FITTED Suite 3.6 User Guide

N. Moitessier, C. Corbeil, P. Englebienne, E. Therrien

Department of Chemistry,
McGill University
Montréal, Québec, Canada

Molecular Forecaster Inc.
Laval, Québec, Canada

24/08/2012

Table of Contents

Preface

I.1. Conventions used in this guide

This guide describes the use of a suite of programs which are usable either from a graphical user interface or via a command-line arguments. FITTED and Forecaster require a set of commands to be issued in the form of a *keyword file*, a standard ASCII text file with instructions that follow the form **`Keyword`** *`Option`*, usually one per line, but in some cases a keyword 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`
  `smart 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.

I.2. Acknowledgements

Over the last years, the development of FITTED has been funded by ViroChem Pharma (research grants) and the Canadian Institutes for Health Research (CIHR Operating grants) while the development of ACE has been funded by the Natural Science and Engineering Research Council (NSERC discovery grant). These partners are warmly acknowledged. More recently, the “Ministere du Développement Economique, de l'Innovation et de l'Exportation du Québec" has recognized the potential of our drug discovery platform and granted us funding for further development and commercialization as part of a program called "Soutien à la maturation technologique". Jeremy Schwartzentruber (code optimization) and Devin Lee (comparative study) are also acknowledged

II. Before using the FITTED Suite

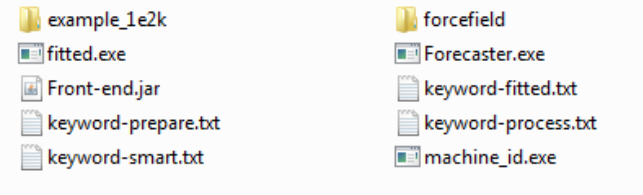
II.1. Recent changes from previous versions

The major changes that happened since the version 3.0 of the FITTED Suite are the merger of all the accessory programs in a single program called Forecaster. New actions (programs) are available either trough the FITTED or the Forecaster Suite. *The keyword files have two new keywords that are required in order to specify the mode and program to use*. Therefore, keyword files from the previous version **will not work** without the addition of these two new keywords. Automatic alignment and superposition of many proteins is now implemented in the Prepare action. This new action is essential when performing flexible protein docking with FITTED.

Since the version 2.6 (but were in the version 3.0), changes that were made are: the reorganization of the folder architecture, the usage of a keyword file for SMART action and the addition of a new action PREPARE. Other changes involve a new improved scoring function RankScore5, faster execution of FITTED, many bugs fixed in PROCESS action and a new charging scheme for atomic partial charges in SMART. A new java graphical interface is also provided in this new release. A **license file** is now required to run the programs.

II.2. What's included in the Suite

The FITTED Suite is available for three different platforms, Linux, Windows and Mac OSX. The package contains two executables, namely fitted.exe and Forecaster.exe (fitted and Forecaster in Linux and Mac OSX). It contains also a forcefield folder with the fitted_ff.txt forcefield file and a machine_id.exe program for generating the license file. Finally, a Java-based graphical user interface for easy file manipulation and program execution. Example files for protein 1e2k are given with the pre-configured keyword files.



II.3. Installation

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

II.3.1. 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 any white space**. However, the path where you run the calculations i.e. the "working directory" can contain white space. You can also create a shortcut for the Front-end.jar gui (see below for instruction how to use the gui).

II.3.2. Linux

To install the Linux version, open a terminal window and execute the install 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, don't 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 for FITTED.

To be added to the bashrc file:

```
export FITTED="your-installation-path/FITTED/"
export PATH="you-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.3.3. 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, don't 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.4. Minimum Requirements

Windows:

Windows XP, Windows Vista, Windows 7 (32-bit and 64-bit architecture)

1 GB of RAM (2GB or more recommended)

Java 1.6 (latest version) for gui

Linux:

Ubuntu 8.10, CentOS 5.2 (32-bit and 64-bit architecture)

Xterm needs to be installed

1 GB of RAM (2GB or more recommended)

Java 1.6 (latest version) for gui

Mac OS X:

Leopard 10.6 (64-bit architecture only)

1GB of RAM (2 GB or more recommended)

Java 1.6 (latest version) for gui

II.5. 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 computer only. Therefore, you need to generate a `machine_id.fitted` file by using the `machine_id` program. This file needs to be sent by email at license@fitted.ca and you will receive a `license.fitted` file. This `license.fitted` file needs to be located in the same folder as the executables.

II.5.1. 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.

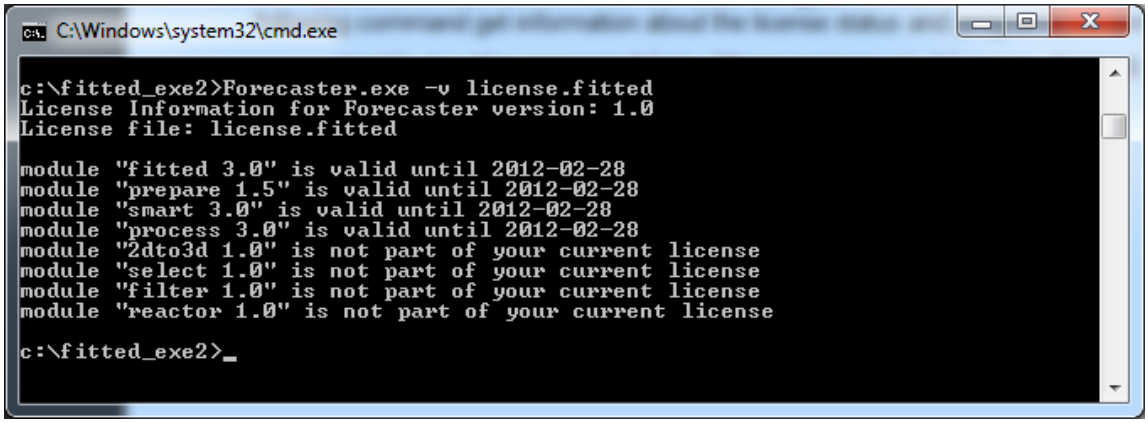
II.5.2. License and version tools.

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
c:\fitted_exe2>fitted.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
<path_to_the_executable>/fitted –v license.fitted
```


C:\Windows\system32\cmd.exe
c:\fitted_exe2>Forecaster.exe -v license.fitted
License Information for Forecaster version: 1.0
License file: license.fitted
module "fitted 3.0" is valid until 2012-02-28
module "prepare 1.5" is valid until 2012-02-28
module "smart 3.0" is valid until 2012-02-28
module "process 3.0" is valid until 2012-02-28
module "2dto3d 1.0" is not part of your current license
module "select 1.0" is not part of your current license
module "filter 1.0" is not part of your current license
module "reactor 1.0" is not part of your current license
c:\fitted_exe2>

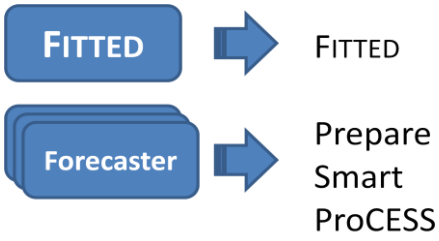
III. Getting started with FITTED

This section describes how to prepare and start a docking run with the FITTED suite of programs. All the examples and the corresponding files can be found in the example folder.

III.1. Overview of the programs of the FITTED Suite

The FITTED Suite is a collection of four different actions (programs).

1. PREPARE adds the hydrogens, mutates the truncated residues and optionally optimizes the tautomers and the water molecules according the user defined number of iterations from a pdb structure and outputs file in mol2 format. It also superposes and makes similar different pdb structures of the same protein for flexible protein docking with FITTED.
2. PROCESS creates the required files from the protein structure to be used for the docking. It requires a mol2 format with hydrogens already added.
3. SMART is used to prepare the ligand for docking. It requires a molecule in 3D mol2 format with hydrogens already added. It will add the partial charges, assign correct bond orders and correct atom types.
4. FITTED is the program used for docking, it requires a ligand that is prepared by SMART and a set of files describing the protein, as prepared by PROCESS.



III.2. Setting up the system manually

In general, the structure files downloaded from the Protein Databank (PDB) require some preparation to ensure optimal results with any docking program. For instance, the protonation state of some residues may be critical to the binding of a ligand, hence to the observed enzymatic activity. The accuracy of the docking obtained with the FITTED suite of programs therefore relies on a careful preparation of the input files. The following sections give general details on what needs to be done to the protein and ligand structure files before the FITTED suite can be used for optimal results.

When the ligand and the protein remain as one file, they will be referred to as the complex from now on. The complex may also include ions (e.g., metals), water molecules and co-factors. X-ray crystal structures downloaded from the PDB most likely do not contain hydrogen atoms. Hydrogens on the ligands are needed for FITTED since the ligand is treated via an all-atom representation while the hydrogens on the protein are required to assign advanced residue names according to the protonation states, and to compute the solvation parameters. Once hydrogens are added, their orientation should be optimized, for instance performing an energy minimization with the heavy atoms fixed.

One of the advantages of FITTED is its ability to have mobile and displaceable water molecules. However, this feature requires the proper setup of waters within the complex. Only water molecules

which are perceived as key for the binding of ligands should be kept, while all others should be removed (maximum of 20 water molecules are allowed). Waters are perceived as critical if they interact with both the protein and the ligand (bridging interactions) and are not exposed to the aqueous medium. If the number of key water molecules varies with the protein structure, copy the location of the missing waters from the other structure. During the docking run FITTED will displace them if necessary.

At this point, the complex(es) can be split into its(their) corresponding protein and ligand structure file(s). The protein file(s) should include the water molecules, ions and co-factors, if any. These files (ligand and protein) should be saved in mol2 format, available within most of the programs. If running a rigid docking (***Mode*** *Rigid*), the protein file is ready to be submitted to PROCESS. If ***Mode*** is *Semiflex*, *Flex* or *Flex_Water*, additional steps may be required to ensure that all protein files are identical (i.e., same number of protein atoms, same number of water molecules, same residue names). If discrepancies were found, PROCESS will exit with an error message (see Appendix C: PROCESS). If some crystal structures have more water molecules than others, waters can be taken/copied from the protein structures that have similar conformations. The following section lists some of the common 'errors' in PDB files which need to be corrected.

- In all cases, if the 'error' appears close to the binding site, the protein structure should not be considered for the study.
  - Mutated residues
    - if the mutation is far from the binding site (at least 10 Å from the binding site) then the residue can be virtually mutated to the desired residue.
  - Incomplete residue
    - In some case, parts of very flexible residues are not observed and are not included in PDB files. Again, if they are far from the active site, they can be virtually reconstructed.
  - Missing Residues
    - If they are far from the active site they can be (i) added where missing or (ii) removed from the other files.
  - Terminal Residues
    - In some cases, terminal residues are not properly described in the PDB or mishandled by the program used to setup the protein. (e.g., terminal COO^- groups are CHO). In this case, the missing atoms should be added.
  - Missing Waters
    - All proteins files should have the same number of waters. If a water molecule is missing, one can be virtually added from another protein file. FITTED will remove the water if it is not needed.
  - Missing atoms
    - Atom actually missing: if it is far away from the active, it can be added.
    - If the atom is there, make sure the atom name and atom type are the same.
    - The atom may be a different part of the protein file. If this is the case renumber the atom within your graphical interface and regenerate the protein input file for ProCESS.
  - Nucleic acids
    - The 5'-terminus should have a 5'-OH, and not a phosphate group. If necessary, remove the phosphate group and protonate the 5' oxygen.
    - The residue names need to be corrected before attempting to run PROCESS, but after adding hydrogens to the system. For this, a pair of scripts (`fix_dna.awk` and `fix_rna.awk`) are provided on the fitted.ca

website. This scripts rename the residues according to the names in Table 1b:

```
fix_rna.awk term5=<5'term> term3=<3'term> file.mol2 > file_new.mol2
```

<5'term> and <3'term> denote the residue numbers (column 7 in the MOL2 file) of the 5'- and 3'-terminal residues, respectively.

