Solvation Gibbs Energy Calculation

To run, PyGBe needs two input files: a ***config*** file (`*.config`) and a ***parameters*** file (`*.param`), which need to be located in the `./bem_pycuda/input_files` directory. The following shows an example of a run script:

```
python main_asymmetric.py input_files/problem.param input_files/problem.config --asymmetric
```

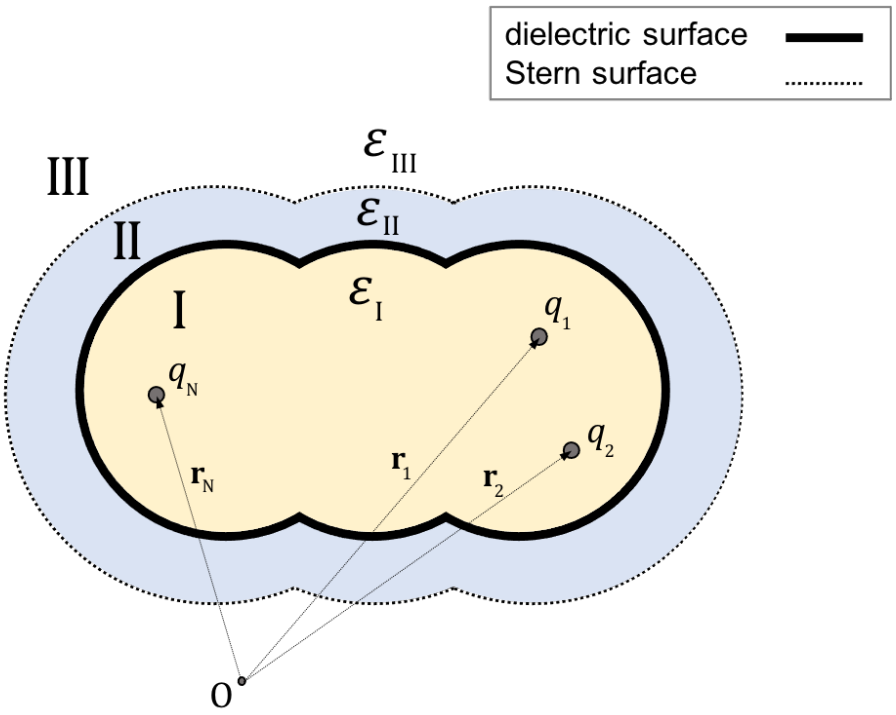
1. The first argument (`main_asymmetric.py`) specifies the python script the carries out solvation free energy calculations
2. The second argument (`input_files/problem.param`) specifies the relative path to the**param** file location
3. The third argument (`input_files/problem.config`) specifies the relative path to the**config** file location
4. The `--asymmetric` flag activates the asymmetric SLIC boundary condition
5. The `--chergeForm` denotes the type of charge distribution

config file

config file specifies the geometry and physical properties of the system. The following is an example of a config file.

```
FILE    path/to/stern_mesh  stern_layer
FILE    path/to/diel_mesh   dielectric_interface
--------------------------------
PARAM   LorY E?  Dielec  kappa  charges?  coulomb?    charge_file       Nparent  parent  N
FIELD   1    0   78.36   1e-12  0         0           NA                0        NA      1
FIELD   1    0   78.36   1e-12  0         0           NA                1        0       1
FIELD   1    1   1       1e-12  1         1           path/to/mol.pqr   1        1       0
```

which corresponds to the system shown below:



config file consists of two parts, separated with dashes, as seen above.

`FILE` section

Lines starting with `FILE` indicate that the line specifies the geometry and type of one interface. The first arguement after `FILE` determines the relative path to the interface mesh files,`filename.vert` and `filename.face`. `filename.vert` containes the location of the vertices of the disretized traingular elements and `filename.vert` is the connectivity table for the vertices. The `*.face` and `*.vert` files can be generated using Michael Sanner's Molecular Surface program

MSMS. This link shows an example on how to use MSMS. To account for more than one surface (ie. for Stern layers, solvent filled cavities, several proteins), more than one FILE line is needed. For instance, the sample config file accounts for two surface, described in lines 1 and 2. After `filename`, the user must specify what kind of surface the `filename` describes. It can be:

- `stern_layer`: surface separates Stern layer (region shown in) from solvent
- `dielectric_interface`: surface separates low dielectric (inside protein shown in) and high dielectric (outside protein) regions.
- `internal_cavity`: surface is an internal cavity (a result of `'-all_components'` flag in MSMS). This is important to specify because by default MSMS changes the vertex ordering for internal cavities.
- `dirichlet_surface`: surface of specified potential. The value of this potential is read from a text file which has to be specified next to *'dirichlet_surface'*.
- `neumann_surface`: surface of specified potential. The value of this potential is read from a text file which has to be specified next to *'neumann_surface'*.

