop. The blue, cyan, red, yellow, and white colors indicate nitrogen, carbon, oxygen, sulfur, and hydrogen atoms, respectively.

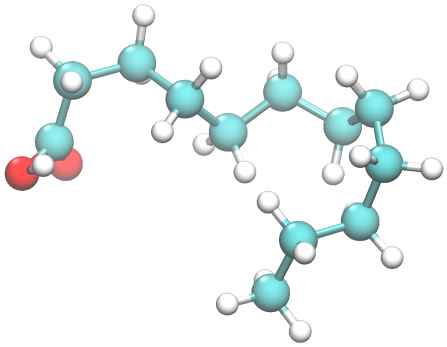


Figure 2: Adopted ligand structure in the present workshop. The blue, cyan, red, yellow, and white colors indicate nitrogen, carbon, oxygen, sulfur, and hydrogen atoms, respectively.

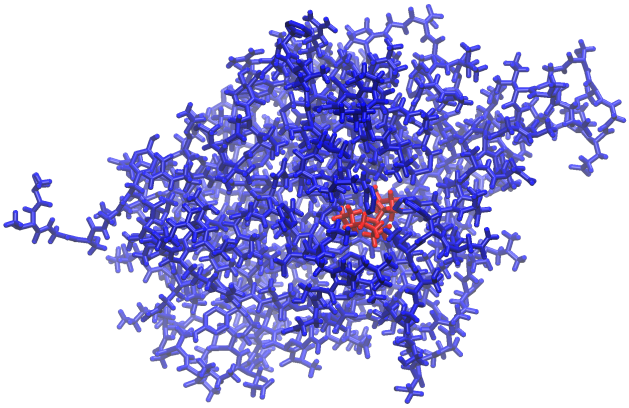


Figure 3: Protein-ligand complex adopted in the present workshop. The blue and red colors represent the protein and ligand molecules, respectively.

3. Enter the `enzim` directory and make the input file.

```
cd enzim
```

```
vi dftb.inp
```

Press I on your keyboard to enter the insert mode (editing mode in the vi editor). Copy and paste the following lines:

```
DC=(SUBTYPE=AUTO DELTARXYZ=3 BUFRAD=3 BETA=400)
SCC=(THIRDFULL=TRUE ECONV=1e-06 DCONV=1e-06 GB=TRUE GBSOLVENTTYPE=1)
DISPERSION=(DISPTYPE=9)

DCDFTB input generated from CMMDE code

5
C 2 -0.1192
C-C.skf C-H.skf C-N.skf C-O.skf C-S.skf
H 1 -0.1860
H-C.skf H-H.skf H-N.skf H-O.skf H-S.skf
N 2 -0.1763
N-C.skf N-H.skf N-N.skf N-O.skf N-S.skf
O 2 -0.1575
O-C.skf O-H.skf O-N.skf O-O.skf O-S.skf
S 3 -0.1441
S-C.skf S-H.skf S-N.skf S-O.skf S-S.skf

3780 -1 1
```

Press ESC. Type in:

```
:r ../structures/enzim.xyz
```

press ENTER. Check whether you have an extra line after you do this procedure. If you don't have an extra line in your input file, you need to create one by hitting the O button (this is the letter "O", not a zero). Then, press ESC again. Save your file by typing:

```
:wq
```

or by simply pressing **SHIFT(hold) + Z + Z**. Either method will work just the same.

4. Make the submission script (`submit.sh`) to the queueing system by typing:

```
vi submit.sh
```

Hit I button. Copy and paste the following lines,

```
#!/bin/bash
#SBATCH --nodes=1
#SBATCH --ntasks=1
#SBATCH --cpus-per-task=12
#SBATCH --time=168:0:0
```

```
source /opt/intel/oneapi/setvars.sh
cd $PWD
mpirun -np 12 dftb_mpiomp.00.x
```

Hit ESC button. Save the file by typing:

```
:wq
```

5. Copy the parameter files

```
cp /home/ssc/SSC_Workshop/parameters/{C,H,O,N,S}-{C,H,O,N,S}.skf .
```

6. Submit the calculation by typing:

```
sbatch submit.sh
```

7. Repeat similar procedure for ligand and complex molecules. Enter the ligand directory and make the input file.

```
cd ../ligan
```

```
vi dftb.inp
```