III.3. Setting up the system with PREPARE

We have developed a new program that automatically prepares the protein from the pdb file and creates two mol2 files: the protein and the ligand. PREPARE requires the information regarding the residue name and residue number of the ligand as it is used in the pdb file. Most common co-factor and ions are recognized by the program. Two modes are available, `normal` or `fitted`. The latter is different only by the conservation of a maximum of 25 water molecules in the protein file (all water molecule beyond 5 Å of the ligand are removed).

PREPARE adds the hydrogens, mutates the truncated residues and optionally optimizes the tautomers and the water molecules according the user defined number of iterations. All error that might occur is listed in the output file and verification should be made to identify any potential problems with the structure.

Recently, PREPARE was modified to be able to take different pdb structures of the same protein and to superpose them and make them similar based on a sequence alignment. Thus this protein ensemble can be used as input in flexible protein docking with FITTED.

III.4. Running the FITTED suite

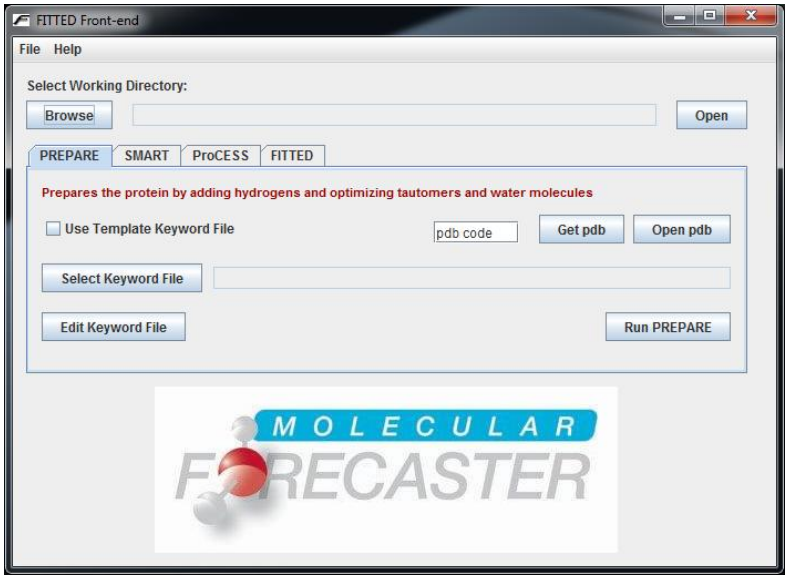
All programs work under Windows, Mac OSX and Linux. All versions are useable from a terminal window as command line or from the graphical user interface.

III.4.1. Running the FITTED suite from the graphical user interface

The graphical user interface (gui) is only a small front-end to help the setup of the docking and the edition of the keyword files. To start the gui, simply double-click on the `Front-end.jar` file (windows only). For Linux and Mac OSX users, type "java –jar Front-end.jar" from a terminal.

```
Fitted@Linux:~$ java –jar <path_to_the_executable>/Front-end.jar
```

A window with four tabs (PREPARE, SMART, PROCESS and FITTED) will open.



The first thing to do when starting to work with the gui is to define the *working directory* by browsing to the desired folder using the **Browse** button at the top of the window. Once a working directory is selected, the path will be shown in the gray box area. The working directory can be opened at any time by simply clicking the **Open** button.

The location of the executables (programs) can be changed from the **File/settings** menu.

III.4.1.1. Running PREPARE within the gui

When you start the gui, the PREPARE tab is shown by default. If you already have prepared your protein in mol2 format with the hydrogens, move to the other tabs.

If you are connected to the internet, you can download the pdb file directly by typing the 4-character code in the **pdb code** box and clicking the **Get pdb** button. Clicking the **Open pdb** button will open the downloaded pdb file using the program associated with the .pdb file extension.

If you require the use of a template keyword file, checking the option **Use Template Keyword File** will copy a template keyword file to the working directly and the path will be updated in the gray text area. If you already prepared the keyword file or you want to start from a previous keyword file, you can select the desired file by clicking the **Select Keyword File** button. IMPORTANT: the keyword file and the input files must be in the same directory. If you select a keyword file from another directory, the working directory will be automatically updated.

The keyword file can be edited by clicking the **Edit Keyword File button**. The program associated with the .txt file extension will be used to edit the keyword file. To prepare a keyword file to be used by PREPARE, refer to the appropriate section of this guide. To run the program, simply click on the **Run PREPARE** button.

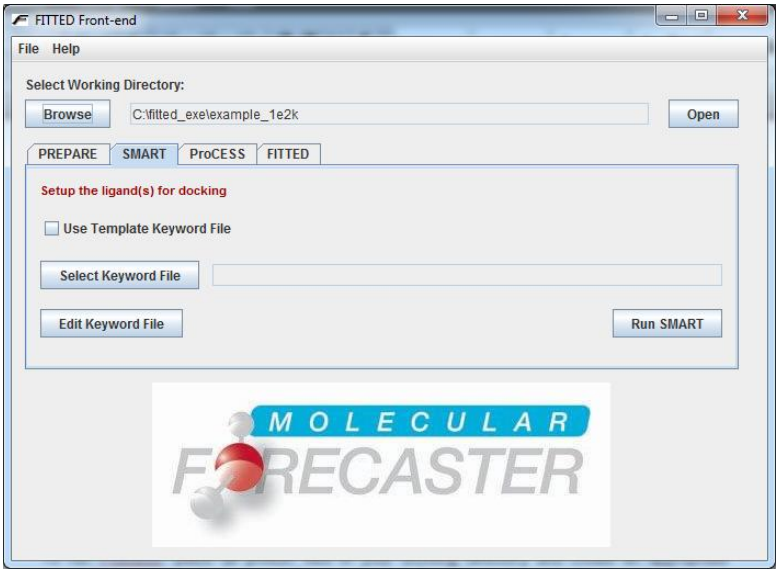
III.4.1.2. Running SMART within the gui

You can select the SMART window by clicking on the Smart tab at the top of the gui. SMART is used to setup the ligand(s) for docking with FITTED.

If you require the use of a template keyword file, checking the option **Use Template Keyword File** will copy a template keyword file to the working directly and the path will be updated in the gray text area. If you already prepared the keyword file or you want to start from a previous keyword file, you can select the desired file by clicking the **Select Keyword File** button.

IMPORTANT: the keyword file and the input files must be in the same directory. If you select a keyword file from another directory, the working directory will be automatically updated.

The keyword file can be edited by clicking the **Edit Keyword File button**. The program associated with the .txt file extension will be used to edit the keyword file. To prepare a keyword file to be used by SMART, refer to the appropriate section of this guide. To run the program, simply click on the **Run SMART** button.



III.4.1.3. Running PROCESS within the gui

You can select the PROCESS window by clicking on the PROCESS tab at the top of the gui. PROCESS is used to setup the protein(s) for docking with FITTED.

If you require the use of a template keyword file, checking the option **Use Template Keyword File** will copy a template keyword file to the working directly and the path will be updated in the gray text area. If you already prepared the keyword file or you want to start from a previous keyword file, you can select the desired file by clicking the **Select Keyword File** button.

IMPORTANT: the keyword file and the input files must be in the same directory. If you select a keyword file from another directory, the working directory will be automatically updated.

The keyword file can be edited by clicking the **Edit Keyword File button**. The program associated with the .txt file extension will be used to edit the keyword file. To prepare a keyword file to be used by PROCESS, refer to the appropriate section of this guide. To run the program, simply click on the **Run PROCESS** button.

III.4.1.4. Running FITTED within the gui

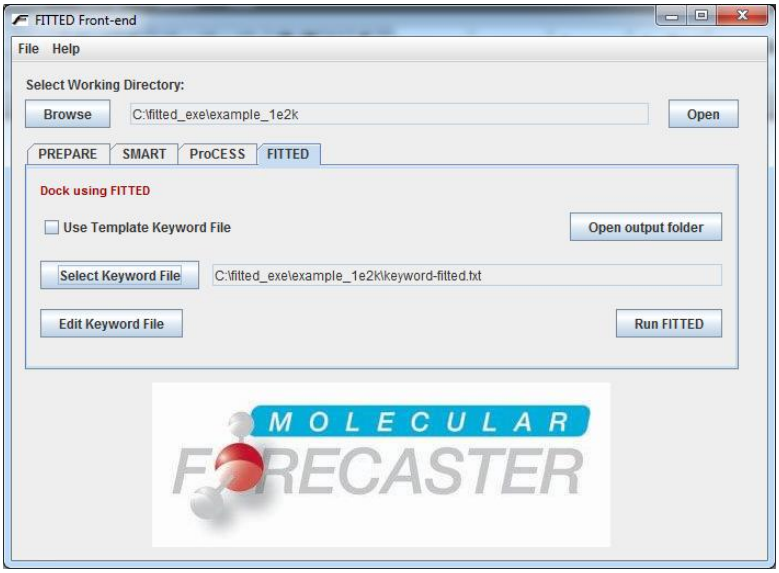
You can select the FITTED window by clicking on the FITTED tab at the top of the gui. FITTED is used dock the desired ligand(s) onto the desired protein(s).

If you require the use of a template keyword file, checking the option **Use Template Keyword File** will copy a template keyword file to the working directly and the path will be updated in the gray text area. If you already prepared the keyword file or you want to start from a previous keyword file, you can select the desired file by clicking the **Select Keyword File** button.

IMPORTANT: the keyword file and the input files must be in the same directory. If you select a keyword file from another directory, the working directory will be automatically updated.

The keyword file can be edited by clicking the **Edit Keyword File button**. The program associated with the .txt file extension will be used to edit the keyword file. To prepare a keyword file to be used by FITTED, refer to the appropriate section of this guide. To run the program, simply click on the **Run FITTED** button.

The program will output all the docking files (results and mol2 files) in the "output" folder (will be created automatically if the folder doesn't exist). This output folder can be opened by clicking on the **Open output folder** button.



