FORECASTER Suite 2016

User Guide

Department of Chemistry,
McGill University
Montréal, Québec, Canada

Molecular Forecaster Inc.
Laval, Québec, Canada

11/06/2016

Table of Contents

Preface

I.1. Conventions used in this guide

The FITTED docking tool has initially been developed as a suite of three programs: SMART (used to prepare the small molecules for docking), PROCESS (used to prepare the protein files for docking) and the docking program FITTED. More recently, these three programs together with several others have been integrated into a single package, namely the FORECASTER Platform.

This guide describes the use of a suite of programs which are usable either from a graphical user interface or via command-line arguments. FORECASTER requires a set of commands to be issued in the form of a *keyword file*, a standard ASCII text file with instructions. Each instruction is given by **`Keyword`** *`Option`*. Although one instruction (i.e., keyword) is often on a single line, some keywords might span multiple lines.

In the remainder of the manual, different typefaces will be used to symbolize the following:

- Filenames and command-line input: constant-width font, standard face.
  Examples: `ligand.mol2`
  `keyword.txt`
  `Forecaster keyword_smart.txt`
- Keyword names: constant-width font, bold face.
  Examples: **`Protein`**
  `Mode`
  `AutoFind_Site`
- Keyword options: constant-width font, italic face.
  Examples: *`1a46.mol2`*
  `Docking`
  `Yes`

Please note that the formatting is for clarity of the manual only as it is not possible to format an ASCII file with different typefaces.

Also note that some keywords in this guide might not be up to date as they are now automatically written using the java GUI (front-end.jar).

I.2. Acknowledgements

Over the last years, the development of FITTED, IMPACTS, ACE and all the other tools of FORECASTER has been funded by ViroChem Pharma (research grants), the Canadian Institutes for Health Research (CIHR Operating grants), AstraZeneca and the Natural Science and Engineering Research Council (NSERC discovery grant). In addition, the “Ministère du Développement Economique, de l'Innovation et de l'Exportation du Québec" has recognized the potential of our drug discovery platform by funding the development and the commercialization as part of a program called "Soutien à la maturation technologique".

I.3. The team of developers:

This software would have not been available without the contribution of outstanding people.

Nicolas Moitessier - group leader (2003-present)

Pablo Englebienne	(2003-2009)
Christopher R. Corbeil	(2005-2009)
Jeremy Schwartzentruber	(2007)
Eric Therrien	(2008-present)
Nathanael Weill	(2010-2011)
Valerie Campagna-Slater	(2010-2011)
Andy Arrowsmith	(2009-2011)
Joshua Pottel	(2011-2015)
Zhaomin Liu	(2011-present)
Anna Tomberg	(2013-present)
Matthieu Moitessier	(2013-2014)

II. Before using the FORECASTER Suite

II.1. Recent changes from previous versions

The FORECASTER Suite 2016 now includes all the program from the FORECASTER Platform. The java-based graphical interface (Front-end) was further developed to integrate all the tools in intuitive automated workflows. A 3D visualizer and a 2D sketcher were also integrated. In addition, new programs were added and other programs were updated and improved.

The FORECASTER Suite is available for three different platforms, Linux, Windows and Mac OSX. The package contains one executable, Forecaster.exe (Forecaster in Linux and Mac OSX). It also contains a force field folder with the fitted_ff.txt force field file and a machine_id.exe program for generating the license file. Finally, a Java-based graphical user interface (front-end) for easier file manipulation and program execution with 3D visualizer and 2D sketcher utilities.

II.2. Installation

To install the suite of programs, simply follow the instructions given below. This procedure will install the programs and all the required files in a system folder (**the path should not include white spaces**). The programs can still be used as a command line with arguments or using the graphical interface. Make sure to install the correct version of the suite that corresponds to your system architecture (32- or 64-bits).

II.2.1. Windows

To install the program on Windows simply unzip the file to the root of the hard drive (ex: c:\). You can install it anywhere else except that the path to the **executables should not contain white space**. However, the path where you run the calculations (i.e. the "working directory") can contain white spaces. You can also create a shortcut for the Front-end.jar gui (see below for instruction how to use the gui).

II.2.2. Linux

To install the Linux version, open a terminal window and execute the installation script with the following command (tcsh or bash):

```
Fitted@Linux:~$ ./install_forecasterXX.bin     (where XX = 32 or 64)
```

The script will guide you through the installation process. The programs can be installed locally (user account) or in a system folder (must be root to run the script, do not use “sudo”). In order to be able to run the program from the command line, you must edit your bashrc file to include the PATH to FITTED.

To be added to the bashrc file:

```
export FITTED=”your-installation-path/FITTED/”

export PATH=”your-installation-path/FITTED/:\$PATH”
```

The programs can then be executed from any directory by simply typing the name of the program (see section III.3.2) and the gui can be launched by typing “front-end” from a terminal window.

II.2.3. Mac OS X

To install the MAC OSX version, open a terminal window and execute the install script with the following command:

```
mac$ ./install_forecasterX.bin
```

The script will guide you through the installation process. The programs can be installed locally (user account) or in a system folder (must be root to run the script, do not use "sudo"). In order to be able to run the program from the command line, you need to provide the full (absolute) path to the executable.

The programs can then be executed from any directory by typing the **full path** to the program (see section III.3.2) and the gui can be launched by typing the following command from a terminal window:

```
mac$ java –jar <full_path_to_the_executable>/Front-end.jar
```

II.3. Minimum Requirements

Windows:

Windows XP, Windows Vista, Windows 7, Windows 8 (32-bit and 64-bit architecture)
2 GB of RAM (4GB or more recommended)
Java 1.6 (latest version) for gui

Linux:

Ubuntu 8.10, CentOS 5.2 (32-bit and 64-bit architecture) and higher versions
Xterm needs to be installed
2 GB of RAM (4GB or more recommended)
Java 1.6 (latest version) for gui

Mac OS X:

Leopard 10.6 (64-bit architecture only) and higher versions
2GB of RAM (4 GB or more recommended)
Java 1.6 (latest version) for gui

II.4. License File

The execution of the programs is controlled by the license file (`license.fitted`). This license ensures that the programs are used on the licensed computers only. Therefore, you first have to generate a `machine_id.fitted` file by using the `machine_id` program. This file should then be sent by email to license@fitted.ca. Using this file, our team will generate the necessary `license.fitted` file and send it back to you. A copy of this `license.fitted` file must be located in the same folder as the executables.

II.4.1. Generating a machine_id.fitted file.

Windows: Double click on the program `machine_id.exe` and a file named `machine_id.fitted` will be created.

Linux and Mac OS X: In a terminal, navigate to the installation folder and execute the `machine_id` program by typing:

```
<path_to_the_executable>/machine_id
```

Email this file to license@fitted.ca to obtain your `license.fitted` file. Repeat this process on each computer you need to run the programs.