The first line in the sample config file specifies

1. the relative path to the Stern surface (shown with dots in the schematic) geometry where `path/to` is a folder containing `stern_mesh.face` and `stern_mesh.vert`.
2. the kind of surface which is `stern_layer`.

Similarly, the second line in the sample config file specifies this information for the dielectric interface (shown with thick black line in the schematic).

`FIELD` section

Lines starting with `FIELD` indicate that the line specifies physical parameters of one region. These parameters include:

- `LorY` indicates that the electrostatic potential in the region is
  - 1: Laplace
  - 2: Yukawa

- `E?`
  - 0: don't calculate the energy in this region
  - 1: calculate the energy in this region

  Note: if region is surrounded by a *dirichlet* or *neumann surface*, surface energy will be calculated.

- `Dielec`: dielectric constant of the region

- `kappa`: inverse Debye-length

- `charges?`
  - 0: no charges inside the region
  - 1: there are charges in the region

- `coulomb?`
  - 0: don't calculate Coulomb energy in this region
  - 1: calculate Coulomb energy in this region

- `charge_file`
  - `NA` if the region does not contain charges
  - relative path to the `*.pqr` file if the region contains charges. For example, the dielectric region in the schematic above (region `I` shown in) has a corresponding charge distribution file `mol.pqr` located at `path/to/` directory relative to the config file location.

- `Nparent`: number of *'parent'* surfaces (surface containing this region).
  - 0: if the region corresponds to an infinite region, e.g., the solvent region `III` in the schematic

    - 1: if the region is bounded by a surface, e.g., the Stern layer `II` bounded by the Stern surface in the schematic

- `parent`: the region's parent surface mesh index (starting from 0), according to their position in the `FILE` section. Takes `NA` if the region is infinite, otherwise an `int`. For instance in the sample config file, the first line in the `FIELD` section describes the properties of solvent as an infinite region (region `III` in the schematic). Because the solvent region is infinite and not bounded by another surface, the `parent` parameter takes the value `NA`. Similarly, the second line in the `FIELD` section corresponds to the Stern layer (shown in in the schematic). This region is bounded by the Stern surface, therefore, the `parent` parameter takes the value `0` because the Stern surface is the first surface (hence the `0` index) defined in the `FILE` section.

- `Nchild`: number of child surfaces, i.e., surfaces completely contained in this region. The Stern surface in the schematic is completely contained in the solvent region `III` so the `Nchild` parameter for the solvent region is `1`. Similarly, the dielectric surface is contained in the Stern layer (region `II`) so the `Nchild` parameter for the Stern layer is `1`.

- `children`: mesh file index (indices if the region has more than one child) for the children surfaces completely contained in this region, using the same convention as the `parent` parameter.

Param file

parameters file controls the computational parameters of the job. The sollowing shows an example of a `*.param` file

```
Precision   double
K           7
Nk          13
K_fine      19
thresold    0.5
BSZ         128
restart     500
tolerance   1e-5
max_iter    1000
P           6
eps         1e-12
NCRIT       500
theta       0.5
GPU         1
```

- `Precision`: double or float. (float not supported yet!).

- `K`: number of Gauss quadrature points per element (1, 3, 4, and 7 are supported).

- `Nk`: number of Gauss quadrature points per triangle edge for semi-analytical integration.

- `threshold`: defines region near singularity where semi-analytical technique is used. if `sqrt(2*Area)/r > threshold`, integration is done semi-analytically.

- `BSZ`: CUDA block size.

- `restart`: number of iterations for GMRES to do restart.

- `tolerance`: GMRES tolerance.

- `max_iter`: maximum number of GMRES iterations.

- `P`: order of expansion in treecode.
- `eps`: epsilon machine.
- `NCRIT`: maximum number of boundary elements per twig box of tree structure.
- `theta`: multipole acceptance criterion of treecode.
- `GPU`
  - 0: don't use GPU.
  - 1: use GPU.

More details regarding the parameter definitions can be found in this paper.