III.4.2. Running the FITTED suite from the command line

To run PREPARE, place the pdb file in your working directory and create an appropriate keyword file (see example). 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_prepare.txt
```

PREPARE will create the `XXXX_pro.mol2` and `XXXX_lig.mol2` file as well as the `XXXX.out` file that contains all the information about the calculations and errors.

To run PROCESS, place all protein files in your working directory and create an appropriate keyword file (see examples keyword files provided). If flexibility is desired make sure the PROCESS

keyword file includes multiple protein structure files for consideration of flexibility. Run the program by typing:

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

PROCESS will create all the files (`XXXX_pro_dock.mol2`, `XXXX_pro_score.mol2`, `XXXX_pro_site.txt`, `XXXX_pro_IS.mol2`, `XXXX_bindSite.mol2`, `XXXX_IS.mol2`) in the working directory within minutes as well as `XXXX.out` which will include information about the calculations and errors. If running the same keyword again, all the files will be **overwritten**.

To run SMART, place the ligand file previously prepared in the working directory. Run SMART by typing:

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

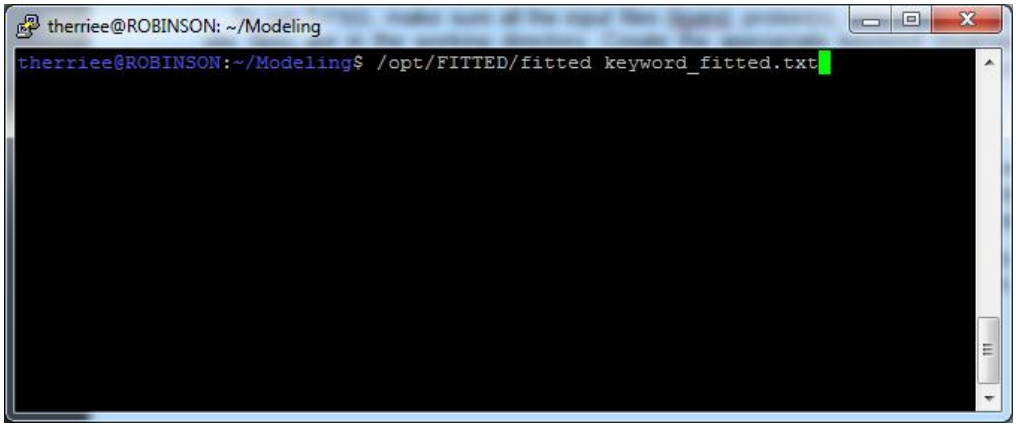
All files will be output to working directory.

To run FITTED, make sure all the input files (ligand, protein(s), active site cavity, and interaction site files) are in the working directory. Create the appropriate keyword (examples for rigid and flexible docking are included in the example folder) and place them in the working directory. To run FITTED, type:

```
<path_to_the_executable>/fitted keyword_fitted.txt
```

All results will be put into a `filename.out` file (name specified in the keyword file) and all errors/warnings in a `filename.log` file (same name as output file). If structures are output ("printed"), these files will be created in the working directory within the output directory.

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. Preparing a keyword file for PREPARE

The following section lists the keywords, their functions and default values. Gray shading indicates a required keyword; 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.

PREPARE 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, 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 for the program specified in the Main_Mode keyword.
- **make_similar**: superposes multiple PDBs then make them similar.
- **make_mol2**: converts a PDB file to a mol2 file.
- **alignment**: provides the sequence alignment of multiple PDB files.
- **superpose**: superposes multiple PDB structures.
- **stats**: collects statistics on the protein side chains.
- **nma**: reconstructs a protein structure from a complete PDB file and a backbone.

Protein <#_proteins>
protein_file1.pdb
protein_file2.pdb

- 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, 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> <chain_to_be_considered>
protein_file1.pdb A
protein_file2.pdb A

- 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, 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).

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).
- Only one ligand molecule is allowed, 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.**

Forcefield forcefield_file.txt

- Name of the force field file to use. If a forcefield other than fitted_ff.txt is to be used. The format of this force field should be consistent with the required format for Fitted.
- Default value is fitted_ff.txt if the keyword is not provided.
- It is highly non-recommended to change this value.

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 name. (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 `N`.

Iterations <number>

- Number if optimization iterations.

  The default is `10`, but `5` is also recommended.

Protonate <atom_to_protonate>

- Atom to be manually protonated by the program.
- 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.
- 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.
- 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).

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 PREPARE 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 mode

```
Protein          1
                 1e2k.pdb

Output           1e2k
Mode             fitted

Ligand_Include   1
                 TMC B 500

Protein_Include  1
                 PTR A 201

Ligand_Exclude   1
                 TMC A 500

Optimize         y
Iteration        10


Run_Mode         make_mol2
Main_Mode        prepare_protein
```

V. Preparing a keyword file for PROCESS

The following section lists the keywords, their functions and default values. Gray shading indicates a required keyword; 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.

Run_Mode process

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

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.

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.

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 ligandfile.mol2

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

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 below)
- 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 favourable 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. For more information see the section on **Error! Reference ource not found.**.
- The default is -0.3.

Min_Weight <min_weight>

- Minimum weight for a pharmacophoric point to be included in the final pharmacophore.
- The defaults are 0.5 0.0 respectively.

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                  lig.mol2
Ligand_Cutoff           9

Truncate                auto
Cutoff                  7
Num_of_IS               50

Run_Mode                process
Main_Mode               process
```

VI. Preparing a keyword file for SMART

SMART is the 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).

The following section lists the keywords, their functions and default values. Gray shading indicates a required keyword; 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`.

-ionize [*Y*|N]

- SMART will ionize thiols to sulfides.
- 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. Preparing a keyword file for FITTED

The following sections list the common keywords (one that are most frequently changed, for a complete list for a specific usage, please contact us.), their functions and default values. Gray shading indicates a required keyword; 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 keywordfile.

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.

VII.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.

Forcefield `forcefield_file.txt`

- Name of the force field file to use. If a forcefield other than fitted_ff.txt is to be used. The format of this force field should be consistent with the required format for Fitted.
- Default value is fitted_ff.txt if the keyword is not provided.

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 grid 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.

VII.2. Run parameters

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

`Dock`

- Normal docking run.
- This is the default.

`VS`

- This mode is now deprecated in this new version. When selecting this mode, it automatically switches to the Dock mode.

`Score`

- Currently not working, bug to fix! Scores the ligand input structure in the provided orientation against all input proteins.

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 time to ensure a complete search).
- If this keyword is missing, the default value is 3 for Dock mode all other modes the default is 1.

Displaceable_Waters [*On*|Off}

- Allows the user to turn off the displaceable waters.
- The default is on which allows displaceable waters.

Corner_Flap [On|*Off*]

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

VII.3. Conjugate gradient parameters

- The default values for all the keywords described in this section are recommended.

GA_* or **GI_***

- There are two sets of the following keywords: one for the parameters used during the generation of the initial population (**GI_***; e.g., **GI_MaxInt**) and another one used during the evolution (**GA_***; e.g., **GA_MaxInt**). **The default values are recommended**.

  XX_MaxIter <maxiter>

- Maximum number of iterations. Once this number is reached the minimization is finished.
- The default is `20`.

`XX_StepSize <stepsize>`

- Initial value of the step taken in the direction of the gradient during minimization.
- The default is `0.02`.

`XX_MaxStep <maxstep>`

- Maximum step size allowed during minimization.
- The default is `1`.

`XX_EnergyBound <energybound>`

- Minimum energy difference between two molecules to be considered similar.
- The default is `1.0` for **`GI_EnergyBound`** and `0.001` for **`GA_EnergyBound`**.

`XX_MaxSameEnergy <maxsameenergy>`

- Number of times that the same energy (defined by `EnergyBound`) can be repeated.
- The default is `3`.

`XX_MaxGrad <maxgrad>`

- Gradient convergence criteria.
- The default is `0.001`.

VII.4. 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.

`VdW` `[1-4|1-5]`

- Selects whether 1,4 and/or 1,5 and greater van der Waals interactions should be considered.

`1-4`

- Used to consider 1,4 interactions and above.
- This is the default setting.

`1-5`

- Used to consider only 1,5 interactions and above.

`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**.

Elec `[1-4|1-5]`

- Select whether 1,4 and/or 1,5 and greater electrostatic interactions should be considered.

  `1-4`
  - Used to consider 1,4 interactions and above.
  - This is the default setting.

  `1-5`
  - Used to consider 1,5 interactions and above.

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**.

HBond `[Y|N]`

- Selects whether or not hydrogen bonds are included in the energy calculation.
- The default is `Y`.

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 Á) for the non-bond interactions with the protein.
- The default value is `9`.

Switchdist <switchdist>

- Switching distance (in Á) 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.

GI_Protein_Nbonds [*United*|All_Atom]

- FITTED will treat protein non-bonded interactions with the ligand as either all atom or united for the generation of the initial population.
- The default for this keyword is **United**.

GA_Protein_Nbonds [*United*|All_Atom]

- FITTED will treat protein non-bonded interactions with the ligand as either all atom or united for the evolutional.
- The default for this keyword is **United**.

GA_Protein_Nbonds2 <generation number>

- FITTED will switch from united to all atom representation of the non-bonded interactions at this generation.
- The defaults is set to Max_Gen2.

Solvation [*On*|Off}

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

VII.5. Scoring parameters

- The default values for all the keywords are highly recommended as they represent the scaling factors optimized for RankScore2 and RankScore5. Please contact us if you need to change the keywords.

VII.6. 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.
- 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.

Max_Tx <max_tx>

Max_Ty <max_ty>

Max_Tz <max_tz>

- Maximum value for translation (in Å) in x, y, and z respectively.
- The default is 5 for the three values.

GI_Num_of_Trials <gi_num_trials>

- Maximum number of successive unsuccessful trials before exiting.
- The default for **Mode Dock** is 10,000 and for **Mode VS** is 1,000.

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.

VII.7. 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 `-4`.

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 `-5.5`.

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, 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**

- 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 for 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 `0.4`.

Diff_Number `<number_rank>`

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

VII.8. Docking of covalent inhibitors

This feature is under validation

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 will be moved to atom *<atom_name>* of residue *<residue>*.

VII.9. 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.

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 [*Y*|N]

- Controls the printout of the detailed energy contributions.

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

N

- Print out only the total energy.

VII.10. 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
```

VII.11. 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
```

VII.12. 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          # First protein file
```

```
#                       input_file_2          # Second protein file

#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    # First reference ligand file
                        lig_ref_file2.mol2    # Second reference ligand file

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|Score|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            off                   # [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
#VdW                    1-4             # VdW interactions to consider [1-4|1-5]
#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
#Elec                   1-4             # Electrostatic interactions to consider [1-4|1-5]
#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
#HBond                  Y               # [Y|N] Includes HBond in energy calculation
#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 numebr 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