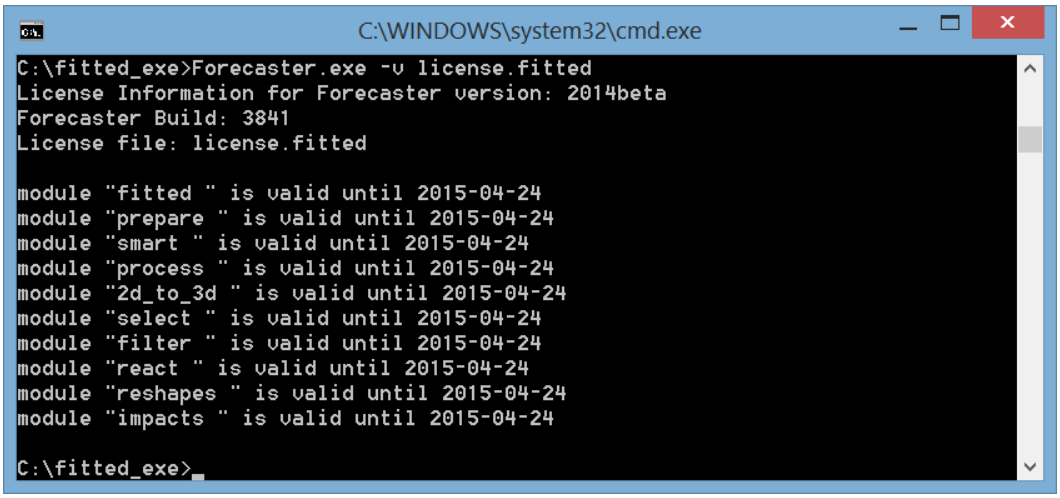
II.4.2. License and version tools.

Windows: In a dos window, navigate to the folder where the executables are installed and execute the following command to get information about the license status and programs version.

```
c:\fitted_exe2>Forecaster.exe -v license.fitted
```

Linux and Mac OS X: In a terminal, navigate to the installation folder and execute the following command get information about the license status and programs version:

```
<path_to_the_executable>/Forecaster -v license.fitted
```



III. Getting started with FORECASTER

III.1. Running the FORECASTER Suite

The suite works under Windows, Mac OSX and Linux through the graphical user interface. The Linux and Mac OSX versions are also useable from a terminal window as command line (CLI).

III.1.1. Running the FORECASTER Suite from the graphical user interface

Please refer to the tutorials, which are available from the fitted.ca website.

III.1.2. Running the FORECASTER suite from the command line

To run the program, place the required files in your working directory and create an appropriate keyword file (see Section IV). You can run the program by typing the following command in the terminal (where `<path_to_the_executable>` is your installation folder).

```
<path_to_the_executable>/Forecaster keyword_file.txt
```

If running more than one file sequentially as in virtual screening runs, scripts can be used to create keyword files, extract data and run FITTED. Examples of these scripts are available on the fitted.ca website or upon request.

IV. Keywords for PREPARE and MATCH-UP

The following section lists the keywords, their functions and default values. Angle brackets <> indicate a numeric value; `plain text` indicates a text string (such as a file name); square brackets `[]` indicate a choice of values, the default shown in *`italics`*. When a default value is assigned to a keyword, the latter can be omitted from the keyword file.

PREPARE keyword files are case-sensitive. Empty lines are allowed, and text after a pound sign (#) is considered a comment.

Although the value of many keywords can be altered, default values should be used unless a specific system requires different settings.

At the end of this section, typical keyword files can be found.

IV.1. Input/output files

`Main_Mode` `prepare_protein`

- Following the keyword, specify the main mode to be run.

`Run_Mode` `mode`

- Following the keyword, specify the run mode which can be.
- **make_similar**: superposes multiple PDBs then make them similar.
- **make_mol2**: converts a PDB file to a mol2 file.
- **make_mol2_flexible**: converts a PDB file to multiple side-chain conformations mol2 files.
- **alignment**: provides the sequence alignment of multiple PDB files.
- **superpose**: superposes multiple PDB structures.

`Protein` `<#_proteins>`

```
protein_file1.pdb
protein_file2.pdb
```

- On the same line following this keyword, specify the number of protein files to be processed.
- On subsequent lines, the protein filenames, pdb files only (1 line per file) with no comments
- Next to the pdb filename, the chain ID is optional. Default is All. It can be any combination as desired, such as A, AB, ABC, AD, or All (for everything).

When `Run_Mode` is `make_similar` or `aligment`:

`Protein` `<#_proteins>`

```
protein_file1.pdb <chain_to_be_considered>
protein_file2.pdb <chain_to_be_considered>
```

- On the same line following this keyword, specify the number of proteins.
- On subsequent lines, the protein filenames, pdb files only.
- Next to the pdb filename, either the chain ID must be listed or All must be given. It can be any combination as desired, such as A, AB, ABC, AD, or All (for everything).

`Output` `output_filename`

- Name of the output file.
- output_filename_pro.mol2, output_filename_lig.mol2 and output_filename.out will be created.

Ligand_Include <#_ligands>

Ligand_name_1 chain number

Ligand_name_2 chain number

- Manually defines the ligand.
- On the same line following this keyword, specify the number of ligand residues.
- On subsequent lines, the residue name, chain and numbers are specified one per line as it appears in pdb (ex: TMC B 500) with no comments (even with pound sign).
- The number of ligand residues refers to the residues that form the molecule as it appears in the pdb file. **ONLY one ligand molecule is allowed.**

IV.2. Parameters for the preparation of the protein (all modes)

Protein_Include <#_protein_residues>

Protein_name_1 chain number

Protein_name_2 chain number

- Residue to be included in the protein mol2 file.
- On the same line following this keyword, specify the number of protein residues.
- On subsequent lines, the residue name, chain and numbers are specified one per line as it appears in pdb (ex: PTR A 201).
- Can be used for protein residues that are not recognized automatically by the program as natural amino-acid residues.

Ligand_Exclude <#_ligand_residues>

Ligand_name_1 chain number

Ligand_name_2 chain number

- Ligand residue to be excluded from the protein mol2 file.
- On the same line following this keyword, specify the number of ligand residues.
- On subsequent lines, the residue name, chain and number are specified one per line as it appears in the pdb file (ex: TMC A 500).

Mutate <residue_name> <res_chain> <res_number> <new_res>

- Residue to be automatically mutated to another residue.
- On the same line following this keyword, specify the residue name, chain, number as it appears in the pdb file followed by the new residue type. (ex: TYR A 58 PHE).

Delete <residue_name> <res_chain> <res_number>

- Residue to be automatically deleted.
- On the same line following this keyword, specify the residue name, chain, number as it appears in the pdb file. (ex: ASP A 19).

IV.3. Additional parameters for the preparation of a mol2 file (mode make_mol2 only)

Mode [*fitted*|normal]

- Mode of execution. In the fitted mode, only a maximum of **20 water molecules** within 5 Å of the ligand are conserved in the protein mol2 file. In the normal mode, no water molecule deletion is performed.
- The default is `fitted`.

Optimize [Y|*N*]

- Optimization of tautomers and water molecules.
- The default is `Y`.

Iterations <number>

- Number if optimization iterations.

  The default is `10`.

Particle_Water <Yes|No> <number>

- Adds water molecules as single particle water to the pdb structure.
- On the same line, Yes should be followed by the number of particle waters to be added (10 is suggested).
- Default is No.