Hit I button. Copy the following header lines:

```
DC=(SUBTYPE=AUTO DELTARXYZ=3 BUFRAD=3 BETA=400)
SCC=(THIRDFULL=TRUE ECONV=1e-06 DCONV=1e-06 GB=TRUE GBSOLVENTTYPE=1)
DISPERSION=(DISPTYPE=9)

DCDFTB input generated from CMMDE code

3
C 2 -0.1192
C-C.skf C-H.skf C-O.skf
H 1 -0.1860
H-C.skf H-H.skf H-O.skf
O 2 -0.1575
O-C.skf O-H.skf O-O.skf

38 0 1
```

Hit ESC button, then type:

```
:r ../structures/ligan.xyz
```

Press O button, then press ESC button again. Save your file by typing:

```
:wq
```

8. Copy all necessary parameters by typing:

```
cp /home/ssc/SSC_Workshop/parameters/{C,H,O}-{C,H,O}.skf .
```

9. Copy the submission script from the previous `enzim` folder by typing:

```
cp ../enzim/submit.sh .
```

Submit the calculation by typing:

```
sbatch submit.sh
```

10. Enter the `complex` directory,

```
cd ../complex
```

Make the input file by typing:

```
vi dftb.inp
```

Hit I button, copy and paste the following lines:

```
DC=(SUBTYPE=AUTO DELTARXYZ=3 BUFRAD=3 BETA=400)
SCC=(THIRDFULL=TRUE ECONV=1e-06 DCONV=1e-06 GB=TRUE GBSOLVENTTYPE=1)
DISPERSION=(DISPTYPE=9)

DCDFTB input generated from CMMDE code

5
C 2 -0.1192
C-C.skf C-H.skf C-N.skf C-O.skf C-S.skf
H 1 -0.1860
H-C.skf H-H.skf H-N.skf H-O.skf H-S.skf
N 2 -0.1763
N-C.skf N-H.skf N-N.skf N-O.skf N-S.skf
O 2 -0.1575
```

```
O-C.skf O-H.skf O-N.skf O-O.skf O-S.skf
S 3 -0.1441
S-C.skf S-H.skf S-N.skf S-O.skf S-S.skf

3818 -1 1
```

Hit ESC button and type the following line:

```
:r ../structures/complex.xyz
```

Press O button and then hit ESC button again. Save your file by typing:

```
:wq
```

11. Copy all necessary parameter files by typing:

```
cp /home/ssc/SSC_Workshop/parameters/{C,H,O,N,S}-{C,H,O,N,S}.skf .
```

12. Copy the previous submission script,

```
cp ../enzim/submit.sh .
```

then, submit the calculation by typing:

```
sbatch submit.sh
```

13. Back to one folder upper by typing:

```
cd ../
```

14. Here, it is good to make a simple Python script to calculate the binding energy. Type the following:

```
vi calc_score.py
```

Hit I button, copy and paste the following lines:

```
#!/usr/bin/env python3

import sys

protein = sys.argv[1]
ligand = sys.argv[2]
```

```
comp = sys.argv[3]
Hartree2kJ = 2625.5
with open("{}/dftb.out".format(protein),'r') as f:
    for line in f:
        if "Final␣DC-DFTB-3rd␣Energy" in line:
            arr = line.split()
            E_protein = float(arr[4]) # Extracting the energy


with open("{}/dftb.out".format(ligand),'r') as f:
    for line in f:
        if "Final␣DC-DFTB-3rd␣Energy" in line:
            arr = line.split()
            E_ligand = float(arr[4]) # Extracting the energy

with open("{}/dftb.out".format(comp),'r') as f:
    for line in f:
        if "Final␣DC-DFTB-3rd␣Energy" in line:
            arr = line.split()
            E_complex= float(arr[4]) # Extracting the energy


Score = E_complex - E_ligand - E_protein
Score = Score*Hartree2kJ # Converting the energy to kJ/mol

print("Docking␣Score␣=␣{:.3f}␣kJ/mol".format(Score))
```

Hit ESC button and save the file by typing:

```
:wq
```

15. Make the python script becomes an executable by typing:

```
chmod +x calc_score.py
```

16. Calculate the docking score by typing:

```
./calc_score.py enzim ligan complex
```

The display in your terminal would be:

```
Docking Score = -4.255 kJ/mol
```

It indicates that the binding energy of the enzyme and the adopted ligand structure in water solvent is -4.255 kJ/mol.