|-------------------------------------------------------------------------------
```

VIII. Analysis of a docking run with FITTED

Once the docking run has completed, you will find a new folder called “output” which contains several files. Each file contains different information and they will be explained separately in this section. The 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, structure of the protein is generated in mol2 and pdb files.

VIII.1. The log file

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

VIII.2. The output file

This file is the most important file since it contains a lot of information pertaining to the docking. This file is named `XXXX.out` based on the value *XXXX* of the **outpout** keyword in the FITTED keyword file. The amount of information within this file is controlled by the **Print_Level** keyword in the keyword file.

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 the score, the better 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).

VIII.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).

VIII.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 `XXXX_rank_1_Lig1_run1.mol2`.

VIII.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

VIII.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_rank_1_Prot1_run1.mol2`.

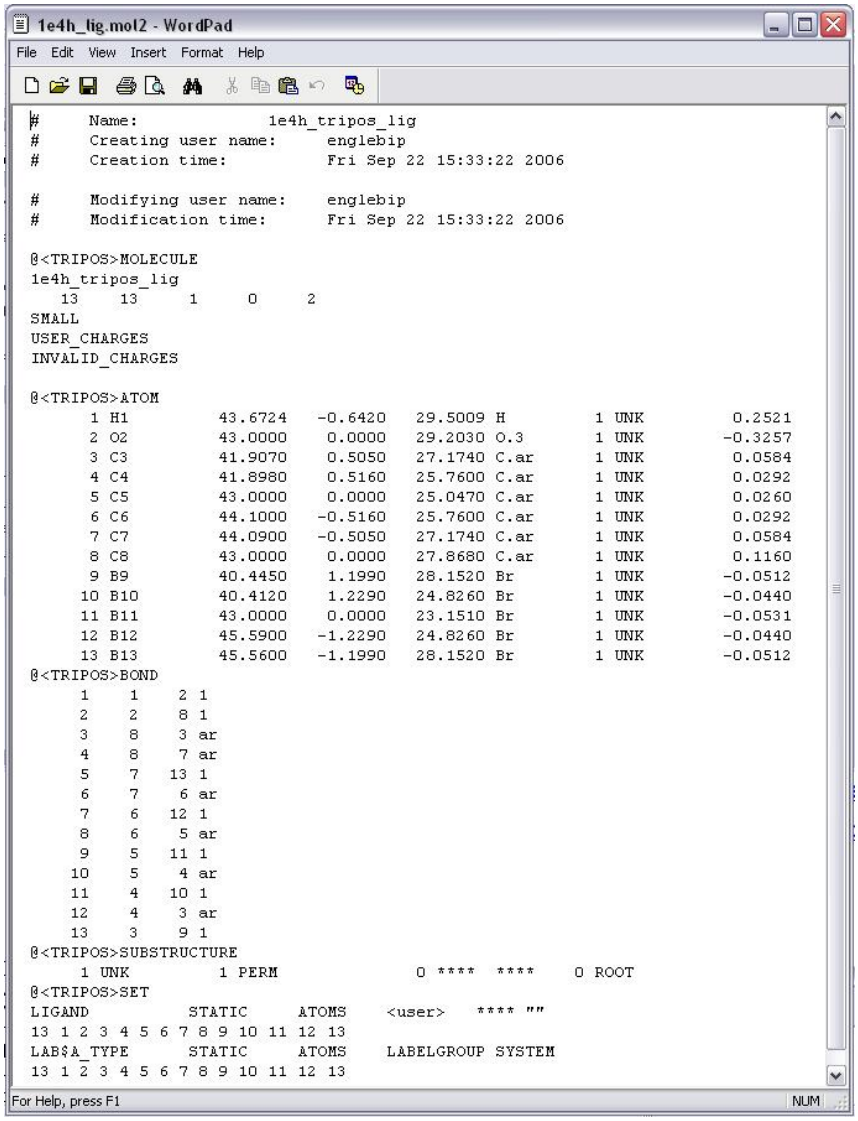
VIII.7. The protein pdb file (flexible protein mode only)

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_rank_1_Prot1_run1.pdb`.

Appendix A: FITTED Input File Formats

The following sections outline the file formats used for FITTED. FITTED uses a modified version of the Sybyl MOL2 format for all of its input files. For more information on the original Sybyl MOL2 format, visit http://www.tripos.com/data/support/mol2.pdf. The following is an example of a standard MOL2 formatted file.

Column #	Description
1	Atom number
2	Atom name
3	x coordinate
4	y coordinate
5	z coordinate
6	Atom type
7	Group number
8	Group name
9	Partial charge



A.1. *The protein files*

A.1.1. The *XXXX*_dock.mol2

The format of the file output by PROCESS is a standard MOL2 format with the following changes:

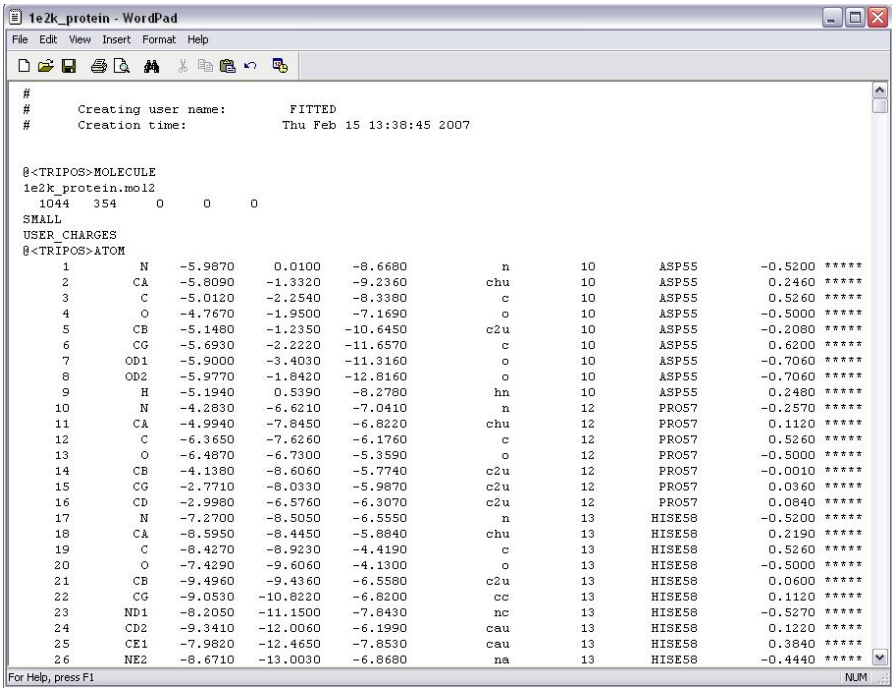
`@<TRIPOS>ATOM` section:

- the atom types (column 6) are Amber united atom types instead of Sybyl atom types
- the group names (column 8) include the advanced residue names (see Appendix A)

`@<TRIPOS>BOND` section:

- only bonds for the flexible residues are listed

Column #	Description
1	Atom number
2	Atom name
3	x coordinate
4	y coordinate
5	z coordinate
6	Atom type
7	Group number
8	Group name
9	Partial charge
10	Misc. Information



A.1.2. The XXXX_score.mol2 file

The format of the file output by PROCESS is a standard MOL2 format with the following changes:

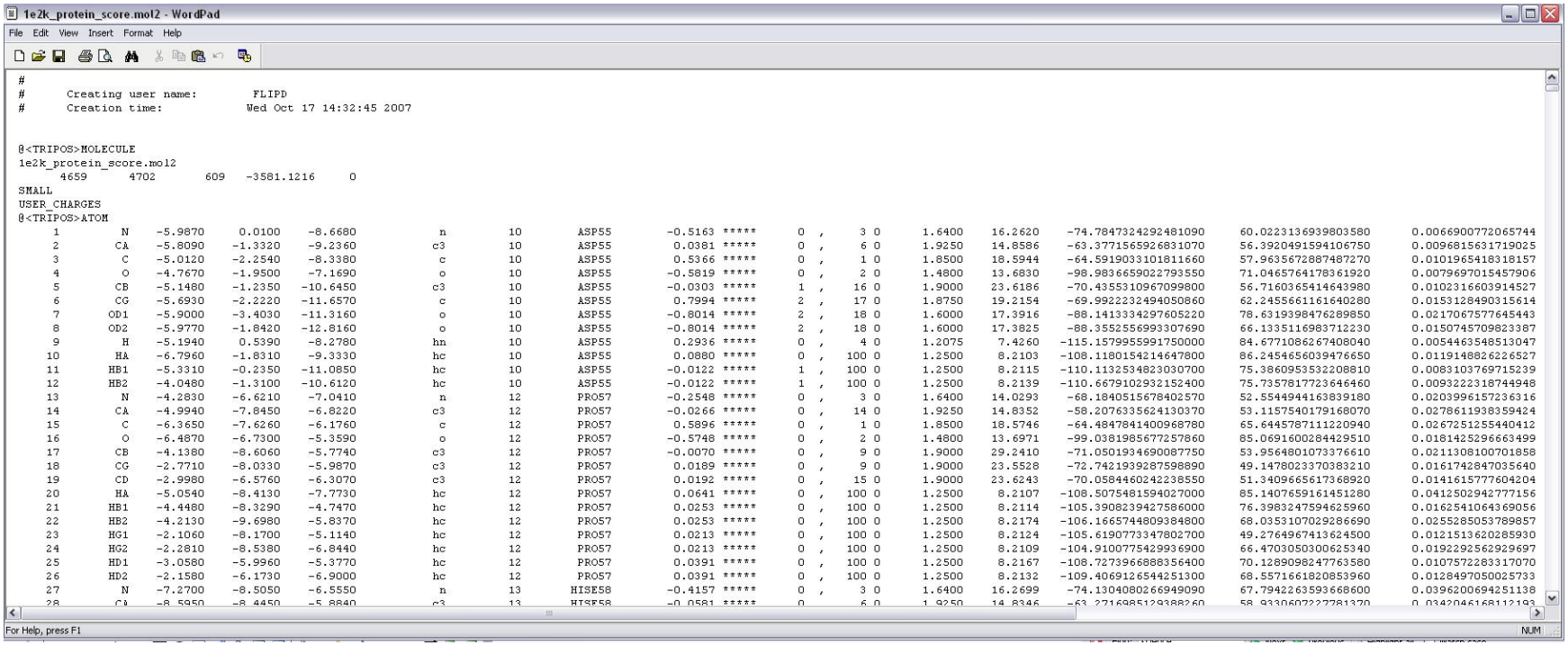
`@<TRIPOS>ATOM` section:

- the atom types (column 6) are Amber united atom types instead of Sybyl atom types
- the group names (column 8) include the advanced residue names (see Appendix A)

`@<TRIPOS>BOND` section:

- all bonds are listed

Column #	Description	Column #	Description
1	Atom number	11	Scaling factor
2	Atom name	12	Place Holder
3	x coordinate	13	OPLS Atom Type
4	y coordinate	14	Place Holder
5	z coordinate	15	van der Waals
6	Atom type		Radii
7	Group number	16	Atom Volume
8	Group name	17	Atomic Solvation
9	Partial charge	18	vdW solvation
10	Misc. Information	19 and >	Water solvation



1e2k_protein_score.mol2 - WordPad

File Edit View Insert Format Help

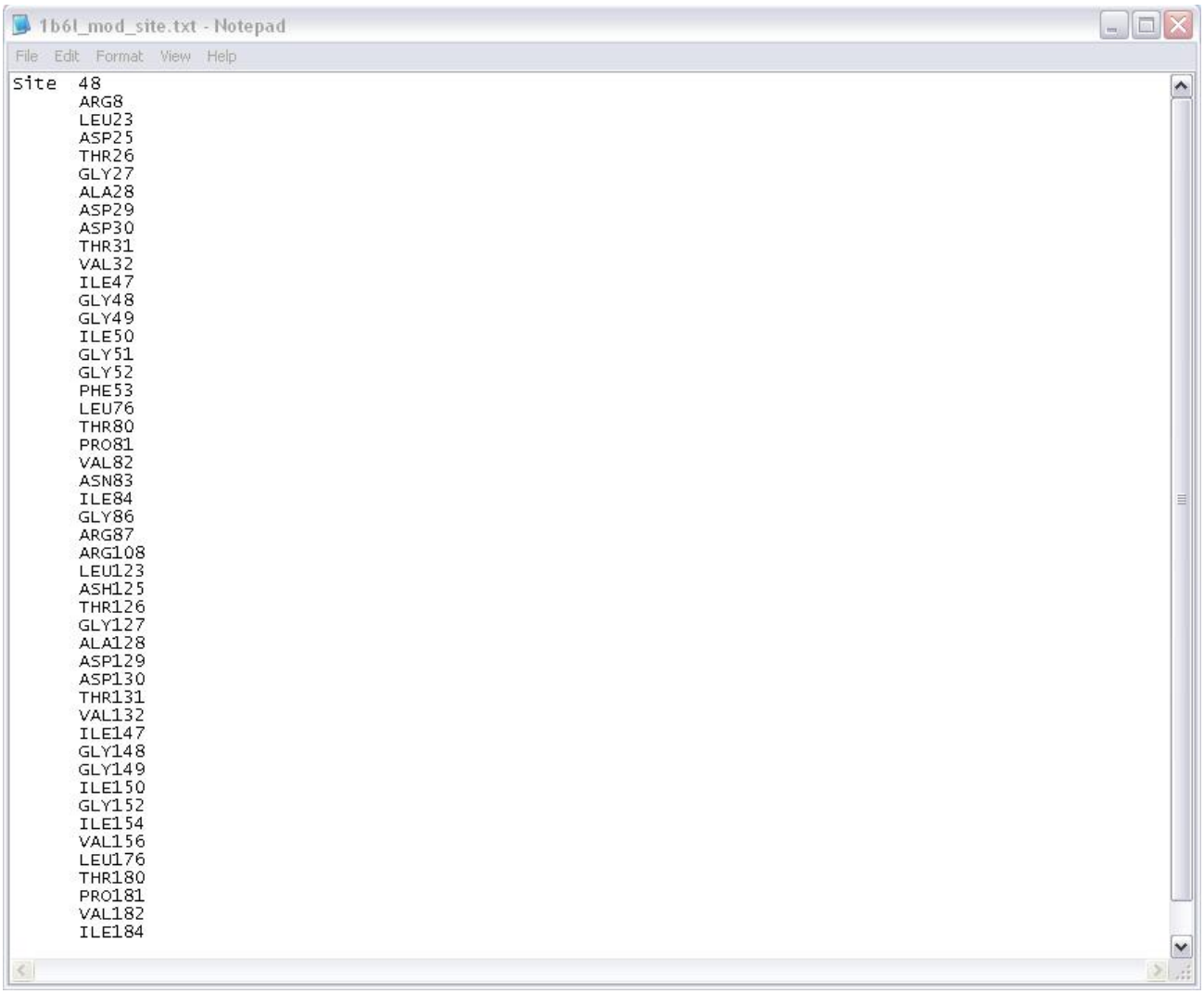
```
#
#      Creating user name:        FLIPD
#      Creation time:             Wed Oct 17 14:32:45 2007


