AMMOS_ProtLig

User's Manual

What is AMMOS

AMMOS (**A**utomatic **M**olecular **M**echanics **O**ptimization for *in silico* **S**creening) [1] employs an automatic procedure for energy minimization of protein-ligands complexes (package AMMOS_ProtLig) or of small chemical compounds present in a library (package AMMOS_SmallMol). As such, the software offers valuable solutions to assist structure-based *in silico* screening experiments or ligand-based projects. The package makes use of molecular mechanics concepts and is based on the program AMMP [2-3], available under GNU license (http://www.cs.gsu.edu/~cscrwh/ammp/ammp.html). AMMOS has been developed in the INSERM – University Paris Diderot, lab. Bioinformatics-MTI (http://www.vls3d.com/). The package AMMOS is written in C and Python and is available for Linux and Mac OS X systems under the GNU General Public license.

Please, cite:

Tania Pencheva, David Lagorce, Ilza Pajeva, Bruno O Villoutreix and Maria A Miteva.
AMMOS: Automated Molecular Mechanics Optimization tool for *in silico* Screening

You may direct questions related to this package to the corresponding author:
Maria Miteva, PhD
Senior Scientist
INSERM Bioinformatics-MTI University Paris-Diderot

5 rue Marie-Andrée Lagroua
75205 Paris Cedex 13
E-mail: maria.miteva@univ-paris-diderot.fr

What is AMMOS_ProtLig

AMMOS_ProtLig performs energy minimization of protein-ligand interactions and can be applied on a huge number of protein-ligand complexes pre-generated with an user-chosen docking programs [4-6]. The molecular mechanics minimization of AMMOS_ProtLig is based on two force fields of AMMP: *sp4* and *sp5* [7]. The entire procedure of AMMOS_ProtLig, from the input of the protein and pre-docked ligands databank to the final databank of the minimized protein-ligand complexes is shown in the following scheme:

