

CaFE Manual

Release 1.0

Hui Liu, Tingjun Hou

March 23, 2016

CONTENTS

| | | |
|----------|---------------------------------------|-----------|
| 1 | Installation | 1 |
| 1.1 | Prerequisites | 1 |
| 1.2 | Download and install | 2 |
| 2 | Applications | 3 |
| 2.1 | The <i>mmpbsa</i> procedure | 3 |
| 2.2 | The <i>lie</i> procedure | 9 |
| 3 | About CaFE | 13 |
| 3.1 | Citation | 13 |
| 3.2 | License | 13 |
| | Bibliography | 14 |

INSTALLATION

1.1 Prerequisites

- [VMD](#) 1.9.2 or later

Note: There is a bug in the `topotools1.5` plugin, which is included in VMD 1.9.2 and invoked by CaFE. You should change line 43 in:

Linux/MacOSX

```
$VMDDIR/lib/plugins/noarch/tcl/topotools1.5/topoatoms.tcl
```

Windows

```
$VMDDIR/plugins/noarch/tcl/topotools1.5/topoatoms.tcl
```

from:

```
$s set element [lindex $elements [ptefrommass $idx]]
```

to:

```
$s set element [lindex $elements [ptefrommass $a]]
```

Alternatively, you can copy the corrected `topoatoms.tcl` file in the `patch` folder, which is distributed with the CaFE source code, and replace the associated file in the `topotools1.5` plugin folder in your installed VMD distribution.

- [NAMD](#) 2.9 or later
- [APBS](#) 1.3 or later
Optionally, for MM/PBSA only
- [DelPhi](#) 5.1 or later
Optionally, for MM/PBSA only

All these tools are commonly used in molecular simulations, please download the installation files and follow the corresponding instructions on their websites. Make sure that all the folders containing the binary files are appended to the `PATH` environment variable.

1.2 Download and install

After installing the above required programs, you could download the CaFE source code: `CaFE_Plugin-master.zip` from https://github.com/HuiLiuCode/CaFE_Plugin.

If you have root access for installation, just extract the compressed contents and rename the `src` folder to `cafe1.0` and copy it to `$VMDDIR/lib/plugins/noarch/tcl` (Linux/MacOSX) or `$VMDDIR/plugins/noarch/tcl` (Windows), where `$VMDDIR` is the directory in which VMD installed.

If you don't have root access, try to extract the contents and rename the `src` folder to `cafe1.0` and copy it to somewhere in your `$HOME` directory. Then, add the following to the VMD start-up file `$HOME/.vmดรc` (Linux/MacOSX) or `vmd.rc` (Windows):

```
set auto_path [linsert $auto_path 0 {/PATH/TO/YOUR/FOLDER}]
```

APPLICATIONS

CaFE is an automatic pipeline tool for post-processing and energetic analysis, which is powered by [VMD \[HW96\]](#) (page 14). Prior to the calculations, topology and trajectory files are needed to be generated by common molecular dynamics (MD) simulation software. Currently, [AMBER \[CD05\]](#) (page 14) and [NAMD \[PJ05\]](#) (page 14) are generally supported. Other simulation packages using the same topology files are supported, too.

Note: If AMBER is used, you'd better set "iwrap=1" in the input file or use *cpptraj* to "image" the generated trajectory. If NAMD is used, you'd better set "wrapAll on" in the configuration file.

CaFE is implemented as a set of Tcl scripts. When using it, you should import the *cafe* package by appending the following line in the analysis scripts first:

```
package require cafe 1.0
```

After that, two major procedures can be invoked for binding free energy calculations.

2.1 The *mmpbsa* procedure

The **mmpbsa** procedure automates post-processing and energetic analysis using the molecular mechanics Poisson-Boltzmann surface area (MM/PBSA) method [\[KP00\]](#) (page 14). A trajectory file containing the binding complex is required, and then the receptor and ligand will be extracted from that according to the one-trajectory protocol [\[KP00\]](#) (page 14). For MM/PBSA calculations, either [APBS \[BN01\]](#) (page 14) or [DelPhi \[LL12\]](#) (page 14) is needed to be installed.