@<TRIPOS>MOLECULE
1e2k_protein_score.mol2
      4659      4702       609   -3581.1216     0
SMALL
USER_CHARGES
@<TRIPOS>ATOM
     1         N    -5.9870     0.0100    -8.6680           n        10       ASP55       -0.5163 *****      0  ,     3 0      1.6400    16.2620    -74.7847324292481090      60.0223136939803580      0.0066900772065744
     2        CA    -5.8090    -1.3320    -9.2360          c3        10       ASP55        0.0381 *****      0  ,     6 0      1.9250    14.8586    -63.3771565926831070      56.3920491594106750      0.0096815631719025
     3         C    -5.0120    -2.2540    -8.3380           c        10       ASP55        0.5366 *****      0  ,     1 0      1.8500    18.5944    -64.5919033101811660      57.9635672887487270      0.0101965418318157
     4         O    -4.7670    -1.9500    -7.1690           o        10       ASP55       -0.5819 *****      0  ,     2 0      1.4800    13.6830    -98.9836659022793550      71.0465764178361920      0.0079697015457906
     5        CB    -5.1480    -1.2350   -10.6450          c3        10       ASP55       -0.0303 *****      1  ,    16 0      1.9000    23.6186    -70.4355310967099800      56.7160365414643980      0.0102316603914527
     6        CG    -5.6930    -2.2220   -11.6570           c        10       ASP55        0.7994 *****      2  ,    17 0      1.8750    19.2154    -69.9922232494050860      62.2455661161640280      0.0153128490315614
     7       OD1    -5.9000    -3.4030   -11.3160           o        10       ASP55       -0.8014 *****      2  ,    18 0      1.6000    17.3916    -88.1413334297605220      78.6319398476289850      0.0217067577645443
     8       OD2    -5.9770    -1.8420   -12.8160           o        10       ASP55       -0.8014 *****      2  ,    18 0      1.6000    17.3825    -88.3552556993307690      66.1335116983712230      0.0150745709823387
     9         H    -5.1940     0.5390    -8.2780          hn        10       ASP55        0.2936 *****      0  ,     4 0      1.2075     7.4260   -115.1579955991750000      84.6771086267408040      0.0054463548513047
    10        HA    -6.7960    -1.8310    -9.3330          hc        10       ASP55        0.0880 *****      0  ,   100 0      1.2500     8.2103   -108.1180154214647800      86.2454656039476650      0.0119148826226527
    11       HB1    -5.3310    -0.2350   -11.0850          hc        10       ASP55       -0.0122 *****      1  ,   100 0      1.2500     8.2115   -110.1132534823030700      75.3860953532208810      0.0083103769715239
    12       HB2    -4.0480    -1.3100   -10.6120          hc        10       ASP55       -0.0122 *****      1  ,   100 0      1.2500     8.2139   -110.6679102932152400      75.7357817723646460      0.0093222318744948
    13         N    -4.2830    -6.6210    -7.0410           n        12       PRO57       -0.2548 *****      0  ,     3 0      1.6400    14.0293    -68.1840515678402570      52.5544944163839180      0.0203996157236316
    14        CA    -4.9940    -7.8450    -6.8220          c3        12       PRO57       -0.0266 *****      0  ,    14 0      1.9250    14.8352    -58.2076335624130370      53.1157540179168070      0.0278611938359424
    15         C    -6.3650    -7.6260    -6.1760           c        12       PRO57        0.5896 *****      0  ,     1 0      1.8500    18.5746    -64.4847841400968780      65.6445787111220940      0.0267251255440412
    16         O    -6.4870    -6.7300    -5.3590           o        12       PRO57       -0.5748 *****      0  ,     2 0      1.4800    13.6971    -99.0381985677257860      85.0691600284429510      0.0181425296663499
    17        CB    -4.1380    -8.6060    -5.7740          c3        12       PRO57       -0.0070 *****      0  ,     9 0      1.9000    29.2410    -71.0501934690087750      53.9564801073376610      0.0211308100701858
    18        CG    -2.7710    -8.0330    -5.9870          c3        12       PRO57        0.0189 *****      0  ,     9 0      1.9000    23.5528    -72.7421939287598890      49.1478023370383210      0.0161742847035640
    19        CD    -2.9980    -6.5760    -6.3070          c3        12       PRO57        0.0192 *****      0  ,    15 0      1.9000    23.6243    -70.0584460242238550      51.3409665617368920      0.0141615777604204
    20        HA    -5.0540    -8.4130    -7.7730          hc        12       PRO57        0.0641 *****      0  ,   100 0      1.2500     8.2107   -108.5075481594027000      85.1407659161451280      0.0412502942777156
    21       HB1    -4.4480    -8.3290    -4.7470          hc        12       PRO57        0.0253 *****      0  ,   100 0      1.2500     8.2114   -105.3908239427586000      76.3983247594625960      0.0162541064369056
    22       HB2    -4.2130    -9.6980    -5.8370          hc        12       PRO57        0.0253 *****      0  ,   100 0      1.2500     8.2174   -106.1665744809384800      68.0353107029286690      0.0255285053789857
    23       HG1    -2.1060    -8.1700    -5.1140          hc        12       PRO57        0.0213 *****      0  ,   100 0      1.2500     8.2124   -105.6190773347802700      49.2764967413624500      0.0121513620285930
    24       HG2    -2.2810    -8.5380    -6.8440          hc        12       PRO57        0.0213 *****      0  ,   100 0      1.2500     8.2109   -104.9100775429936900      66.4703050300625340      0.0192292562929697
    25       HD1    -3.0580    -5.9960    -5.3770          hc        12       PRO57        0.0391 *****      0  ,   100 0      1.2500     8.2167   -108.7273966888356400      70.1289098247763580      0.0107572283317070
    26       HD2    -2.1580    -6.1730    -6.9000          hc        12       PRO57        0.0391 *****      0  ,   100 0      1.2500     8.2132   -109.4069126544251300      68.5571661820853960      0.0128497050025733
    27         N    -7.2700    -8.5050    -6.5550           n        13      HISE58       -0.4157 *****      0  ,     3 0      1.6400    16.2699    -74.1304080266949090      67.7942263593668600      0.0396200694251138
    28        CA    -8.5950    -8.4450    -5.8840          c3        13      HISE58       -0.0581 *****      0  ,     6 0      1.9250    14.8346    -63.2716985129388260      58.9330607227781370      0.0342046168112193
```

For Help, press F1 NUM

A.1.3. **The *XXXX*_site.txt file**

The XXXX_site.txt file is output by PROCESS and contains the binding site residue list. The first line of the file must start with `Site` followed by the number of residues. The following lines (1 per line) list the names of the binding site residues.



A.2. **The Ligand file**

The format output by SMART is based on the MOL2 format. Some modifications were introduced in order to implement the *bitstring* (when using the *filter* mode) describing the presence of functional groups, and to aid in checking the chirality, atom connectivity and ring perception. The changes from the standard MOL2 format are as follows:

`@<TRIPOS>MOLECULE` section:

- the second line (data associated with the molecule) is expanded by a number of fields describing the ligand and the functional groups present (*bitstring*). The presence of a particular group is indicated by a `1` on the respective field. The order of the fields is as follows:

```
number of atoms
number of bonds
molecular weight
net charge
number of hydrogen bond donors
number of hydrogen bond acceptors
number of rotatable bonds
number of rings
number of ionisable groups
number of aromatic group
number of aldehyde
number of ester
number of lactone
number of amide
number of amide
number of lactame
number of acid
number of nitrile
number of imine
number of nitro
number of Michael acceptor
number of azide
number of isocyanate
number of acyl chloride
number of sulphonamide
number of carbamate
number of ammonium
number of oxime
number of secondary amine
number of primary amine
number of ketone
number of boronate
```

number of oxygens

number of nitrogens

number of sulphurs

number of hetero atoms

number of toxic metals

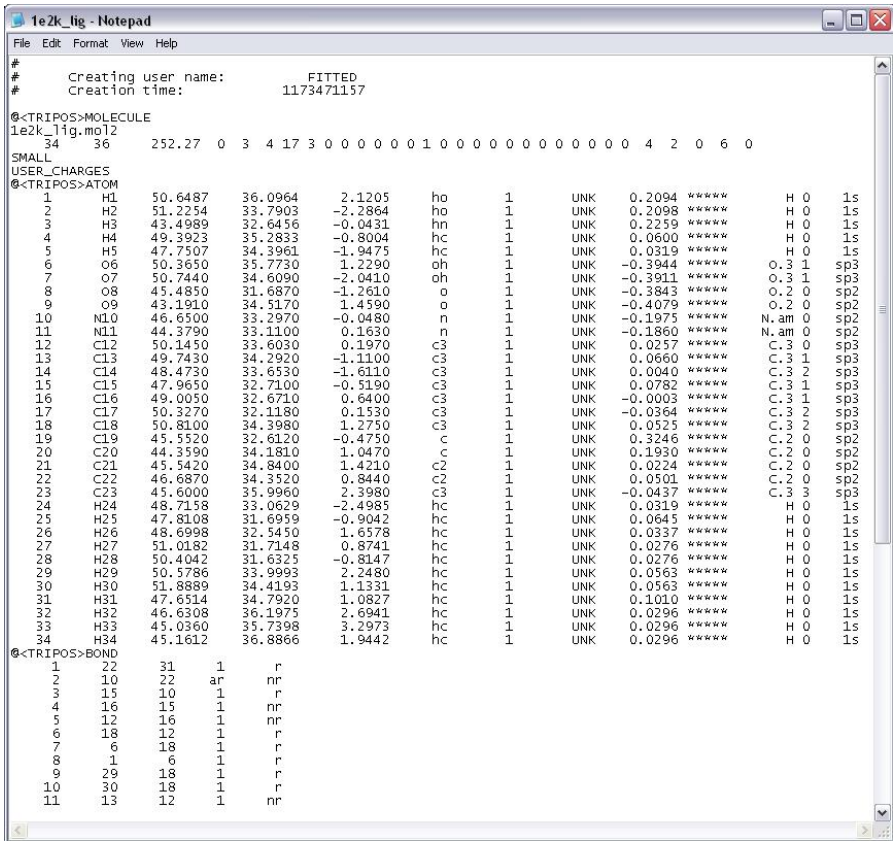
`@<TRIPOS>ATOM` section:

- the Sybyl atom types (column 6) are replaced by GAFF atom types; the corresponding Sybyl atom types are stored in column 11.
- the number of hydrogen atoms attached to an atom is stated on column 12.
- the hybridization of an atom is stated on column 13.

`@<TRIPOS>BOND` section:

- an additional column specifies the bond as rotatable, `r`, or non rotatable, `nr`.

Column #	Discription
1	Atom number
2	Atom name
3	x coordinate
4	y coordinate
5	z coordinate
6	Atom type
7	Group number
8	Group name
9	Partial charge
10	Misc. Information
11	Sybyl Atom type
12	Number of Hydrogens
13	hybridization



1e2k_lig - Notepad

File Edit Format View Help

```
#
#      Creating user name:          FITTED
#      Creation time:           11734711157

@<TRIPOS>MOLECULE
1e2k_lig.mol2
    34     36     252.27  0  3  4 17 3 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 4  2  0  6  0
SMALL
USER_CHARGES
@<TRIPOS>ATOM
     1      H1    50.6487    36.0964     2.1205     ho       1       UNK     0.2094 *****        H 0    1s
     2      H2    51.2254    33.7903    -2.2864     ho       1       UNK     0.2098 *****        H 0    1s
     3      H3    43.4989    32.6456    -0.0431     hn       1       UNK     0.2259 *****        H 0    1s
     4      H4    49.3923    35.2833    -0.8004     hc       1       UNK     0.0600 *****        H 0    1s
     5      H5    47.7507    34.3961    -1.9475     hc       1       UNK     0.0319 *****        H 0    1s
     6      O6    50.3650    35.7730     1.2290     oh       1       UNK    -0.3944 *****      O.3 1    sp3
     7      O7    50.7440    34.6090    -2.0410     oh       1       UNK    -0.3911 *****      O.3 1    sp3
     8      O8    45.4850    31.6870    -1.2610      o       1       UNK    -0.3843 *****      O.2 0    sp2
     9      O9    43.1910    34.5170     1.4590      o       1       UNK    -0.4079 *****      O.2 0    sp2
    10     N10    46.6500    33.2970    -0.0480      n       1       UNK    -0.1975 *****     N.am 0    sp2
    11     N11    44.3790    33.1100     0.1630      n       1       UNK    -0.1860 *****     N.am 0    sp2
    12     C12    50.1450    33.6030     0.1970     c3       1       UNK     0.0257 *****      C.3 0    sp3
    13     C13    49.7430    34.2920    -1.1100     c3       1       UNK     0.0660 *****      C.3 1    sp3
    14     C14    48.4730    33.6530    -1.6110     c3       1       UNK     0.0040 *****      C.3 2    sp3
    15     C15    47.9650    32.7100    -0.5190     c3       1       UNK     0.0782 *****      C.3 1    sp3
    16     C16    49.0050    32.6710     0.6400     c3       1       UNK    -0.0003 *****      C.3 1    sp3
    17     C17    50.3270    32.1180     0.1530     c3       1       UNK    -0.0364 *****      C.3 2    sp3
    18     C18    50.8100    34.3980     1.2750     c3       1       UNK     0.0525 *****      C.3 2    sp3
    19     C19    45.5520    32.6120    -0.4750      c       1       UNK     0.3246 *****      C.2 0    sp2
    20     C20    44.3590    34.1810     1.0470      c       1       UNK     0.1930 *****      C.2 0    sp2
    21     C21    45.5420    34.8400     1.4210     c2       1       UNK     0.0224 *****      C.2 0    sp2
    22     C22    46.6870    34.3520     0.8440     c2       1       UNK     0.0501 *****      C.2 0    sp2
    23     C23    45.6000    35.9960     2.3980     c3       1       UNK    -0.0437 *****      C.3 3    sp3
    24     H24    48.7158    33.0629    -2.4985     hc       1       UNK     0.0319 *****        H 0    1s
    25     H25    47.8108    31.6959    -0.9042     hc       1       UNK     0.0645 *****        H 0    1s
    26     H26    48.6998    32.5450     1.6578     hc       1       UNK     0.0337 *****        H 0    1s
    27     H27    51.0182    31.7148     0.8741     hc       1       UNK     0.0276 *****        H 0    1s
    28     H28    50.4042    31.6325    -0.8147     hc       1       UNK     0.0276 *****        H 0    1s
    29     H29    50.5786    33.9993     2.2480     hc       1       UNK     0.0563 *****        H 0    1s
    30     H30    51.8889    34.4193     1.1331     hc       1       UNK     0.0563 *****        H 0    1s
    31     H31    47.6514    34.7920     1.0827     hc       1       UNK     0.1010 *****        H 0    1s
    32     H32    46.6308    36.1975     2.6941     hc       1       UNK     0.0296 *****        H 0    1s
    33     H33    45.0360    35.7398     3.2973     hc       1       UNK     0.0296 *****        H 0    1s
    34     H34    45.1612    36.8866     1.9442     hc       1       UNK     0.0296 *****        H 0    1s
@<TRIPOS>BOND
     1     22     31     1      r
     2     10     22    ar     nr
     3     15     10     1      r
     4     16     15     1     nr
     5     12     16     1     nr
     6     18     12     1      r
     7      6     18     1      r
     8      1      6     1      r
     9     29     18     1      r
    10     30     18     1      r
    11     13     12     1     nr
```

A.3. **The binding site cavity file**

The binding site cavity file is used to determine the empty space within the protein via a collection of spheres of different radius. It resembles a MOL2 formatted file with the following changes:

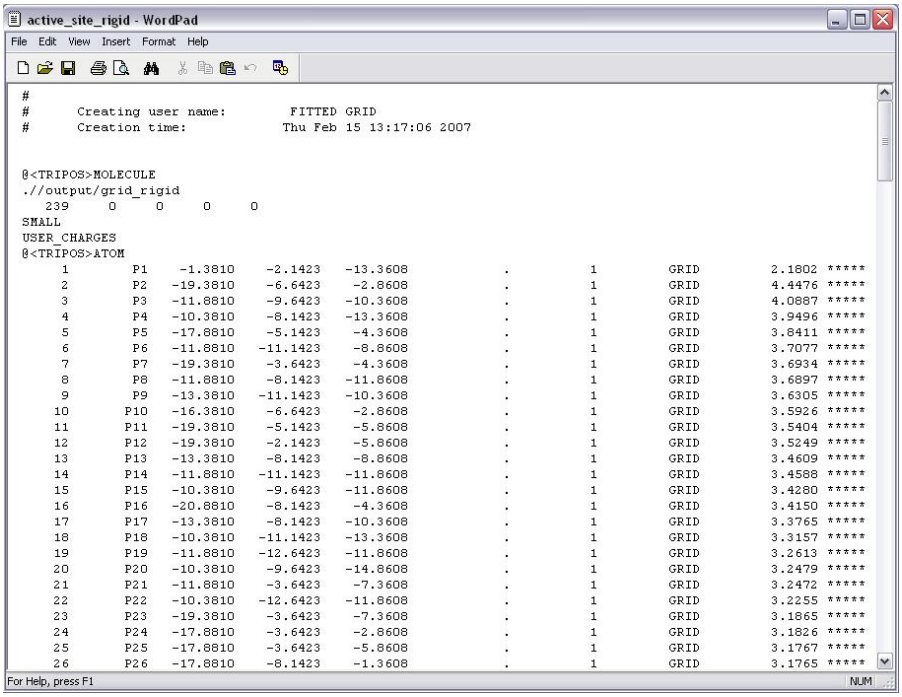
`@<TRIPOS>MOLECULE` section:

- on the second line the number of spheres is specified as the first field; fields 2-5 are `0`.

`@<TRIPOS>ATOM` section:

- column 6 (Sybyl atom type) is unnecessary, therefore it is replaced by a dash.
- column 9 (partial charges) is replaced by the radius of the sphere.

Column #	Discription
1	Point number
2	Point name
3	x coordinate
4	y coordinate
5	z coordinate
6	Point Type
7	Group number
8	Group name
9	Radius
10	Misc. information



A.4. **The interaction site, binding site cavity and XXXX_IS.mol2 files**

The interaction sites and binding site cavity file are used to create conformations that already have good interaction with the protein. Again the format resembles mol2 format with the addition of columns for the interaction site type and weight.

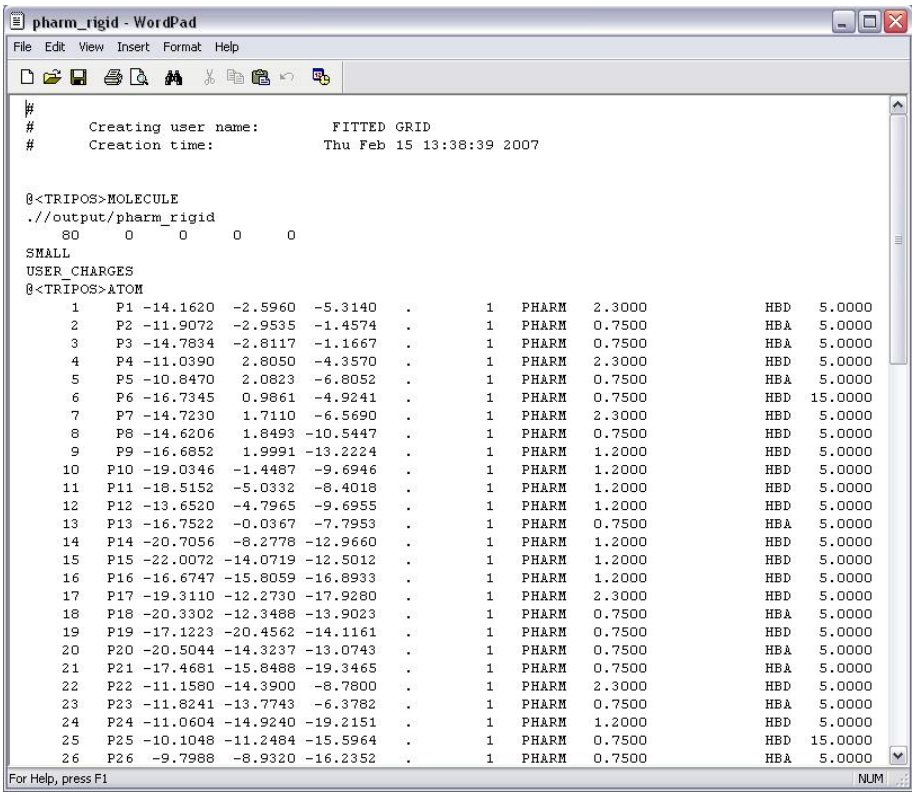
`@<TRIPOS>MOLECULE` section:

- on the second line the number of constraints is specified as the first field; fields 2-5 are `0`.

`@<TRIPOS>ATOM` section:

- column 6 (Sybyl atom type) is unnecessary, therefore it is replaced by a dash.
- column 7 (group number) is replaced by a point type descriptor.
- column 9 (partial charges) is replaced by the radius of the constraint.
- column 10 specifies the type of the pharmacophoric point (HBD, HBA, HYD, ARO, or any combination such as HBA/HYD).
- column 11 specifies the weight of the constraint.

Column #	Discription
1	Point number
2	Point name
3	x coordinate
4	y coordinate
5	z coordinate
6	–
7	Point type
8	`PHARM`
9	Radius
10	Pharmacophoric type
11	Weight



A.5. **The force field file**

The force field file is where all the parameters for the FITTED force field are kept. Additionally, SMART uses the force field file to assign MMFF charges to molecules if so requested. The force field is a modified GAFF force field with MMFF charge parameters in free format, so it can be edited with any text editor. Although most of the parameters for drug-like molecules are present, some may be missing. When adding a parameter to the force field file, some rules must be followed.

Each section starts with a title (e.g., **`#fitted_bond_parameters`**), followed by the actual parameters (i.e, `1.0 1 c c 1.5500 290.100`) and ends with a line with blank parameters designated by stars (i.e., `0.0 1 * * 0.0000 0.000`). The title and end lines should not be removed and any line added before the title line and after the end line will be ignored.

FITTED also allows for the use of wildcard parameters for angles and torsion parameters, where I, J, K or L can represent any atom type, by using the wildcard character `*` in the respective column. Using wildcards (`*`) for all the atoms will be read as an end line.

A.5.1. **Adding parameters to the bond list**

The bond list starts 2 lines following the **`#fitted_bond_parameters`** title, and the end is signaled by having both the I and J atom types (columns 3 and 4) as `*`. Any parameters added after this last line will be ignored. The parameters added must be in a single line in the following format

Units: R (Å), K (kcal/mol Å)

#1	#2	#3	#4	#5	#6
Force field file version number	Reference number	Atom type of I	Atom type of J	R_0	K_2