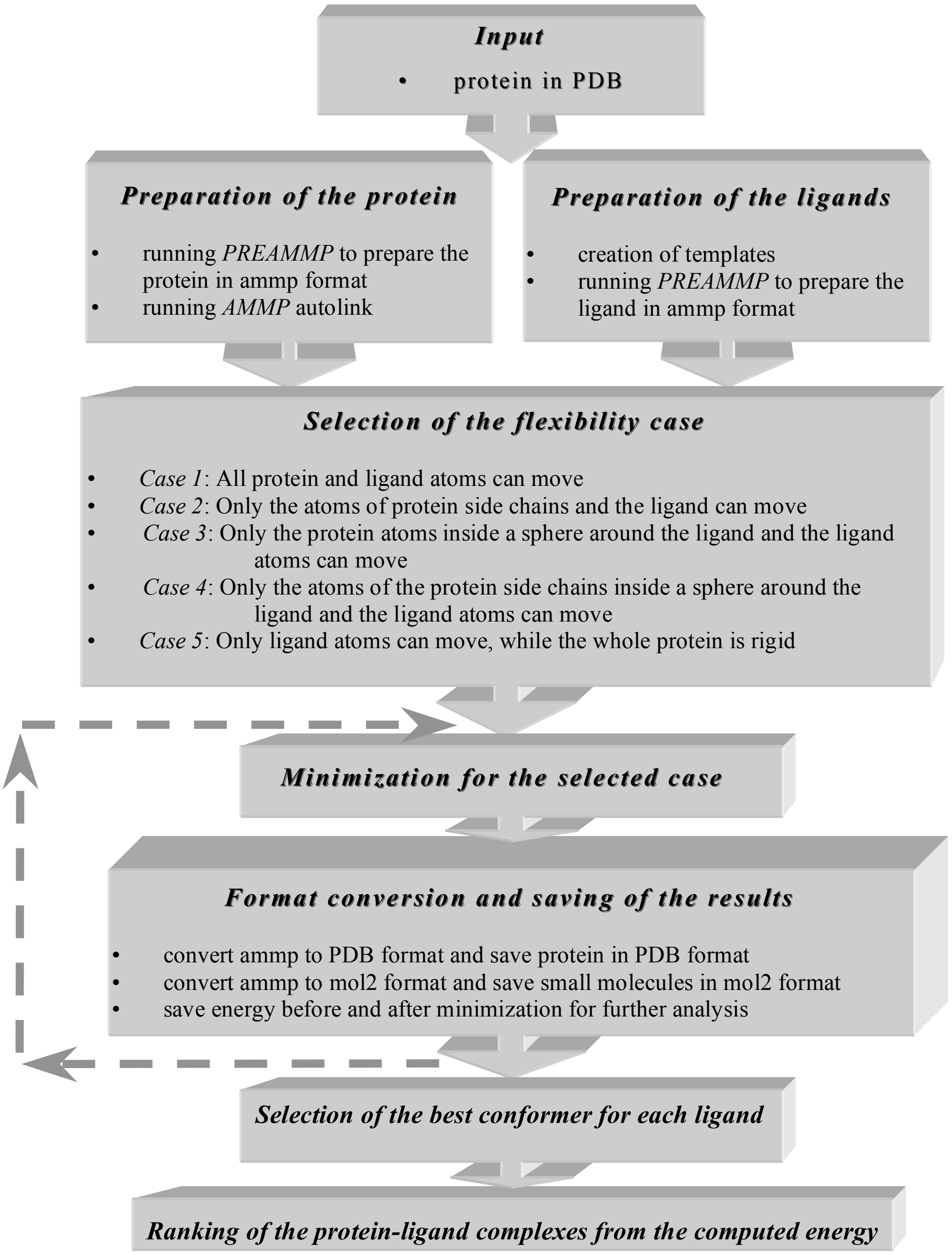
Input
• protein in PDB
Preparation of the protein
• running PREAMMP to prepare the protein in ammp format
• running AMMP autolink
Preparation of the ligands
• creation of templates
• running PREAMMP to prepare the ligand in ammp format
Selection of the flexibility case
• Case 1: All protein and ligand atoms can move
• Case 2: Only the atoms of protein side chains and the ligand can move
• Case 3: Only the protein atoms inside a sphere around the ligand and the ligand atoms can move
• Case 4: Only the atoms of the protein side chains inside a sphere around the ligand and the ligand atoms can move
• Case 5: Only ligand atoms can move, while the whole protein is rigid
Minimization for the selected case
Format conversion and saving of the results
• convert ammp to PDB format and save protein in PDB format
• convert ammp to mol2 format and save small molecules in mol2 format
• save energy before and after minimization for further analysis
Selection of the best conformer for each ligand
Ranking of the protein-ligand complexes from the computed energy

Figure 1. Schematic flow of the AMMOS algorithm;
the arrows show the cycle for the automated procedure

Back to the top

How to install AMMOS_ProtLig

Step 1: Uncompress the package in a Linux shell as follows:

```
> tar -zxvf  AMMOS_ProtLig.tar.gz
```

The directory *~AMMOS_ProtLig* contains the subdirectories:

- "*doc/*": user manual
- "*install/*"
- "*bin/*": executable binary files
- "*progs/*":
  - "*ammp/*": the program AMMP
  - "*preammp/*": the program PREAMMP
  - "*Python_scripts/*": the Python scripts of AMMOS_ProtLig
  - "*vls_min/*": the C programs source and input files for the AMMOS_ProtLig energy minimization protocols
- "*example/*": an example for running AMMOS_ProtLig

Step 2: Compile the source codes

In the *~AMMOS_ProtLig/install* subdirectory run the compiling script "*Makefile*" by typing:

```
> make
```

(warning you need Python 2.4 or above)

The script will compile the source codes automatically and generate executable files for the programs AMMP, PREAMMP, and AMMOS_ProtLig. This script will automatically install all executable files into the directory *~AMMOS_ProtLig/bin/.*

Step 3:

User should edit the "*.cshrc*" (or "*.bashrc*") file to add the path of *~AMMOS_ProtLig/bin* in *PATH* environment variable. For example, in c-shell:

```
> set path=($path ~/AMMOS_ProtLig/bin)
```