2.1.1 Usage

```
mmpbsa -top file -trj file [-args ...]
```

Mandatory arguments

- **-top:** topology file name
available values: a valid file name
default value: none
- **-trj:** trajectory file name
available values: a valid file name

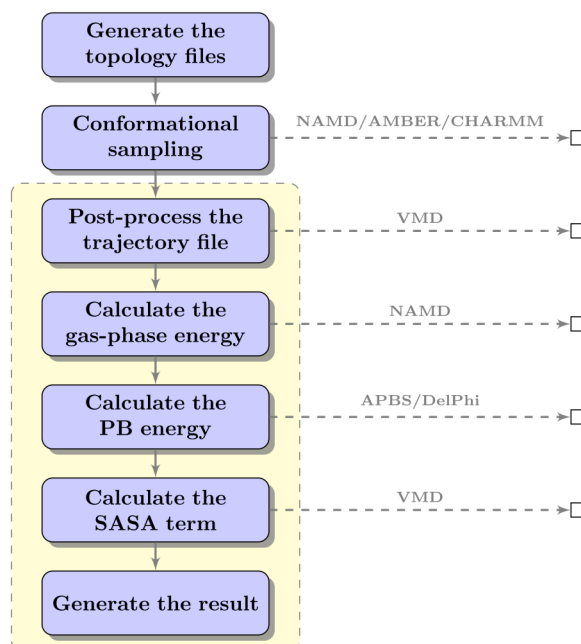


Fig. 2.1: **Work flow for MM/PBSA calculations.** The procedures in yellow background are automated by CaFE.

default value: none

Optional arguments

- **-top_type:** topology file type
available values: `parm7` (AMBER format), `psf` (CHARMM/NAMD format), `auto`
default value: `auto`
- **-trj_type:** trajectory file type
available values: all of the coordinate and trajectory file types supported by VMD
default value: `auto`
- **-par:** CHARMM-formatted force field parameter file, required only if the PSF topology type is used
available values: a valid file name
default value: `$CAFEDIR/par_all122_prot.inp`
- **-out:** output file name
available values: a valid file name
default value: `result.log`
- **-debug:** debug level
available values: 0 (remove all of the intermediate files), 1 (remove a part of them), 2 (keep all)
default value: 0

- **-first**: the first frame index, starting from 0
available values: an integer ≥ 0
default value: 0
- **-last**: the last frame index, starting from 0
available values: an integer ≥ 0 or -1 (stands for the last frame of the trajectory)
default value: -1
- **-stride**: stride
available values: an integer ≥ 1
default value: 1
- **-com**: complex selection
available values: a valid selection string used by VMD
default value: ""
- **-rec**: receptor selection
available values: a valid selection string used by VMD
default value: ""
- **-lig**: ligand selection
available values: a valid selection string used by VMD
default value: ""
- **-mm**: do gas-phase calculations or not
available values: 0 (do not) or 1 (do)
default value: 0
- **-mm_exe**: path to the NAMD binary
available values: a valid path
default value: "namd2"
- **-mm_diel**: dielectric constant
available values: a float
default value: 1.0
- **-pb**: do PB calculations or not
available values: 0 (do not), 1 (use DelPhi), 2 (use APBS)
default value: 0
- **-pb_exe**: path to the DelPhi/APBS binary
available values: a valid path
default value: "delphi77"
- **-pb_siz**: radii parameter file, not a necessity (DelPhi-only)
available values: a valid file name
default value: ""