```
#---------------------------------------------
#Ver  Ref     I     J          R0          K2
#---------------------------------------------
#fitted_bond_parameters
#---------------------------------------------
 1.0   1      c     c         1.5500     290.100
 1.0   1      c     c1        1.4600     379.800
 1.0   1      c     c2        1.4060     449.900
 1.0   1      c     c3        1.5080     328.300
 1.0   1      c     ca        1.4870     349.700
[...]
 1.2   1     ct     ss        1.7700     256.600
 1.2   1     ct     nh        1.3640     449.000
 1.2   1     nt     nt        1.3400     450.000
 0.0   1      *      *        0.0000       0.000
```

A.5.2. **Adding parameters to the angle list**

The angle list starts 2 lines below the **`#fitted_angle_parameters`** title, and the end is signaled by having all I, J and K atom types (columns 3-5) as `*`. FITTED also allows for the use of wildcard parameters, where I and/or K can represent any atom, by using the wildcard character `*` in the respective column. Parameters added to the force field including wildcards should be placed at the *end* of the angle list. The less specific the parameter (higher number of wildcards), the lower in the list it should be placed. The parameter added must be in a single line in the following format:

Units: * add units for R, K (kcal/mol rad)

#1 Force field file version number	#2 Reference number	#3 Atom type of I	#4 Atom type of J	#5 Atom type of K	#6 R_0	#7 K_2

```
#-----------------------------------------------------
# E = K2 * (Theta - Theta0)^2
#-----------------------------------------------------
#Ver  Ref     I     J     K       Theta0          K2
#-----------------------------------------------------
#fitted_angle_parameters
#-----------------------------------------------------
 1.0   1     hw    ow    hw      104.5200     100.0000
 1.0   1     hw    hw    ow      127.7400       0.0000
 1.0   1     br    c     br      113.1000      66.9000
 1.0   1     br    c     c3      110.7400      63.3000
 1.0   1     br    c     o       121.4600      63.2000
[...]
 1.2   1     *     n4    hn      109.0000      35.0000
 1.2   1     *     n4    *       109.5000      60.0000
 1.2   1     *     na    *       120.0000      60.0000
 1.2   1     *     nb    *       120.0000      60.0000
 0.0   1     *      *    *         0.0000       0.0000
```

A.5.3. **Adding parameters to the torsion list**

The torsion list starts 2 lines below the **`#fitted_torsion_parameters`** title, and the end of the list is signaled by having all I, J, K and L atom types as `*`. FITTED also allows for the use of wildcard parameters, where I and L can represent any atom, by using the wildcard character `*` in the respective column. Parameters added to the force field including wildcards should be placed at the *beginning* of the torsion list. The parameter added must be in a single line in the following format:

Units: V (kcal/mol)

#1	#2	#3	#4	#5	#6	#7	#8	#9	#10	#11	#12	#13	#14
Force field file version number	Reference number	Atom type of I	Atom type of J	Atom type of K	Atom type of L	V1	ϕ_{01}	V2	ϕ_{02}	V3	ϕ_{03}	V4	ϕ_{04}

```
#-------------------------------------------------------------------------------------------
# E = SUM(n=1,4) { (Vn/m) * [ 1 + cos(n*Phi - Phi0(n)) ] }
# m = multiplicity or total number of torsions centered on the same bond.
#-------------------------------------------------------------------------------------------
#Ver  Ref     I     J     K     L   V1  Phi0     V2    Phi0    V3   Phi0    V4     Phi0
#-------------------------------------------------------------------------------------------
#fitted_torsion_parameters
#-------------------------------------------------------------------------------------------
 1.0   1    *     c     c     *   0.0000  0.00  1.2000 180.00  0.0000 0.00  0.0000  0.00
 1.0   1    *     c     c1    *   0.0000  0.00  0.0000 180.00  0.0000 0.00  0.0000  0.00
 1.0   1    *     c     cg    *   0.0000  0.00  0.0000 180.00  0 .000 0.00  0.0000  0.00
 1.0   1    *     c     ch    *   0.0000  0.00  0.0000 180.00  0.0000 0.00  0.0000  0.00
 1.0   1    *     c     c2    *   0.0000  0.00  8.7000 180.00  0.0000 0.00  0.0000  0.00
[...]
 1.0   1   hc    c3    c3    f   0.1900  0.00  0.0000   0.00  0.0000 0.00  0.0000  0.00
 1.0   1   hc    c3    c3    cl  0.2500  0.00  0.0000   0.00  0.0000 0.00  0.0000  0.00
 1.0   1   hc    c3    c3    br  0.5500  0.00  0.0000   0.00  0.0000 0.00  0.0000  0.00
 0.0   1   *      *     *    *   0.0000  0.00  0.0000   0.00  0.0000 0.00  0.0000  0.00
```

A.5.4. **Adding parameters to the out-of-plane list**

The angle list starts 2 lines below the **`#fitted_oop_parameters`** title, and the end of the list is signaled by having all I, J, L and K atom types as `*`. FITTED also allows for the use of wildcard parameters, where I, K and/or L can represent any atom, by using the wildcard character `*` in the respective column. Parameters added to the force field including wildcards should be placed at the *end* of the angle list. The less specific the parameter (higher number of wildcards), the lower in the list it should be placed. The parameter added must be in a single line in the following format:

Units: K (kcal/mol)

#1	#2	#3	#4	#5	#6	#7	#8	#9
Force field file version number	Reference number	Atom type of I	Atom type of J	Atom type of K	Atom type of L	Kchi	N	Chi0

```
#-----------------------------------------------------------------------
# E = Kchi * [ 1 + cos(n*Chi - Chi0) ]
#-----------------------------------------------------------------------
#Ver  Ref     I     J     K     L           Kchi        n           Chi0
#-----------------------------------------------------------------------
#fitted_oop_parameters
#-----------------------------------------------------------------------
 1.0   1     c     c2    c2    c3         1.1000        2       180.0000
 1.0   1     c     ca    c3    ca         1.1000        2       180.0000
 1.0   1     c     n     c3    hn         1.1000        2       180.0000
 1.0   1     c     n     c3    o          1.1000        2       180.0000
 1.0   1     c2    na    c2    c3         1.1000        2       180.0000
[...]
 1.0   1     c2    c2    *     *         10.5000        2       180.0000
 1.0   1     o     c     *     *          1.1000        2       180.0000
 1.0   1     *     c     *     *         10.5000        2       180.0000
 1.0   1     *     ca    *     *          7.1000        2       180.0000
 0.0   1     *     *     *     *          0.0000        2       180.0000
```

A.5.5. **Adding parameters to the van der Waals list**

The vdW list starts 2 lines below the **`#fitted_vdW_parameters`** title, and the end of the list is signaled by I atom type as `*`. The parameter added must be in a single line in the following format:

Units: R_i (Å), ESPI (kcal/mol)

#1	#2	#3	#4	#5
Force field file version number	Reference number	Atom type of I	R_i^*	ESPI

```
#-------------------------------------------------
#type r-eps
#combination arithmetic
#-------------------------------------------------
# E = EPSij * { (Rij*/Rij)^12 - 2(Rij*/Rij)^6 }
# where  EPSij = sqrt( EPSi * EPSj)
#         Rij* = (Ri* + Rj*)/2
#-------------------------------------------------
#Ver  Ref     I            Ri*           EPSi
#-------------------------------------------------
#fitted_vdW_parameters
```

```
#-------------------------------------------------
 1.0   1     h1           2.7740       0.01570
 1.0   1     h2           2.5740       0.01570
 1.0   1     h3           2.3740       0.01570
 1.0   1     h4           2.8180       0.01500
 1.0   1     h5           2.7180       0.01500
[...]
 1.2   1     n0           3.7360       0.00277
 1.2   1     k            5.3160       0.000328
 1.2   1     zn2          2.2000       0.0125
 0.0   1       *          0.0000       0.00000
```

A.5.6. **Adding parameters to the hydrogen bond list**

The Hbond list starts 2 lines below the **`#fitted_Hbond_parameters`** title, and the end of the list is signaled by having both the I and J atom types as `*`. The parameter added must be in a single line in the following format:

Units: A, B (kcal/mol)

#1	#2	#3	#4	#5	#6
Force field file version number	Reference number	Atom type of I	Atom type of J	A	B