Protonate <atom_to_protonate>

- Atom to be manually protonated by the program. If PREPARE does not assign the correct protonation state, the user can override PREPARE to force a given protonation state using this keyword.
- On the same line following this keyword, specify the residue name, chain, number and atom name as it appears in the pdb file. (ex: HIS A 58 NE2).

Deprotonate <atom_to_deprotonate>

- Atom to be manually deprotonated by the program. If PREPARE does not assign the correct protonation state, the user can override PREPARE to force a given protonation state using this keyword
- On the same line following this keyword, specify the residue name, chain, number and atom name as it appears in the pdb file. (ex: RTL A 701 O2).

Hybridization <atom_to_hybridize>

- Atom to manually change the hybridization by the program. If PREPARE does not assign the correct hybridization state, the user can override PREPARE to force a given hybridization state using this keyword
- On the same line following this keyword, specify the residue name, chain, number, atom name, and hybridization state as it appears in the pdb file. (ex: RTL A 701 C15 sp2).

Flexibility <Yes|No>

- New conformations will be generated for flexible side-chains within the active site.
- Only one pdb file should be used with the Protein keyword.
- Default is No.
- Other keywords may be used (otherwise default values will be used): Max_Num_of_Flex, Num_Of_Confs and Num_Of_Mutations.

Max_Num_of_Flex <number>

- Maximum number of flexible side-chains to be considered when the Flexibility keyword is used. (Not all will be considered simultaneously (see keyword below). The *number* most flexible according to Najmanovich, et al. (*Proteins: Structure, Function and Genetics* **2000**, *39*, 261) will be considered.
- Default is 5

Num_Of_Mutations <number>

- The number of simultaneously flexible side chain to be produced per protein conformation when the Flexibility keyword is used.
- Default is 2

Num_Of_Confs <number>

- The number of new protein conformations to be generated when the Flexibility keyword is used.
- Default is 5

IV.4. A simple PREPARE keyword file for make_mol2 mode

```
Protein          1
                 1e2k.pdb

Output           1e2k

Ligand_Include   1
                 TMC A 500

Optimize         y
Iterations       5

Run_Mode         make_mol2
Main_Mode        prepare_protein
```

IV.5. A simple MATCH-UP keyword file for make_similar mode

```
Protein          2
                 1e2k.pdb A
                 1e2p.pdb A

Output           tk

Ligand_Include   2
                 TMC A 500
                 CCV A 500

Run_Mode         make_similar
Main_Mode        prepare_protein
```

IV.6. An advanced PREPARE keyword file for make_mol2_flexible mode with automated generation of new protein conformations

```
 Protein               1
                       1E2K.pdb

 Output                1e2k_flexible
 Mode                  fitted

 Ligand_Include        1
                       TMC A 500

 Optimize              yes
 Iterations            1

 Flexibility           Yes
 Max_Num_of_Flex       5
 Num_Of_Confs          5
 Num_Of_Mutations      2

 Main_Mode             prepare_protein
 Run_Mode              make_mol2_flexible
```

V. Keywords for PROCESS

The following section lists the keywords, their functions and default values. Angle brackets <> indicate a numeric value; `plain text` indicates a text string (such as a file name); square brackets [] indicate a choice of values, the default shown in *italics*. When a default value is assigned to a keyword, the latter can be omitted from the keywordfile.

PROCESS keywords files are case-sensitive. Empty lines are allowed, and text after a pound sign (#) is considered a comment.

Although the value of many keywords can be altered, default values should be used unless a specific system requires different settings.

At the end of this section, a typical keyword file can be found.

V.1. Input/output files

Main_Mode process

- Following the keyword, specify the main mode to be run process in the present case.

Run_Mode process

- Following the keyword, specify the run mode process in the present case.

Protein <#_protein_struct>
protein_file1.mol2
protein_file2.mol2

- Following the keyword, specify the number of protein structure files to be processed
- On the following lines, specify the protein file names, one per line with no comment (even with a pound sign)

Output output_filename

- Name of the output file.

Binding_Site_Cav BindSite_filename

- Name of the file where to output the binding site cavity.
- If this keyword is not present PROCESS will not create a binding site cavity file.

Interaction_Sites IS_filename

- Name of the file where to output the interaction sites definition file.
- If this keyword is not present PROCESS will not create an interaction sites definition file.

Ligand <#_ligands>
ligand1.mol2
ligand2.mol2

- Ligand file(s) (in MOL2 format) used to define the active site and its center. It should be in the same frame as the protein.

Constraints xxxx_pro_cons

- Name of the file where to output the constraints (without the mol2 extension).
- Must be used with the keyword `Constraint_With_Residues`.

Constraint_With_Residues <#_constraints>
Residue_name_1 chain atoms
Residue_name_2 chain atoms

- Manually defines the protein residues to build a constraint from.
- Atoms can be: backbone, side-chain or any specific atom.
- Example for specific atom: GLU123 A NH1
- Example for backbone: GLU123 A backbone
- Example for side-chain: GLU123 A side-chain

V.2. Reading the input files and preparing the output protein files

AutoFind_Site [*Y*|N]

- This function allows the user to have PROCESS automatically finding the flexible residues/binding site.
- The default is `Y`.

Ligand_Cutoff <ligand_cutoff>

- Protein residues within this cutoff (in Å) are considered part of the binding site.
- The default is 6.0.

Binding_Site <#_flex_residues>
flex_residue_1_name
flex_residue_2_name

- Manually defines the active site. (The active site can be automatically defined by providing a ligand, see above)
- On the same line following this keyword, specify the number of flexible residues. **This list should be as exhaustive as possible to avoid missing any important residue defining the active site**.
- On subsequent lines, the residue name/numbers (according to **Find_Residues**) are specified, one per line.

Truncate [Y|N|*auto*]

- Determine if the protein will be truncated, keeping only residues within **Cutoff** of the binding site residues.
- The default is *auto*.
- The protein will be truncated keeping residues within cutoff distance of the ligand and not within cutoff distance from the binding site residues.

Cutoff_Truncate <cutoff>

- Any residue that does not have an atom within this distance (in Å) from an atom of a flexible residue or of the given ligand will be deleted from the protein file that PROCESS will output.
- The default value is 9.

Find_Residues [*Name*|Number]

- If **Active_Site** is used, define in which way PROCESS will identify the residues that make up the binding site.

  Name

  - Search residues by group name.
  - This is the default.

  Number

  - Search residues by group number.

V.3. Parameters for the binding cavity file

Grid_Center <grid_center>

- Specifically defines the center of the binding site
- The default is to automatically find it using the center of a ligand.

Grid_Size <size x> <size y> <size z>

- Specifies the size of the box for the binding site.
- The default is 15 15 15.

Grid_Boundary [*Soft*|Hard]

Soft

- When converting from the grid to spheres, the boundary of the box will be ignored (defined by **Grid_Size**) and spheres can include volume outside of the box.
- This is the default.

Hard

- The active site cavity file will be constrained within the box defined by **Grid_Size**.

Grid_Resolution <grid_resolution>

- Following this keyword is the resolution (Å) of the grid.
- The default is 1.5.

Grid_Sphere_Size <grid_sphere_size>

- Specifies the size of a sphere used to trim the sides of the box to make it rounder.
- The default 15.

Grid_Clash <grid_clash>

- If a protein atom is within this distance of a grid point, the point is removed from the grid.
- The default is 1.5.