- **-pb_crg:** charge parameter file, not a necessity (DelPhi-only)
available values: a valid file name
default value: " "
- **-pb_rad:** type of PB radii
available values: bondi, rowland, mparse, parse, charmm, roux, parm7 (valid only if the topology is an AMBER parm7 file, using the radii included in it)
default value: bondi
- **-pb_indi:** internal dielectric constant
available values: a float
default value: 1.0
- **-pb_exdi:** external dielectric constant
available values: a float
default value: 80.0
- **-pb_scale:** the reciprocal of grid spacing
available values: a float
default value: 2.0
- **-pb_perfil:** percentage of filling
available values: a float
default value: 80.0
- **-pb_prbrad:** probe radius
available values: a float
default value: 1.4
- **-pb_linit:** max number for linear iterations
available values: an integer > 3
default value: 1000
- **-pb_maxc:** convergence threshold
available values: a float > 0.0
default value: 0.0001
- **-pb_bndcon:** boundary condition (DelPhi-only)
available values: 1 (zero boundary condition), 2 (dipolar), 4 (coulombic)
default value: 4
- **-pb_bcfl:** boundary condition (APBS-only)
available values: zero (zero boundary condition), sdh (single Debye-Huckel), mdh (multiple Debye-Huckel)
default value: sdh

- **-pb_chgm**: the method that charges are mapped to the grids (APBS-only)
available values: `sp10` (linear splines), `sp12` (cubic B-spline), `sp14` (quintic B-spline)
default value: `sp10`
- **-pb_srfm**: the model used to construct the dielectric and ion-accessibility coefficients (APBS-only)
available values: `mol`, `smol`, `sp12`, `sp14`
default value: `smol`
- **-pb_swin**: spline window width (APBS-only)
available values: a float > 0.0
default value: `0.3`
- **-pb_sdens**: density of grids (APBS-only)
available values: a float > 0.0
default value: `10.0`
- **-sa**: do SA calculations or not
available values: `0` (do not), `1` (use VMD)
default value: `0`
- **-sa_rad**: type of SA radii
available values: `bondi`, `rowland`, `mparse`, `parse`, `charmm`, `roux`, `parm7` (valid only if the topology is an AMBER `parm7` file, using the radii included in it)
default value: `bondi`
- **-sa_gamma**: surface tension
available values: a float
default value: `0.005`
- **-sa_beta**: surface offset
available values: a float
default value: `0.0`
- **-sa_prbrad**: probe radius
available values: a float
default value: `1.4`
- **-sa_samples**: number of samples
available values: an integer > 0
default value: `500`

For more details about the parameters, please see the manuals of [VMD](#) (supported file types, selection string, SA calculations) and [APBS/DelPhi](#) (PB calculations).

2.1.2 Example

Now, suppose that we have performed the conformational sampling by using MD simulations, there are two folders in the working folder `workdir`:

1. `md`, which contains topology file `com.psf`, and generated trajectory `com.dcd`;
2. `toppar`, which contains force field parameter files.

In the PSF file, the receptor and ligand have been put into segments named “PRO” and “LIG”, respectively. The trajectory file is totally 1 ns long and saved every 100 ps. We want to perform MM/PBSA calculations from 0 to 1 ns and produce snapshots every 100 ps. APBS is used in the PB calculations.

A typical analysis script is something like this:

```
package require cafe 1.0

mmpbsa -top      ../md/com.psf \
        -trj      ../md/com.dcd \
        -out      mmpbsa.log \
        -par      ../toppar/par_all22_prot.prm \
        -par      ../toppar/par_all36_cgenff.prm \
        -par      ../toppar/toppar_water_ions_namd.str \
        -par      ../toppar/lig.str \
        -com      "segname PRO LIG" \
        -rec      "segname PRO" \
        -lig      "segname LIG" \
        -first    0 \
        -last     -1 \
        -stride   1 \
        -mm       1 \
        -pb       2 \
        -pb_exe   apbs \
        -pb_rad   mparse \
        -pb_bcf1  mdh \
        -pb_chgm  spl2 \
        -sa       1 \
        -sa_rad   mparse \
        -sa_gamma 0.00542 \
        -sa_beta  0.92

quit
```

Create a new folder `mmpbsa` in `workdir` and save the above contents to a file named `mmpbsa.vmd` in it.

Note: Don't forget the backslash!

Then run commands as follows in the terminal, which will perform the MM/PBSA calculations and generate results in plain text:

```
$ cd workdir/mmpbsa
$ vmd -dispdev text -eofexit < mmpbsa.vmd > vmd.log
```

The output file `mmpbsa.log`, which contains the calculated binding free energy, is self-explanatory. Files that are required to perform the exemplified calculations are in the `examples` folder distributed alongside the source code.

