# User guide for FEtool.py – v1.0

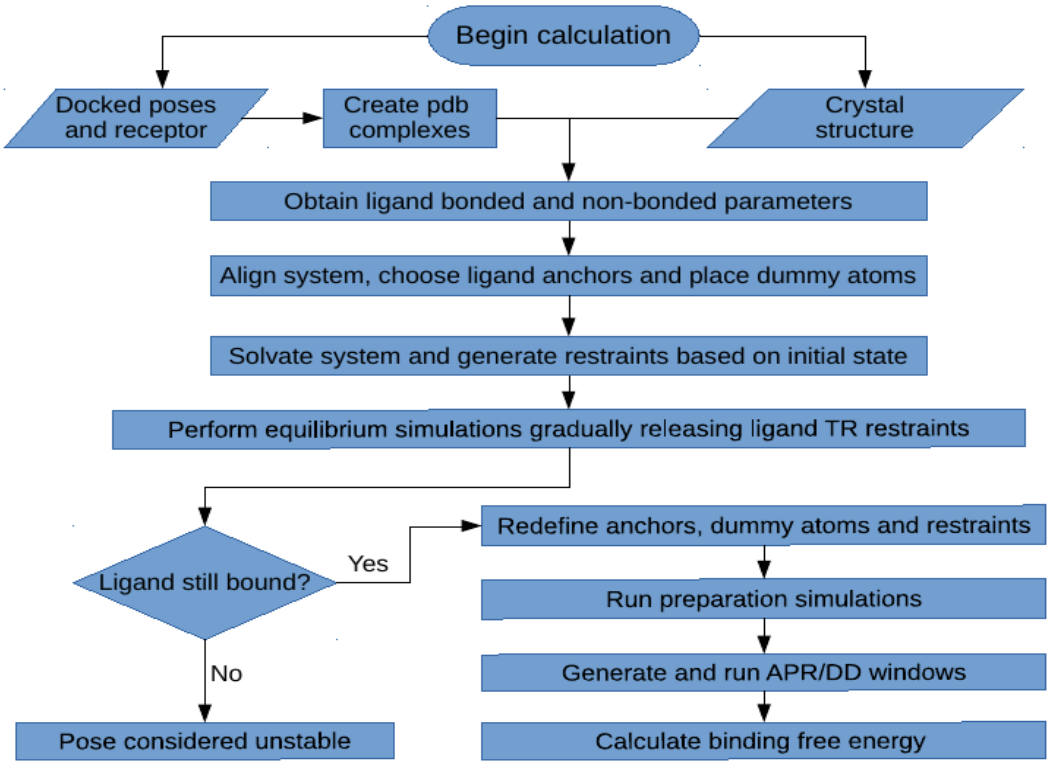
## 1. Introduction

The program FEtool.py is designed to fully automate absolute binding free energy calculations, starting only from a crystal structure or a docked complex. The building of the simulation boxes, generation of all the needed parameters, set up of the various simulation windows, running the simulations, and the final free energy analysis are all done without any manual interference. FEtool.py uses the *pmemd.cuda* sofware from AMBER, which has shown very high performance on Graphics Processing Units (GPUs) at a reduced cost (http://ambermd.org/gpus16/benchmarks.htm). We believe that our implementation can be used for high-throughput search of high-affinity ligands to a given receptor, using a rigorous physics-based free energy approach. FEtool.py can also be applied for parameter testing and optimization, as well as the comparison of alchemical (double decoupling -DD) and physical routes (attach-pull-release - APR) for standard binding free energy calculations.

In this user guide we will first describe the workflow of the program, then the various components of the free energy calculation, and how the simulations are analyzed in order to obtain the quantities of interest. All the parameters needed for the program input file, and how they apply to the various calculation steps, will also be described in detail.

## 2. Workflow

The FEtool.py protocol follows the workflow shown below:



The calculations start by first building the initial complex, starting from the docked receptor or ligand files, or directly from the receptor-ligand co-crystal structure. After that the parameters are obtained using Antechamber, with the General Amber Force-Field (GAFF) for the bonded and LJ parameters, and the AM1-BCC model for the partial atomic charges. The system is then aligned to a reference structure of a similar protein using the program MUSTANG, so that the ligand anchors and dummy atom coordinates can be automatically assigned.

With all the coordinates of the initial system already set, the complex is then placed in a water box with a given ion concentration, and the necessary restraints are applied. An initial equilibration is then performed, with all the receptor restraints activated, and the translational/rotational restraints of the ligand being gradually released in order to find a nearby energy minimum. At the end of this last step, the ligand might still be bound or it might have left the binding site in the case of unstable binding mode. If the latter happens, this pose is considered unstable and no further simulations are performed for this system.

If the ligand is still bound after equilibration, which should usually be the case, then the preparation of the system for the binding free energy calculations is performed. The preparation starts from the last state from equilibrium, reassigning the ligand anchors, repositioning the dummy atoms, solvating/ionizing the system and and redefining the restraints. This is necessary since the unrestrained ligand can adopt a different binding mode in the last stage of the equilibration step, which requires a new reference state for the free energy calculations. The preparation simulations may involve the pulling of the ligand from the binding site to bulk, if APR is to be used, or only the simulation of the restrained ligand in the bound state, if only DD is employed.

Starting from the last state of the preparation step, all the necessary simulation windows are now created for the binding free energy calculation. They involve various components for the application/removal of restraints, as well as the pulling/decoupling of the ligand, which are explained in detail in the next section. Once the free energy simulations are concluded, they can be analyzed using the Multistate Bennett Acceptance Ratio (MBAR), or Thermodynamic Integration with Gaussian Quadrature (TI), depending on the stage and the choice in the input file.

## 3. Components

The FEtool.py expression for the calculated binding free energy is defined as follows:

$$-\Delta G^{o}_{bind} = \Delta G_{p,conf,att} + \Delta G_{l,conf,att} + \Delta G_{l,TR,att} + \Delta G_{transfer} + \Delta G_{l,TR,rel} + \Delta G_{l,conf,rel} + \Delta G_{p,conf,rel}$$

In the expression above, the *att* index denotes attachment of restraints in the bound state, and *rel* indicates release of restraints with the ligand in bulk. The *l* and *p* indexes are for ligand and protein (receptor), respectively, *conf* is for conformational restraints and *TR* is for translational/rotational restraints. The $\Delta G_{transfer}$ term is the free energy of transferring of the ligand from the receptor binding site to bulk with all restraints applied, using either a physical reaction coordinate (APR), or an alchemical transformation (DD):

$$\Delta G_{transfer-APR} = \Delta G_{pull} \qquad \textbf{(APR)}$$

$$\Delta G_{transfer-DD} = \Delta G_{dec,elec,site} + \Delta G_{dec,LJ,site} - \Delta G_{dec,elec,bulk} - \Delta G_{dec,LJ,bulk} \qquad \textbf{(DD)}$$

In the case of APR, $\Delta G_{transfer-APR}$ is equal to the pulling free energy of the ligand from the binding site to

bulk, which is done using umbrella sampling, as in Ref []. For the double decoupling procedure, $\Delta G_{transfer\text{-}DD}$ is equal to the sum of four terms, as shown in the equation above. The index *dec* stands for decoupling, *elec* for electrostatic interactions and *LJ* for Lennard-Jones interactions, with these calculations being performed both in the binding site or in bulk.