V.4. Parameters for the Interaction Sites file

XXX_Weight <xxx_weight>

- This group of keywords (xxx being Hydrophobic, Metal, HBA or HBD) specifies the parameters for the assignment of pharmacophoric points. xxx_weight is used to give weight for favorable xxx-type interactions. Defaults parameters are highly recommended.

  Hydrophobic_Weight <hydro_weight>

  - Defines the weight for hydrophobic interaction points.
  - The default is 1.

Metal_Weight <metal_weight>

- Defines the weight for metal interaction points.
- The default is 50.

HBA_Weight <hba_weight>

- Defines the weight for hydrogen bond acceptor interaction points.
- The default is 5.

HBD_Weight <hbd_weight> <hbd_penalty>

- Defines the weight for hydrogen bond donor interaction points.
- The default is 5.

If too many points are found, one can reduce this number by using the following keywords:

Pharm_Polar_Softness <pharm_polar_soft>

- Maximum distance (in Å) between two polar points to merge.
- The default is 0.0.

Pharm_Nonpolar_Softness <pharm_nonpolar_soft>

- Maximum distance (in Å) between two non-polar points to merge.
- The default is 0.0.

Hydrophobic_Level <hydro_level>

- Van der Waals interaction between a probe on the grid point with hydrophobic carbons to be considered hydrophobic. If the interaction is found lower than hydro_level, an hydrophobic point is added at this location.
- The default is -0.3.

Min_Weight <min_weight>

- Minimum weight for a pharmacophoric point to be included in the final pharmacophore.
- The default value is 0.5.

Num_of_IS <num_of_spheres>

- This determines the maximum number of interaction site spheres in the interaction sites file.
- The default is 75.

V.5. A simple PROCESS keyword file for rigid protein docking setup

```
Protein                 1
                        1e2k_pro.mol2

Output                  1e2k

Binding_Site_Cav        1e2k_bindSite
Interaction_Sites       1e2k_IS

AutoFind_Site           yes
AutoFind_Center         yes

Ligand                  1
                        1e2k_lig.mol2

Run_Mode                process
Main_Mode               process
```

V.6. A simple PROCESS keyword file for flexible protein docking setup

```
Protein                 2
                        1e2k_pro.mol2
                        1e2p_pro.mol2

Output                  tk-process

Binding_Site_Cav        tk_bindSite
Interaction_Sites       tk_IS

AutoFind_Site           yes
AutoFind_Center         yes

Ligand                  2
                        1e2k_lig.mol2
                        1e2p_lig.mol2

Run_Mode                process
Main_Mode               process
```

V.7. An advanced PROCESS keyword file

```
Protein                 1
                        protein.mol2
Output                  protein
Binding_Site_Cav        BindSite.mol2
Interaction_Sites       IS.mol2
AutoFind_Site           yes
Ligand                  1
                        lig.mol2
Ligand_Cutoff           9

Truncate                auto
Cutoff                  7
Num_of_IS               50

Run_Mode                process
Main_Mode               process
```

VI. Keywords for SMART

SMART is a module used to prepare ligand structures in a modified MOL2 format for use by FITTED. It can also assign atomic partial charges and prepare ligand structures for use with ACE (Asymmetric Catalyst Evaluation) or add descriptors to be used with REDUCE.

The following section lists the keywords, their functions and default values. Angle brackets <> indicate a numeric value; `plain text` indicates a text string (such as a file name); square brackets `[]` indicate a choice of values, the default shown in *`italics`*. When a default value is assigned to a keyword, the latter can be omitted from the keywordfile.

SMART keywords files are case-sensitive. Empty lines are allowed, and text after a pound sign (#) is considered a comment.

Although the value of many keywords can be altered, default values should be used unless a specific system requires different settings.

At the end of this section, a typical keyword file can be found.

VI.1. Input/output files

❖ *please notice the '-' (dash) before some keywords*

`Main_Mode` `smart`

- Following the keyword, specify the main mode to be run.

`Run_Mode` `smart`

- Following the keyword, specify the run mode for the program specified in the Main_Mode keyword.

`Molecule` `XXXX_lig.mol2`

- Name of the ligand file.
- Supported file formats are mol2 and 3D sdf.
- Files can contain either single or multiple molecules.

`Output` `output_filename`

- Name of the output file. Should be **different** that the input filename.
- If not specified, SMART will automatically append "**_1**" to the filename.

VI.2. Parameters for the preparation of the ligand file

`-mode` `[`*`fitted`*`|filter|ace|metabolism]`

- Instructs SMART to write the file in selected format.
- The default is *`fitted`*.

`-inf` `[mol2|sd|sdf|fitted|amber]`

- File format of the input ligand.
- If not specified, SMART will automatically detect the file format from the input file extension.

`-outf` `[`*`mol2`*`|std|debug]`

- File format of the output ligand.
- The default value is `mol2`.

-multi [*Y*|N]

- SMART will output a multi mol2 file.
- The default value is Y.

-split [*number*]

- SMART will output multi mol2 files each containing the number of molecules as specified.
- This is used for splitting a multiple ligands file into separate files for docking in parallel.

-charge [*MMFF*|DGH|none|input]

- SMART will assign the atomic partial charges based on the selected method.
- none will zero all the partial charges and input will keep the charges as they appear in the input mol2 file.
- The default value is MMFF.

-assign_bond [*Y*|N]

- SMART will assign the bond orders.
- The default value is N.

-name Field_ID

- Field containing the name of the molecule to be used in the sdf file.
- Usually, this field contains brackets that should be included (ex: <Corporate_ID>).

VI.3. A simple typical SMART keyword file

```
Molecule          1e2k_lig.mol2
Output            1e2k_lig_1

Run_Mode          smart
Main_Mode         smart
```

VI.4. An advanced typical SMART keyword file

```
Molecule          1e2k_lig.mol2
Output            1e2k_lig_1

-mode             fitted
-inf              mol2
-outf             fitted
-charge           DGH
-assign_bond      yes
-multi            y
-ionize           y
-name             <corporate_id>

Run_Mode          smart
Main_Mode         smart
```

VII. Keywords for CONVERT

The following section lists the keywords, their functions and default values. Angle brackets <> indicate a numeric value; `plain text` indicates a text string (such as a file name); square brackets `[]` indicate a choice of values, the default shown in *`italics`*. When a default value is assigned to a keyword, the latter can be omitted from the keyword file.

CONVERT keyword files are case-sensitive. Empty lines are allowed, and text after a pound sign (#) is considered a comment.

Although the value of many keywords can be altered, default values should be used unless a specific system requires different settings.

At the end of this section, typical keyword files can be found.

VII.1. Input/output files

Main_Mode `convert`

- Following the keyword, specify the main mode to be run process in the present case.

Run_Mode `2d_to_3d`

- Following the keyword, specify the run mode process in the present case.
- When using the reverse conversion, `2d_to_3d` should be used to convert a 3D molecule to 2D sketch.

Molecule `input_filename`

- Name of the input file.

Output `output_filename`

- Name of the output file.

VII.2. Parameters for the conversion of the molecules

Name `mol_ID`

- Following the keyword, specify the name of the field in SDF file which corresponds to the name of the molecules.
- Ex: <Cat_No>

Tautomers `[Yes|No]`

- The program will generate other possible tautomers for each ligand if any.
- The default is `Yes`.