2.2 The *lie* procedure

The **lie** procedure automates post-processing and energetic analysis using the linear interaction energy (LIE) method [AJ94] (page 14). Two topology as well as MD trajectory files containing the solvated complex and ligand are required.

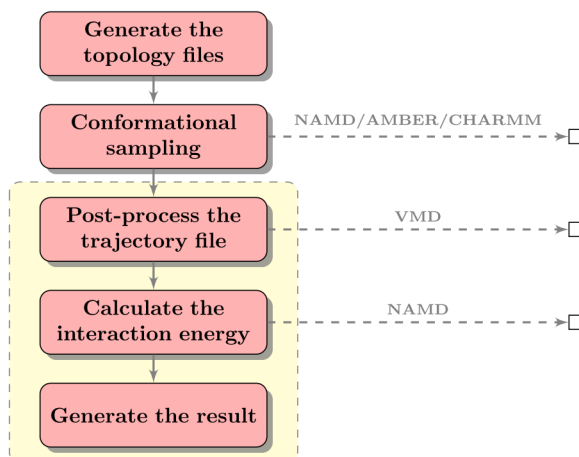


Fig. 2.2: **Work flow for LIE calculations.** The procedures in yellow background are automated by CaFE.

2.2.1 Usage

```
lie -top_bound file -trj_bound file -top_free file -trj_free file [-args ...]
```

Mandatory arguments

- **-top_bound**: topology file name for the bound state
available values: a valid file name
default value: none
- **-trj_bound**: trajectory file name for the bound state
available values: a valid file name
default value: none
- **-top_free**: topology file name for the free state
available values: a valid file name
default value: none
- **-trj_free**: trajectory file name for the free state
available values: a valid file name
default value: none

Optional arguments

- **-top_type**: topology file type
available values: `parm7` (AMBER format), `psf` (CHARMM/NAMD format), `auto`
default value: `auto`
- **-trj_type**: trajectory file type
available values: all of the coordinate and trajectory file types supported by VMD
default value: `auto`
- **-par**: CHARMM-formatted force field parameter file, required only if the PSF topology type is used
available values: a valid file name
default value: `$CAFEDIR/par_all122_prot.inp`
- **-out**: output file name
available values: a valid file name
default value: `result.log`
- **-debug**: debug level
available values: 0 (remove all of the intermediate files), 1 (remove a part of them), 2 (keep all)
default value: 0
- **-first_bound**: the first frame index for the bound state, starting from 0
available values: an integer ≥ 0
default value: 0
- **-last_bound**: the last frame index for the bound state, starting from 0
available values: an integer ≥ 0 or -1 (stands for the last frame of the trajectory)
default value: -1
- **-stride_bound**: stride for the bound state
available values: an integer ≥ 1
default value: 1
- **-lig_bound**: ligand selection for the bound state
available values: a valid selection string used by VMD
default value: `" "`
- **-first_free**: the first frame index for the free state, starting from 0
available values: an integer ≥ 0
default value: 0
- **-last_free**: the last frame index for the free state, starting from 0
available values: an integer ≥ 0 or -1 (stands for the last frame of the trajectory)
default value: -1

- **-stride_free**: stride for the free state
available values: an integer ≥ 1
default value: 1
- **-lig_free**: ligand selection for the free state
available values: a valid selection string used by VMD
default value: ""
- **-mm_exe**: path to the NAMD binary
available values: a valid path
default value: "namd2"
- **-alpha**: van der Waals coefficient
available values: a float
default value: 0.18
- **-beta**: electrostatic coefficient
available values: a float
default value: 0.33
- **-gamma**: offset
available values: a float
default value: 0.0

For more details about the parameters, please see the manual of [VMD](#) (supported file types, selection string).

2.2.2 Example

Now, suppose that we have performed the conformational sampling by using MD simulations, there are two folders in the working folder `workdir`:

1. `md`, which contains topology files `com.psf` and `lig.psf`, and generated trajectories `com.dcd` and `lig.dcd`;
2. `toppar`, which contains force field parameter files.