References

[1] Ken-ichi Otake et al. "Confined water-mediated high proton conduction in hydrophobic channel of a synthetic nanotube". en. In: *Nature Communications* 11.1 (Feb. 2020). Number: 1 Publisher: Nature Publishing Group, p. 843. ISSN: 2041-1723. DOI: 10.1038/s41467-020-14627-z. URL: https://www.nature.com/articles/s41467-020-14627-z (visited on 02/27/2024).

[2] Aditya Wibawa Sakti et al. "Density-Functional Tight-Binding Molecular Dynamics Simulations of Excess Proton Diffusion in Ice Ih, Ice Ic, Ice III, and Melted Ice VI Phases". In: *The Journal of Physical Chemistry A* 122.1 (Jan. 2018), pp. 33–40. ISSN: 1089-5639. DOI: 10.1021/acs.jpca.7b10664. URL: http://dx.doi.org/10.1021/acs.jpca.7b10664 (visited on 01/30/2018).

[3] Aditya Wibawa Sakti et al. "Divide-and-Conquer Density-Functional Tight-Binding Molecular Dynamics Study on the Formation of Carbamate Ions during CO2 Chemical Absorption in Aqueous Amine Solution". In: *Bulletin of the Chemical Society of Japan* 90.11 (Aug. 2017), pp. 1230–1235. ISSN: 0009-2673. DOI: 10.1246/bcsj.20170142. URL: http://www.journal.csj.jp/doi/abs/10.1246/bcsj.20170142 (visited on 10/23/2017).

[4] Aditya Wibawa Sakti, Yoshifumi Nishimura, and Hiromi Nakai. "Divide-and-Conquer-Type Density-Functional Tight-Binding Simulations of Hydroxide Ion Diffusion in Bulk Water". In: *The Journal of Physical Chemistry B* 121.6 (Feb. 2017). Publisher: American Chemical Society, pp. 1362–1371. ISSN: 1520-6106. DOI: 10.1021/acs.jpcb.6b10659. URL: https://doi.org/10.1021/acs.jpcb.6b10659 (visited on 02/27/2024).

[5] Aditya Wibawa Sakti et al. "Effects of Salt Concentration on the Water and Ion Self-Diffusion Coefficients of a Model Aqueous Sodium-Ion Battery Electrolyte". In: *The Journal of Physical Chemistry B* 126.11 (Mar. 2022). Publisher: American Chemical Society, pp. 2256–2264. ISSN: 1520-6106. DOI: 10.1021/acs.jpcb.1c09619. URL: https://doi.org/10.1021/acs.jpcb.1c09619 (visited on 02/27/2024).

[6] Hiromi Nakai, Aditya Wibawa Sakti, and Yoshifumi Nishimura. "Divide-and-Conquer-Type Density-Functional Tight-Binding Molecular Dynamics Simulations of Proton Diffusion in a Bulk Water System". In: *The Journal of Physical Chemistry B* 120.1 (Jan. 2016), pp. 217–221. ISSN: 1520-6106. DOI: 10.1021/acs.jpcb.5b12439. URL: http://dx.doi.org/10.1021/acs.jpcb.5b12439 (visited on 01/14/2016).

[7] Dhea Afrisa Darmawan et al. "Fabrication of solid polymer electrolyte based on carboxymethyl cellulose complexed with lithium acetate salt as Lithium-ion battery separator". en. In: *Polymer Composites* 45.3 (2024). _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/pc.27902, pp. 2032–2049. ISSN: 1548-0569. DOI: 10.1002/pc.27902. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/pc.27902 (visited on 11/17/2024).