Racemic `[Yes|No]`

- The program will make all chiral molecules racemic.
- The default is `No`.

VIII. Keywords for FITTED

The following sections list the common keywords (those are most frequently changed, for a complete list for a specific usage, please contact us.), their functions and default values. Angle brackets <> indicate a numeric value; `plain text` indicates a text string (such as a file name); square brackets `[`*`choice1`*`|choice2]` indicate a choice of values, the default shown in *`italics`*. When a default value is assigned to a keyword, the latter can be omitted from the keyword file.

Note that keyword files are case-sensitive. Empty lines are allowed, and text after a pound sign (#) is considered a comment.

Although the value of many keywords can be altered, default values should be used unless a specific system requires different settings. These keywords are essentially used by the developers for optimization and evaluation of the program. In general, modification of a specific value does not significantly improve or affect the accuracy but may result in longer or quicker docking runs.

At the end of this section, a typical keyword file can be found.

VIII.1. Input/output files

Protein <# of files>

input_file_1

input_file_2

- Following this keyword is the number of protein structure files used as input (same protein different conformation). These protein files should be prepared using PROCESS prior to the actual docking.
- On the following lines are the protein file names, one per line, **without the file extension** (.mol2).

Ligand ligand_file.mol2

- Name of the ligand file to be docked (in MOL2 format). This ligand files should be prepared using SMART prior to the actual docking.
- The ligand file can contain a single molecule or multiple molecules (multi-mol2).

Ref <#_of_files>

lig_ref_file1.mol2

lig_ref_file2.mol2

- Following this keyword is an integer stating how many reference files are used to calculate the **root-mean-square deviation (RMSD)** of the ligand heavy atoms. These ligand files should be in the same reference frame as the protein structure. The possible symmetric conformations of the ligand are calculated *in silico*.
- RMSD calculation can only be done when the ligand's bioactive conformation is known (e.g. self-docking study).
- 2 reference files may be needed in some instances where the ligand or protein active site is C_n symmetric (n >=2)
- On the following line(s), the reference file(s) (in MOL2 format) are listed, one per line.
- If this keyword is missing, no RMSD values will be computed.

Output filename

- Name of the output file.

Binding_Site_Cav XXXX_BindSite.mol2

- Following this keyword is the file defining the empty space present in the active site cavity (a set of spheres prepared by PROCESS).
- If this keyword is missing, no binding site clash filter will be used (it is highly recommended to use both **Interaction_Sites** and **Binding_site_cav** keywords).

Interaction_Sites XXXX_IS.mol2

- Name of the file containing the interaction site description (prepared by PROCESS).
- If this keyword is missing, no interaction site filter will be used. (It is highly recommended to use both **Interaction_Sites** and **Binding_site_cav**)

Pharmacophore pharmacophore_file.mol2

- Name of the file containing the pharmacophore constraints on the ligands (prepared by PROCESS). Typically this keyword is used to ensure that the individuals produced match this constraint, but it can be softened by setting **Min_Constraint**.
- If this keyword is missing, no constraint will be used.

Protein_Ref <#_of_files>

ref_file_1.ext

ref_file_2.ext

- Following this keyword is the number of reference protein structure files used to compute the protein RMSD (deviation of the modeled protein structure from the reference structures).
- On the following lines are the protein file names, one per line. These files will be used in addition to the Protein files listed before to calculate a root-mean-square-deviation (RMSD) between the protein generated during a fitted docking run and the Protein_ref files. Additional files can be needed if the protein has a symmetrical structure (e.g., HIV-1 protease)
- If this keyword is missing, protein input files will be used as references.

VIII.2. Run parameters

Mode [*Dock*|Filter|VS|Score|Local]

Dock

- Normal docking run.
- This is the default.

VS

- Faster docking mode for virtual screen. Slightly less accurate.

Local

- Performs a local search on the ligand input structure. The provided orientation/translation/conformation is used as a starting point and only slight modifications to the ligand conformation, orientation and translation are carried out.

SAR

- Performs a local search on the ligand input structure. The provided orientation/translation/conformation is used as a starting point and only slight modification to the ligand orientation and translation are carried out while a complete search of conformations is done.

Flex_Type [Rigid|Semiflex|Flex_water|Flex]

rigid

- The ligand is docked onto one protein structure.
- This is the default if only one protein structure is used.

`semiflex`

- The ligand is docked onto multiple protein structures (requires **`Protein`** ≥ *2*). Proteins can be exchanged during the evolution but not the genes corresponding to side chains or water molecules (a more complete description of this mode is given in reference 1).
- This is the default if more than one protein structure is used.

`flex_water`

- The ligand is docked into multiple protein structures (requires **`Protein`** ≥ *2*). Similar to `Semiflex`, except that each water molecule evolves independently.

`flex`

- The ligand is docked onto multiple protein structures (requires **`Protein`** ≥ *2*). The side chains and waters are allowed to be exchanged independently from the protein backbone.

`Number_of_Runs` `<number of runs>`

- More than one run per ligand can be performed (The ligand may be docked several times to ensure a complete search).
- If this keyword is missing, the default value is 3 for Dock mode. For all other modes the default is 1.

`Number_of_Runs` `<number of runs>`

`Displaceable_Waters` `[`*`On`*`|Off}`

- Allows the user to turn off the displaceable waters (They will be kept explicit)
- The default is `on` which allows displaceable waters.

`Particle_Waters` `<Yes|No>`

- Instructs the program to use particle waters (Need to be previously added by PREPARE).
- Default is No.

`Corner_Flap` `[On|`*`Off`*`]`

- Turns the corner flap conformational search for rings on or off.
- By default, it is set to `Off`.

VIII.3. Energy parameters

`Score_Initial` `[`*`none`*`|score|minimize]`

- Scoring of the initial ligand binding mode.

`none`

- No scoring of the initial input structure is performed.
- This is the default setting.

`score`

- Only the score of the initial input ligand is output.

`minimize`

- The score of the initial pose and the score of the energy minimized structure will be output.

`VdWScale_1-4` `<vdwscale_1-4>`

- Scaling factor for the 1,4 van der Waals interactions.

- The default is `1.0`.

VdWScale_1-5 `<vdwscale_1-5>`

- Scaling factor for the 1,5+ van der Waals interactions.
- The default is `1.0`.

E_VdWScale_Pro `<e_vdwscale_pro>`

- Scaling factor for the ligand-protein van der Waals interactions.
- The default is `1.0`.

E_VdWScale_Wat `<e_vdwscale_wat>`

- Scaling factor for the ligand-water van der Waals interactions.
- The default is set the value as the same as **E_vdWScale_Pro**.

ElecScale_1-4 `<elecscale_1-4>`

- Scaling factor for the 1,4 electrostatic interactions.
- The default is `1.0`.

ElecScale_1-5 `<elecscale_1-5>`

- Scaling factor for the 1,5+ electrostatic interactions.
- The default is `1.0`.

E_ElecScale_Pro `<e_elecscale_pro>`

- Scaling factor for the ligand-protein electrostatic interactions.
- The default is `1.0`.