In both the PSF files, the ligand has been put into a segment named "LIG". Each of the two trajectory files is 1 ns long and saved every 100 ps. We want to perform LIE calculations from 0 to 1 ns and produce snapshots every 100 ps.

A typical analysis script is something like this:

```
package require cafe 1.0

lie -top_bound      ../md/com.psf \
    -trj_bound      ../md/com.dcd \
    -top_free       ../md/lig.psf \
    -trj_free       ../md/lig.dcd \
    -out            lie.log \
    -par            ../toppar/par_all22_prot.prm \
    -par            ../toppar/par_all36_cgenff.prm \
    -par            ../toppar/toppar_water_ions_namd.str \
```

```
-par          ../toppar/lig.str \  
-lig_bound    "segname LIG" \  
-lig_free     "segname LIG" \  
-first_bound  0 \  
-last_bound   -1 \  
-stride_bound 1 \  
-first_free   0 \  
-last_free    -1 \  
-stride_free  1 \  
-alpha        0.18 \  
-beta         0.5 \  
-gamma        0.0
```

quit

Note: The values of `alpha`, `beta` and `gamma` are dependent on your simulated systems. Please see related references for the choice of values.

Create a new folder `lie` in `workdir` and save the above contents to a file named `lie.vmd` in it.

Note: Don't forget the backslash!

Then run commands as follows in the terminal, which will perform the LIE calculations and generate results in plain text:

```
$ cd workdir/lie  
$ vmd -dispdev text -eofexit < lie.vmd > vmd.log
```

The output file `lie.log`, which contains the calculated binding free energy, is self-explanatory. Files that are required to perform the exemplified calculations are in the `examples` folder distributed alongside the source code.

ABOUT CAFE

CaFE is a free and open-source VMD plugin for binding affinity prediction using end-point free energy methods. It is being developed in [Hou Lab](#) at [Zhejiang University](#).

3.1 Citation

If CaFE is utilized in scientific work that results in a publication, it is expected that the following reference is cited:

Liu H, Hou T.

CaFE: a tool for binding affinity prediction using end-point free energy methods.

In submission

3.2 License

CaFE is freely available under the [GPL License](#):

```
CaFE: Calculation of Free Energy
```

```
Copyright (C) 2014-2016 Zhejiang University
```

```
This program is free software: you can redistribute it and/or modify  
it under the terms of the GNU General Public License as published by  
the Free Software Foundation, either version 3 of the License, or  
(at your option) any later version.
```

```
This program is distributed in the hope that it will be useful,  
but WITHOUT ANY WARRANTY; without even the implied warranty of  
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the  
GNU General Public License for more details.
```

```
You should have received a copy of the GNU General Public License  
along with this program. If not, see <http://www.gnu.org/licenses/>.
```

BIBLIOGRAPHY

- [KP00] Kollman, P.A., et al. (2000) Calculating structures and free energies of complex molecules: Combining molecular mechanics and continuum models, *Acc. Chem. Res.*, 33, 889-897.
- [BN01] Baker, N.A., et al. (2001) Electrostatics of nanosystems: Application to microtubules and the ribosome, *Proc. Natl. Acad. Sci.*, 98, 10037-10041.
- [LL12] Li, L., et al. (2012) Delphi: A comprehensive suite for delphi software and associated resources, *BMC Biophys.*, 5, 9.
- [HW96] Humphrey, W., Dalke, A. and Schulten, K. (1996) VMD: Visual molecular dynamics, *J. Mol. Graph.*, 14, 33-38.
- [AJ94] Åqvist, J., Medina, C. and Samuelsson, J.-E. (1994) A new method for predicting binding affinity in computer-aided drug design, *Protein Eng.*, 7, 385-391.
- [HW96] Humphrey, W., Dalke, A. and Schulten, K. (1996) VMD: Visual molecular dynamics, *J. Mol. Graph.*, 14, 33-38.
- [PJ05] Phillips, J.C., et al. (2005) Scalable molecular dynamics with NAMD, *J. Comput. Chem.*, 26, 1781-1802.
- [CD05] Case, D.A., et al. (2005) The amber biomolecular simulation programs, *J. Comput. Chem.*, 26, 1668-1688.