WARNING: this present version should run on both, Linux and Mac versions (eg Mac OS X, 10.4 and 10.5, PPC and Intel). On Mac 10.5 for example, assuming you have the developer tools & shell is bash. You need to generate a .bashrc file and add something like this if you want to install AMMOS only for your account:

```
PATH=$PATH:"/Users/bruno/Desktop/data.dir/AMMOS_onVLS3D_last_Nov7/AMMOS_ProtLig/bin:"

export PATH
```

but then when you start a new terminal or login again…this file .bashrc is eventually not read by the system. So you also have to create a: .bash_profile file, and there you can add: source ~/.bashrc

Back to the top

How to run AMMOS_ProtLig

In the working directory where AMMOS_ProtLig computations will be ran, the following files should be present: the protein in PDB format, the compound collection containing the pre-docked ligands in mol2 format and the *input parameter* file (see for example the input.param file in the *~AMMOS_ProtLig/example* directory). After the minimization, the results will be saved in the same formats. The user should edit this file to give the correct paths and chosen parameters:

The input parameter file:
path_of_*AMMOS_ProtLig* = *path of the package AMMOS_ProtLig*[1]
protein= *name of the PDB protein file*
bank= *name of the pre-docked ligands databank in mol2*
case_choice= *the case of allowed flexible protein atoms: 1, 2, 3, 4 or 5* (see for more details below)
radius= *the radius (in Å) of a sphere around the ligand with allowed flexible receptor atoms inside*[2]

Notes:
[1] The total number of symbols of the path_of_*AMMOS_ProtLig* should not exceed 80.
[2] This parameter will be used only for the *case 3* and *case 4*, where partial flexibility of the protein in the sphere with specified size is ensured.

To run AMMOS_ProtLig for energy minimization of protein-ligand complexes and to rank them according to their interaction energy:

```
> AMMOS_ProtLig_sp4.py input_parameter_file
```

You can employ *sp4* or *sp5* force field.

AMMOS_ProtLig Procedure:
The entire automatic procedure for energy minimization of protein-ligand complexes with AMMOS_ProtLig, from the input files (the protein and the pre-docked ligand collection) to the output (the final databank of the minimized protein-ligand complexes), is shown in figure 1. The automatic procedure for many small molecules is accomplished via a Python script. The preparation of the protein and the small molecules in the specific ammp format is carried out with the program PREAMMP. The corresponding templates for proteins are available in the directory *~AMMOS_ProtLig/progs/preammp/pdb* (*the correspondence between the templates and the protein atom names as well as for the N-terminal residue should be ensured by the user before running AMMOS_ProtLig. In this version the pdb templates correspond to a pdb file with hydrogens added by Chimera-1.3, Warning, with some other versions of Chimera, the naming of hydrogens can be different, also due to changes in the PDB file format, please check this point carefully*). The ammp script *prepare_protein.ammp* automatically builds all incomplete residues in the protein targets with the program AMMP. Then, the procedure *mol2_to_templ_sp4 (or sp5) (available in ~AMMOS_ProtLig/progs/vls_min)* creates a PDB and template files for the small ligands, based on the initial mol2 file. Some warning could appear if unknown atom types or bonds

are present. User should select the desired "case" allowing or not movements of the receptor atoms in the *input parameter file*. The preparation of active/inactive (moving/rigid) atoms for the protein is ensured by the procedure *active_case**. Five different cases have been elaborated for choice of active/inactive atoms in the protein, while ligands are always active. The "*" corresponds to the case number among the following possibilities:

- *Case 1*: All protein and ligand atoms can move
- *Case 2*: Only the atoms of protein side chains and ligand can move
- *Case 3*: Only the protein atoms inside a sphere around the ligand and the ligand atoms can move
- *Case 4*: Only the atoms of protein side chains inside a sphere around the ligand and the ligand atoms can move
- *Case 5*: Only ligand atoms can move, while the whole protein is rigid

The input files required by AMMP for the minimization procedures are *min_case*.ammp* (available in *~AMMOS_ProtLig/progs/vls_min*). They allow for the selection of the optimization method (in the presented version *Conjugate gradient*), and the number of iteration steps (in this version 2x500). Experienced users can select other optimization method available in AMMP, as well as number of iteration steps.