E_ElecScale_Wat `<e_elecscale_wat>`

- Scaling factor for the ligand-water electrostatic interactions.
- The default value is set the same as **E_ElecScale_Pro**.

E_HbondScale_Pro `<e_hbondscale_pro>`

- Scaling factor for the ligand-protein hydrogen bond interactions.
- The default is `1.0`.

E_HbondScale_Wat `<e_hbondscale_wat>`

- Scaling factor for the ligand-water hydrogen bond interactions.
- The default value is set the same as **E_HbondScale_Pro**.

Cutdist `<cutdist>`

- Cutoff distance (in Å) for the non-bond interactions with the protein.
- The default value is `9`.

Switchdist `<switchdist>`

- Switching distance (in Å) for the non-bond interactions with the protein.
- The default value is `7`.

Cutdist_Wat `<cutdist_wat>`

- Cutoff distance for the non-bond interactions with the water molecules.
- The default value is `1.20`

Switchdist_Wat <switchdist_wat>

- Switching distance for the non-bond interactions with the water molecules.
- The default is 1.75.

Solvation [*On*|Off}

- Allows the user to turn off the calculation of the solvation energy
- The default is on.

VIII.4. Scoring parameters

- The default values for all the keywords are highly recommended as they represent the scaling factors optimized for RankScore. Please contact us if you need to change one of these keywords.

VIII.5. Initial population parameters

Pop_Size <pop_size>

- Population size for the genetic algorithm conformational search.
- When 10000 is given as value, automatic determination based on the ligand's number of torsions is done.
- The default is automatic for rigid docking, 200 for flexible docking when keyword is omitted.

Min_MatchScore <min_matchscore>

- This keyword is used only if an interaction site file is provided. If the **Mode** is set to Dock, Min_MatchScore is automatically calculated.
- Minimum match of the interaction sites.
- The default is 20.

Min_PharmScore <min_constraint>

- This keyword is used only if a pharmacophore file is provided to guide the docking.
- Minimum percent match of the pharmacophore.
- The default is 100.

Anchor_Atom <anchor_atom>

- Sequence number of the atom to be used as an anchor. This is used to identify the center of translation and rotation for the GA.
- If this keyword is not specified, the anchor is automatically set to the gravity center of the ligand.

Anchor_Coor <anchor_x> <anchor_y> <anchor_z>

- Following this keyword must be the x, y and z coordinates of the protein active site center.
- If this keyword is not used, it is automatically set to the center of the protein active site defined by the active site (flexible) residues.

Matching_Algorithm [*On*|Off]

- Turns on or off the matching algorithm.
- By default, it is set to On.

Num_of_Top_IS <num_of_top_IS>

- Number of top Interactions sites that the interaction site triangles must contain at least one of.

- The default is `10`.

`Stringent_Triangles` `<weight_of_triangles>`

- Is a factor by which the triangles are selected. The higher **`Stringent_Triangles`** is set, the more the matching algorithm will favour triangles that have not been used.
- The default value is `5`.

`Stringent_MS` `<stringent_MS>`

- Is a weight factor used in calculation of **`Min_MatchScore`**. The higher this value, the stricter **`Min_MatchScore`** becomes.
- The default value is `4`.

VIII.6. Evolution parameters

`Max_Gen` `<max_gen>`

- Determine the maximum number of generations for the genetic algorithm.
- The default is `175`.

`CutScore_1` `<cutscore_1>`

- Upper bound score at **`Max_Gen`** to further proceed with the docking run. If there is one individual within the top 3 below this CutScore_1 then the program proceeds to Max_Gen_1
- The default is `-5`.

`CutScore_2` `<cutscore_2>`

- Upper bound score at **`Max_Gen_2`** to further proceed with the docking run. If there is one individual within the top 3 below this CutScore_2 then the program proceeds to Max_Gen_2
- The default is `-7.5`.

`Max_Gen_2` `<max_gen_2>`

- As for **`Max_Gen_1`**, if after **`Max_Gen_1`** generations none of the top poses has a score below the one specified by **`CutScore_2`**, the program exits. Otherwise, the program proceeds until it reaches **`Max_Gen_2`**.
- The default is `Max_Gen`.

`Seed` `<seed>`

- Select the starting point within the random number generator. If the same run is done with the same seed (on the same computer), the exact same result will be obtained. If a different seed is used, the GA will follow a different path. Changing the seed helps the developers to evaluate the convergence of a run.
- The default is `100`.

`pLearn` `<plearn>`

- Probability of energy minimization of the parents at every generation.
- The Default is `0.1`.

`pCross` `<pcross>`

- Probability of crossover at every generation.
- The default is `0.85`.

`pMut` `<pmut>`

- Probability of mutation at every generation.
- The default is `0.05`.

pMutRot `<pmutrot>`

- Probability of mutation of the orientation of the ligand at every generation.
- The default is `0.30`.

pMutWat `<pmutwat>`

- The maximum rate of mutation of the water at ***Max_Gen*** generations
- The default is `0.35`.

pElite `<pElite>`

- The percentage of the best of the population to be directly passed on to the next generations.
- The default is `0.01`.

pElite_Every_X_Gen `<pElite_Every_X_Gen>`

- ***pElite*** will be used every **pElite_Every_X_Gen** generation(s).
- The default is `2`.

pElite_SSize `<pElite_SSize>`

- The individual to be passed directly onto the next generation will be selected random from the top **pElite_SSize** individuals of the population.
- The default is `10`.

pOpt `<popt>`

- Probability of optimization of the ligand at every generation.
- The default is `0.20`.

Evolution `[Steady_State|Metropolis|Elite]`

`Steady_State`

- During the evolution, out of a pair of two children and their 2 parents the two best will be saved.
- This is the default.

`Metropolis`

- During the evolution, out of a pair of two children and their 2 parents two individuals will be saved following the Metropolis criterion. If the children are higher in energy they are checked to see if they have a high probability to exist at room temperature. If they do they are saved.

`Elite`

- During the evolution, the top ***pop_size*** individuals of the children and parents will be kept in the next generation.

GA_Num_of_Trials `<ga_num_trials>`

- Maximum number of successive unsuccessful trials to create children.
- The default is `1000`.

Diff_Avg_Best `<difference_avg_best>`

- The absolute difference between the average energy of the population and the best individual of the population. If the calculated value is below `difference_avg_best` then the population is considered to be converged.
- The default is `1`.

Diff_N_Best <difference_n_best>

- The absolute difference in energy between the individual with the lowest energy and the individual ranked **Diff_Number.**
- If **Diff_Number** is defined the default value is 0.5.

Diff_Number <number_rank>

- The number of the indivuals to be used with **Diff_N_Best**
- By default this criteria is not used.

VIII.7. Docking of covalent inhibitors

Covalent_Residue <residue_name>

- Following this keyword is the name of the residue, the covalent inhibitor will react with. Only CYS and SER are implemented in the current version (e.g., SER554)

Covalent_Ligand [*Only*|Both]

- Controls the covalent docking. FITTED will automatically identify the aldehyde, boronate or nitrile groups (other groups will eventually be implemented) and assign the proper atom types when covalent poses will be considered

  Only
  - Only covalent poses will be considered
  - This is the default.

  Both
  - Covalent and non-covalent poses will be considered concomitantly.

Proton_Moved_To <residue> <atom_name>

- The proton of a catalytic residue (e.g., serine hydroxyl group) will be moved to atom *<atom_name>* of residue *<residue>* (e.g., a neighboring histidine residue).

VIII.8. Output/convergence parameters

Print_Level [0|*1*|2|3]

- Controls the amount of data output.
- The default value is 1.

Print_Structures [*Final*|Full|None]

- Controls the output of the structures during or at the end of the docking.

  Final
  - Only the final structures will be printed.
  - This is the default.

  Full
  - The structures (protein and ligand) will be printed during the run along with the final structures.

  None
  - No structures will be printed.

All_Poses <yes|no>

- Tells the program to output only the best scoring pose (no) or all the poses (yes) for each ligand.

- If this keyword is missing, the default value is no.

Print_Num_Structures <print_num_structures>

- Select how many of the top poses are printed as MOL2 files.
- The default is 1.

Number_of_Best <number_of_best>

- Select how many individuals to print the score, energy and RMSD during the run.
- The default is 10 in **Mode** Dock and 1 in **Mode** VS.

Print_Best_Every_X_Gen <print_best_every_x_gen>

- How often to print a summary of the run.
- The default is (Max_Gen + 1).

Print_Energy_Full [*Yes*|No]

- Controls the printout of the detailed energy contributions.

  Yes
  - Print out a breakdown of the energy (bond energy, angle energy, etc.).
  - This is the default.

  No
  - Print out only the total energy.

VIII.9. A simple FITTED keyword file for rigid protein docking

```
Protein                     1
                            1e2k_pro

Ligand                      1e2k_lig_1.mol2

Output                      1e2k
Forcefield                  fitted_ff.txt
Parameters                  Auto
Ref 1
                            1e2k_lig_1.mol2
Binding_Site_Cav            1e2k_bindSite.mol2
Interaction_Sites           1e2k_IS.mol2

Mode                        Dock
Flex_Type                   Rigid
```

VIII.10. A simple FITTED keyword file for flexible protein docking

```
Protein               4
                      1e2k_aligned_mutated_pro
                      1e2p_aligned_mutated_pro
                      1ki3_aligned_mutated_pro
                      2ki5_aligned_mutated_pro


Ligand                1e2k_aligned_mutated_lig_1.mol2

Output                tk-flex
Parameters            Auto
```

```
       Ref 1
                             1e2k_aligned_mutated_lig_1.mol2

       Binding_Site_Cav      tk_bindSite.mol2
       Interaction_Sites     tk_IS.mol2

       Mode                  Dock
       Flex_Type             Flex
```

VIII.11. A template FITTED keyword file with all the possible keywords

```
########
#
# This template file contains all the keywords in use by FITTED. For a detailed
# explanation of their use please see FITTED user guide.
#
########
#
# INPUT/OUTPUT FILES
#
#################################################################################################
Protein               <# of files>          # Number of protein input files
                      input_file_1
                      input_file_2

Protein_Ref           <# of files>          # Number of prot files used for RMSD
                      input_file_1          # First ref protein file
                      input_file_2          # Second ref protein file

Ligand                ligand_file.mol2      # Ligand structure file

Ref                   <#_of_files>          # Number of reference ligand files
                      lig_ref_file1.mol2
                      lig_ref_file2.mol2

Output                filename              # Name of the output file
Forcefield            fitted_ff.txt         # Force field file name

Binding_Site_Cav      bindSite.mol2         # Name of cavity file created by ProCESS
Interaction_Sites     IS.mol2               # Name of interaction file created
                                            # by ProCESS

Pharmacophore         pharmacophore_file.mol2  # Name of Pharmacophore file

#
# Run parameters
#
#################################################################################################
Mode                  Dock                  # [Local|VS|SAR|Filter] Running mode

Number_of_Runs        3                     # Number of runs to carry out. If using any other mode
                                            # than Dock, the default is 1

Flex_Type             Rigid                 # Type of docking to be performed. If more than one
                                            # protein is used, the default is set to Semiflexible
Displaceable_Waters   On                    # [On|Off] Toggle displaceable waters
Corner_Flap           on                    # [On|Off] Toggle ring conformational search


#
# Conjugate gradient parameters
#
#################################################################################################
#GI_Max_Iter          40                    # Maximum number of iters during init pop gen
#GI_StepSize          0.02                  # Initial step size along direction
#GI_MaxStep           1.0                   # Maximum Step size
```

```
#GI_MaxGrad            0.001                 # Gradient convergence criteria
#GI_EnergyBound        0.001                 # If energy change after GI_MaxSameEnergy
#GI_MaxSameEnergy      3              # iters is < GI_EnergyBound, consider equivalent

#GA_Max_Iter           40             # Maximum number of iterations in evolution
#GA_StepSize           0.02           # Initial step size along direction
#GA_MaxStep            1.0            # Maximum Step size
#GA_MaxGrad            0.001          # Gradient convergence criteria
#GA_EnergyBound        0.001          # If energy change after GA_MaxSameEnergy
#GA_MaxSameEnergy      3              # iters is < GA_EnergyBound, consider equivalent

#
# Energy parameters
#
#############################################################################################
#Score_Initial         none           # [none|score|minimize] Scoring initial input
#VdWScale_1-4          1.0            # Scaling factor for 1,4 vdW interactions
#VdWScale_1-5          1.0            # Scaling factor for 1,5 vdW interactions
#E_VdWScale_Pro        1.0            # Scaling factor for lig-prot vdW energy
#E_VdWScale_Wat        1.0            # Scaling factor for lig-wat vdW energy
#ElecScale_1-4         1.0            # Scaling factor for 1,4 elec energy
#ElecScale_1-5         1.0            # Scaling factor for 1,5 elec energy
#E_ElecScale_Pro       1.0            # Scaling factor for lig-prot elec energy
#E_ElecScale_Wat       1.0            # Scaling factor for lig-wat elec energy
#E_HbondScale_Pro      1.0            # Scaling factor for lig-prot Hbond energy
#E_HbondScale_Wat      1.0            # Scaling factor for lig-wat Hbond energy
#Cutdist               9              # Cutoff dist (in A) for lig-prot non-bond
#Switchdist            7              # Switching dist (in A) for lig-prot non-bond

#Cutdist_Wat           1.20           # Cutoff dist for lig-wat non-bond
#Switchdist_Wat        1.75           # Switching dist for lig-wat non-bond

#GI_Protein_Nbonds     United         # [United|All-atom] Prot repr for init pop gen
#GA_Protein_Nbonds     United         # [United|All-atom] Prot repr for evolution

#GA_Protein_Nbonds2    <Max_Gen2>     # Gen to switch from United to All-atom
#Solvation             On             # [On|Off} calculation of the solvation energy

#
# GENETIC ALGORITHM PARAMETERS
#
#############################################################################################
#
# Initial population parameters
#
#############################################################################################
Pop_Size               100            # Number of individuals in the population
#Min_MatchScore        25             # Initial Min_MatchScore
#Min_PharmScore        100            # Minimum value for PharmScore

#Anchor_Atom   <anchor_atom>          # Number of atom to be used as ctre of rot

#Anchor_Coor   <anchor_x> <anchor_y> <anchor_z> # x, y and z coord of BS ctre

#Max_Tx                5.0            # Max value (in A) for translation in x
#Max_Ty                5.0            # Max value (in A) for translation in y
#Max_Tz                5.0            # Max value (in A) for translation in z

#GI_Num_of_Trials      10000          # Max number of successive unsuccessful trials

#
# MATCHING ALGORITHM
#
#############################################################################################
#Matching_Algorithm    On             # [On|Off] Toggle matching algorithm
#Num_of_Top_IS         10             # Number of top IS points that interaction site
                                      # triangle must contain at least one of
#Stringent_Triangles   5.0            # Factor by which triangles are selected.
                                      # The higher Stringent_Triangles is set,
                                      # the more the matching algorithm
                                      # will favour triangles that have not been used.
```

```
#Stringent_MS          4              # Weight factor used in calculation of
                                      # Min_MatchScore. The higher this value,
                                      # the stricter Min_MatchScore.
#
# Evolution
#
##############################################################################################
Max_Gen                 200            # Maximum number of generations

#Max_Gen_1              <Max_Gen_1>    # Generation number for 1st checkpoint
#CutScore_1             -4             # Upper bound score at Max_Gen_1

#Max_Gen_2              <Max_Gen_2>    # Generation number for 2nd checkpoint
#CutScore_2             -5.5           # Upper bound score at Max_Gen_2

#Seed                   100            # Random number gen seed. If 0, Seed is random

#Resolution             120            # Resolution for bond rotation during init pop gen.
                                       # For example, if a resolution of 120 is selected,
                                       # the bond rotation will occur in multiples of (360/120)
                                       # or 30 degrees.
#pLearn                 0.1            # Probability of energy minimiz of parents each gen
#pCross                 0.85           # Probability of crossover at each gen
#pMut                   0.05           # Probability of mutation at each gen
#pMutRot                0.30           # Probability of mutation of ligand orient each gen
#pMutWat                0.35           # Max rate of mutation of water at Max_Gen generations

#pElite                 0.01           # Percentage of best individuals passed to next gen
#pElite_Every_X_Gen     2              # pElite will be used every pElite_Every_X_Gen
#pElite_SSize           10             # Number of top indiv to select pElite from
#pOpt                   0.20           # Probability of optimization of children ligs each gen

#Evolution              Steady_State   # Type of evolution
#GA_Num_of_Trials       1000           # Max number of successive unsuccessful trials
                                       # to create children
###
### CONVERGENCE CRITERIA
###
#Diff_Avg_Best          1              # Min diff btw avg energy of pop'n and best indiv
#Diff_N_Best            0.5            # Min diff btw top and N-ranked indiv
#Diff_Number            Pop_Size       # N-ranked individual for Diff_N_Best
#
# Covalent docking
#
##############################################################################################
#Covalent_Residue       <residue_name> # Name of reacting prot residue
#                       SER54          # Only CYS and SER implemented so far

#Covalent_Ligand        Only           # Consider only covalent or both types

#Proton_Moved_To        <residue_name> <atom_name>    # Proton will move to
                                                     # atom <atom_name>
                                                     # of res <residue_name>
#
# Output
#
##############################################################################################

#Print_Level            1              # [1-4] Controls verbosity
#Print_Structures       Final          # Whether to output structures
#Print_Num_Structures   1              # Number of structures printed
#Print_Best_Every_X_Gen 5              # Print summary of run every X generations
#Number_of_Best         10             # Number of indivs to print summary during run
#Print_Energy_Full      no             # Output detailed energy breakdown

##############################################################################################
```