The scripts *ammp_to_mol2* and *ammp_to_pdb_case** ensure the conversion from the ammp format to the initial mol2 or PDB format for the ligands and the protein, respectively, and save the new coordinates after the minimization. Finally *save_energy* saves all internal, external and total energies before and after the minimization procedure.

After the minimization stage, AMMOS_ProtLig ensures:

1) keeping the new coordinates of the ligands after minimization in mol2 format.
2) keeping the new coordinates of the protein active atoms after minimization in PDB format.
3) keeping the external, internal and total energies (external energy refers to the interaction energy between the protein and the ligand, the internal energy is the energy of the ligand and total energy is a sum of internal and external terms).
4) re-ranking of all minimized protein-ligand complexes (in a single conformer or multiple conformer ligands if available) according to calculated AMMP interaction energy.
5) keeping any warning that may appear during the AMMOS_ProtLig run.

Back to the top

Example

An example of energy minimization of protein-ligand complexes with AMMOS_ProtLig showing the procedure is available in the directory *~AMMOS_ProtLig/example*. The input protein receptor in PDB format, the initial pre-docked ligands in mol2 format, and the input parameter file (input.param) (see explanations above) are given. Users can simply run:

- *AMMOS_ProtLig_sp4.py input.param_multiconf*
- *or*

➢ *AMMOS_ProtLig_sp4.py input.param_singleconf*

At the end of the minimization, final results can be found in the directory *~AMMOS_ProtLig/example/ligands_databank_multiconf_case4_OUTPUT*:

- *ligand_databank_multiconf_case4_minimized.mol2* contains the coordinates of all minimized bound ligand conformers (see explanations above)
- *ligand_databank_multiconf_case4_output.pdb* contains the coordinates of the protein after the minimization (see explanations above)
- *ligand_databank_multiconf_case4_energy.txt* contains the external, internal and total energy of the protein-ligand complexes before and after the minimization (see explanations above)
- *ligand_databank_multiconf_case4_total_warnings.txt* contains any warning that can appear during the minimization run (see explanations above)
- *ligand_databank_multiconf_case4_rankedenergy_singleconf.txt* contains a list of ranked ligands according to the best external energy conformers (see explanations above)

Back to the top

References

1. Pencheva T., D. Lagorce, I. Pajeva, B.O. Villoutreix, M.A. Miteva. AMMOS: Automated Molecular Mechanics Optimization tool for *in silico* Screening.
2. Harrison R., C. Reed, I. Weber. Analysis of Comparative Modeling Predictions for CASP2 Targets 1, 3, 9, and 17. *Proteins: Structure, Function, and Genetics*, 1997, Suppl. 1, 68-73
3. Weber I., R. Harrison. Molecular Mechanics Calculations on Protein–Ligand Complexes. *Perspectives in Drug Discovery and Design*, 1998, 9/10/11**,** 115-127
4. Sperandio O., M.A. Miteva, F. Delfaud, B.O. Villoutreix. Receptor-Based Computational Screening of Compound Databases: The Main Docking-Scoring Engine. *Current Protein and Peptide Science*, 2006, 7, 369-393
5. Miteva M.A., W.H. Lee, M.O. Montes, B.O. Villoutreix. Fast Structure-Based Virtual Ligand Screening Combining FRED, DOCK, and Surflex. *Journal of Medicinal Chemistry,* 2005, 48*,* 6012-6022
6. Sauton N., D. Lagorce, B.O. Villoutreix, M.A. Miteva. MS-DOCK: Accurate multiple conformation generator and rigid docking protocol for multi-step virtual ligand screening. *BMC Bioinformatics*, 2008, 9:184
7. Bagossi P., G. Zahuczky, J. Tözsér, I. Weber, R. Harrison. Improved Parameters for Generating Partial Charges: Correlation with Observed Dipole Moments, *Journal of Molecular Modeling*, 1999, 5, 143-152

Back to the top