IX. Analysis of a docking run with FITTED

Once the docking run went to completion, a new folder called “output” contains several files. Each file will be explained separately in this section. The docked pose of the ligand are generated as mol2 file and can be visualized within the protein mol2 file. When docking in rigid protein mode, no protein structure is generated and the input mol2 file of the protein can be used. In flexible protein mode, structures of the protein are generated in mol2 and pdb formats.

IX.1. The log file

The log file should have the `XXXX.log` filename where *XXXX* is the value of the **output** keyword in the FITTED keyword file. This file contains any error that might occur during the docking.

IX.2. The output file

This file reports information pertaining to the docking run. This file is named `XXXX.out` based on the value *XXXX* of the **output** keyword in the FITTED keyword file. The amount of information within this file is controlled by the **Print_Level** keyword in the keyword file. A summary of the results can be found in the result file (see below).

At the beginning of the output file, all the parameters used for the docking are printed with their corresponding value. Information about the generation of the initial population appears, followed by the evolution of the population (genetic algorithm). At the end, when the convergence is reached, a table is printed with the information about the top poses of this run. When more than one run is performed (default is 3), the information is added continuously. When more than one ligand is docked within the same docking (multi-mol2 ligand file), the information about the next ligand is added in the same order as in the ligand file.

The table labeled “Best Complexes” contains the information used to identify the best pose. For each requested top pose of a single run, the ligands are ranked by energy. To this energy is then associated a score value that can be used to compare with different ligand molecules. Therefore, to identify the best pose out of the 3 run performed for the same ligand, the ligand with the lowest energy should be taken and the score associated to this ligand can then be used for comparison. The score is also based on the energy plus additional terms based on the RankScore scoring function, therefore, the lower is the score, the most likely is the pose.

```
|-------------------------------------------------|
|         Best Complexes (Ranked by Energy)       |
|-------------------------------------------------|
| Rank    | Score    | Energy   | rmsd   | mscore |
|-------------------------------------------------|
| Lig 1   |  -38.368 |  -32.865 |   0.50 | 16.179 |
| Lig 2   |  -38.201 |  -32.838 |   0.51 | 16.179 |
| Lig 3   |  -38.440 |  -32.755 |   0.50 | 16.179 |
| Lig 4   |  -38.434 |  -32.664 |   0.51 | 17.026 |
| Lig 5   |  -38.231 |  -32.657 |   0.53 | 16.179 |
|-------------------------------------------------|
```

In addition to this previous table, information about the internal energy strain of the ligand can be found as well as the on/off state of the water molecules (when displaceable waters are used).

IX.3. The results file

This file named `XXXX-results.txt` is a brief summary of the output file and contains only the minimum information about the poses (the Best Complexes table). You may refer to this file if the run went uneventfully.

IX.4. The ligand mol2 file

When the docking run is over, the best pose for each run is generated as a mol2 file that can be visualized. The name is `docking_Docked_Poses.mol2`.

IX.5. The sdf file

An sdf file is also created (`XXXX.sdf`) which contains the top pose of each run along with the associated energy, score, rmsd and mscore as sdf fields. This file can be visualized easily in any chemistry spreadsheet program that supports chemical structures or any chemical database programs.

IX.6. The protein mol2 file (flexible protein mode only)

Once a flexible protein docking run is performed, a mol2 file of the composite protein structure is generated for visualization. This file contains only the binding site (flexible residues). The name is `XXXX_Prot1_run1.mol2`.

IX.7. The protein pdb file (flexible protein mode only)

In addition to the protein mol2 file generated when flexible protein docking is performed, the complete composite protein structure is generated as a pdb file for the best pose of each run. The name is `XXXX_Prot1_run1.pdb`.