```
#-------------------------------------------------
# E = Aij/r^12 - Bij/r^10
#-------------------------------------------------
#Ver  Ref     I    J           A              B
#-------------------------------------------------
#fitted_Hbond_parameters
#-------------------------------------------------
 1.0   3     hw   nb        7557.0000     2385.0000
 1.0   3     hw   nc       10238.0000     3071.0000
 1.0   3     hw   o         7557.0000     2385.0000
 1.0   3     hw   oh        7557.0000     2385.0000
 1.0   3     hw   os        7557.0000     2385.0000
[...]
 1.2   3     zn   s6       15000.0000     5000.0000
 1.2   3     zn   ss       15000.0000     5000.0000
 1.2   3     zn   sh       15000.0000     5000.0000
 0.0   3      *    *           0.0000        0.0000
```

A.5.7. **Adding parameters to the bond charge increment list**

The bond charge increment list starts below the **`#fitted_charge_parameters`** title, and the end of the list is signaled by having both the I and J atom types as `*`. Each line specifies a bond charge increment for a bond between atoms of type I and J (bci_{IJ}), such that the resulting charge on J is the bci_{IJ}, while on I is $–bci_{IJ}$.The parameter added must be in a single line in the following format:

#1	#2	#3	#4	#5
Version number	Atom type of I	Atom type of J	bci	Comment

```
############################
#Cl I   J     bond_inc source
############################
#fitted_charge_parameters
0   1    1    0.0000   #C94
0   1    2   -0.1382   C94
0   1    3   -0.0610   #C94
0   1    4   -0.2000   #X94
[...]
0  80   81   -0.4000   #C94
0 101    1    0.0000   empirical
0 101    6   -0.1900   empirical
0 101   37   -0.0000   empirical
*  *    *     *      *
```

A.5.8. **Adding parameters to the partial bond charge increment / formal charge adjustment factor list**

As a more general way of describing bci's, MMFF includes a partial bci parameter that is assigned to each atom type [15]; a bci for a bond can be obtained as the sum of the partial bci corresponding to each atom type involved. Additionally, the formal charge on some groups is spread among neighbouring atoms; this is specified in the formal charge adjustment factor for the central atom type in those functional groups [15].

The parameter list starts below the **`#fitted_mmff_addl_charges`** title, and the end of the list is signaled by having both the I and J atom types as `*`. The parameter added must be in a single line in the following format:

#1	#2	#3	#4	#5
Version number	Atom type	Partial bci	Formal charge adj	Comment

```
###
# MMFF Partial Bond Charge Incs and Formal-Charge Adj. Factors: 19-MAY-1994
#
# source: J. Comp. Chem. 17, 616 (1996)
###
# type    pbci      fcadj   Origin/Comment
###
#fitted_mmff_addl_charges
0   1     0.000     0.000   E94
0   2    -0.135     0.000   E94
0   3    -0.095     0.000   E94
0   4    -0.200     0.000   E94
[...]
0  96     2.000     0.000  Ionic charge
0  97     1.000     0.000  Ionic charge
0  98     2.000     0.000  Ionic charge
0  99     2.000     0.000  Ionic charge
*  *      *         *      *
```

Appendix B: FITTED errors and warnings

`ERROR: Molecule outside maximum number of angles.`

- FITTED can only handle molecules with [3 × #atoms] angles. If there are more then please contact the developers at nicolas.moitessier@mcgill.ca

`ERROR: Molecule outside maximum number of torsions.`

- FITTED can only handle molecules with [6 × #atoms] torsions. If there are more then please contact the developers at nicolas.moitessier@mcgill.ca

`ERROR: Forcefield file <forcefield filename> not found`

- The force field file listed in the keyword file is not found in the `forcefield/` directory.

`ERROR: Molecule too big for active site`

- Increase **`GI_Num_of_Trials`**
- Increase **`GI_Initial_E`**
- Increase **`GI_Minimized_E`**
- Increase **`Grid_Size`** in PROCESS to create a larger active site cavity.
- If none of these work, the molecule is too big for the active site and cannot be docked.

`ERROR: Protein input file <Protein file name> not present`

- The protein file could not be found in the `input/` directory.

`ERROR: Ligand input file <ligand file name> not present`

- The ligand file could not be located in the `input/` directory.

`ERROR: Binding_Site_Cav file <Active site filename> not present`

- **`Binding_Site_Cav`** file could not be located within the `input/` directory. Without an active site file the docking may take longer and be less accurate.

`WARNING: Binding_Site_Cav needed for generation of initial population`

- FITTED issues this warning but will not exit. Without an active site file the docking may take longer and be less accurate.

`ERROR: Reference file <Reference file name> not present.`

- The reference file could not be located within the `input/` directory.

`ERROR: Missing Forcefield Parameters`

- FITTED exits because there are missing force field parameters. Either add them to the force field file or use Parameters Auto keyword to have FITTED automatically assign parameters.

`WARNING: Missing Forcefield parameters, assigning parameters automatically`

- List below is the parameter which was assigned automatically. If you do not like the automatic assignment add the parameter with your desired value into the force field file

`ERROR: <keyword_filename> Can not be opened`

- If the keyword file is not found in the keyword directory then an `error.log` will be created with this error. Please put keyword in `keyword/` directory.

`ERROR: Coordinates not found in protein structural file`

- @<TRIPOS>ATOM is not found in the protein file preceding the coordinates

`ERROR: Array size for number of protein atoms and bonds not in Protein 1 mol2 file.`

- @<TRIPOS>MOLECULE is not found in the first protein mol2 file

`ERROR: Array size for number of ligand atoms and bonds not in Ligand mol2.`

- @<TRIPOS>MOLECULE is not found in the first protein mol2 file

`ERROR: Coordinates not found in ligand file`

- @<TRIPOS>ATOM is not found in the ligand file preceding the coordinates

`ERROR: Check water names and atom types.`

- The water atom name and atom types are non-standard. Change to standard names.

`ERROR: Bonds not found in ligand file`

- @<TRIPOS>BOND is not found in the ligand file preceding the list of bonds

`ERROR: No assignment of Rotatable bonds`

- Please assign rotatable bonds either manually or by using SMART

`ERROR: Bonds not found in protein file`

- @<TRIPOS>BOND is not found in the protein file preceding the list of bonds

`ERROR: Protein keyword not found in keyword file.`

- The keyword **`Protein`** is not found within the keyword file. Please include this within you keyword file followed by the number of protein files and on the next lines a list of the protein files for the docking/virtual screening run.

`ERROR: Can not find <residue name>`

- Can not a residue listed in the keyword file. Please check the spelling.

`ERROR: Flex file <protein file name> not found`

- Make sure `<protein file name>_flex.txt` is in the input directory.

`Error: Can not find coordinates in <Binding_Site_filename> is not present in the active site cavity file`

- @<TRIPOS>MOLECULE is not found in the **`Binding_Site_Cav`** file

`Error: Can not find coordinates in <Binding_Site_filename>`

- @<TRIPOS>ATOM is not found in the **`Binding_Site_Cav`** file preceding the coordinates

`Error: Can not find coordinates in <pharmacophore_filename>`

- @<TRIPOS>ATOM is not found in the **`Pharmacophore`** file preceding the coordinates

`Error: Can not find coordinates in <Interaction_Sites_filename>`

- @<TRIPOS>ATOM is not found in the **`Interaction_Sites`** file preceding the coordinates

`Error: Ligand can not match minimum pharmacophore`

- Increase value of **`Min_PharmScore`**.

`Error: Ligand can not match minimum Interaction Sites`

- Increase value of **`Min_MatchScore`**.

`ERROR: Reference file <reference_filename> not present`

- The reference file is not located within the `input/` directory.

`ERROR: Invalid parameter specified for covalent residue.`

- Make sure the residue name is listed in the keyword the same way it is listed in the protein file.

`ERROR: FITTED cannot find O/S and H for the covalent residue`

- Format in protein input file may be incorrect. In particular, make sure that for serine the alcohol atom names are set as OG and HG.

`ERROR: Invalid parameter specified for other catalytic residue.`

- Make sure the residue and atom name are specified the same in the keyword and protein file.

`ERROR: The proteins do not have the same number of atoms`

- Make sure to run ProCESS with all proteins in one keyword file.

`ERROR: Problem with creation of z-matrix for ligand.`

- Make sure there is not a missing bond in the bond list of the ligand mol2 file. FITTED cannot handle mol2 with multiple structures.

`ERROR: Problem with creation of z-matrix for active site residue <residue_name>.`

- A bond is missing from the bond list in one of the protein mol2 files. Either add the missing bond(s) or remove the residue from the `XXXX_site.txt` file if it not critical to binding of the ligand.

Appendix C: PROCESS errors and warnings

`Number of proteins not in keyword file.`

- If the number of protein files does not follow Protein_Conformations keyword.

`Coordinates not found in structural file`

- If in either the protein or ligand mol2 file @<TRIPOS>ATOM does not precede the coordinates of the structure.

`Bonds not found in structural file`

- If in either the protein or ligand mol2 file @<TRIPOS>BOND does not precede the bond list.

`Ligand file not present now closing`

- If **`Ligand`** is not found in the keyword file.

`User wanted automatic finding of active site center, Ligand Reference not given.`

- If the keyword **`AutoFind_Site`** is used in the keyword and **`Ligand`** is not found in the keyword file.

`<Protein file name> file not present. Program now Closing.`

- The protein file given can not be found in the `input/` directory.

`<Ligand file name> file not present. Program now Closing`

- The ligand file given can not be found in the `input/` directory

`Side chain <residue name> Not found in <protein file name>`

- The residue given can not be found in the protein file.

`Unknown residue name: <residue name>`

- The residue is not known. Refer to Tables 1a and 1b for accepted residue names.

Appendix D: SMART errors and warnings

The following is a list of errors and warnings that SMART outputs to the corresponding log file in the `output/` directory. Errors indicate serious problems that cause SMART to either skip a molecule or exit. Warnings are potential problems that might cause the SMART output to be incorrect; critical examination of the output and input structures in these cases is *strongly* encouraged.

`ERROR: File <filename> cannot be opened.`

- The input file specified could not be read. Make sure that the file is located in the `input/` directory. Check the spelling and the file permissions.

`ERROR: Atom <atom_name> cannot find element`

- The specified atom has a non-standard Sybyl atom type, or is not in the range of atomic numbers 1-35 (H-Br), 44-46 (Ru-Pd), 53 (I) or 78 (Pt). Without a proper element assignment, atom types cannot be assigned. In particular, look for: i) P atoms in phosphates and analogous functional groups: the Sybyl atom type for the P atom should be “P.3”; ii) S atoms in sulfoxides, sulfones and derivatives: the Sybyl atom type for the S atom should be “S.o” or “S.o2” respectively.

`ERROR: could not write to <filename>`

- The specified output file could not be written. Check permissions on the `output/` and parent directories, that there is enough empty space in the volume and that the filename is valid.

`ERROR: cannot create Z-matrix. Does the molecule have a torsion?`

- In order to be processed by SMART, a molecule must at least have 4 atoms connected sequentially in order to define a torsion. If a torsion cannot be defined, the molecule is skipped.

`WARNING: Sum of partial charges does not equal net charge`

- When assigning MMFF charges, the partial charges assigned do not match the predicted formal charge. Check atom type assignment and bond connectivity.

`WARNING: Cannot assign atomic weight to atom <atom_number> <atom_name>`

- When generating the bit string, the molecular weight is calculated from the sum of atomic weights. Currently, only atoms of atomic number 1-17 (H-Cl), 34-35 (Se, Br) and 53 (I) are parameterized.

`WARNING: Atom <atom_name> has a formal charge of <formal charge>`

- When automatically assigning the bond orders (-assign_bond command-line option), this message is output to the log file for every atom with a formal charge higher than 1. Check the bond order assignment in these molecules to make sure it is correct.

`WARNING: Missing bond increment. Bond # <bond_number> Atoms <atom_name1> <atom_name2>; MMFF atom types <MMFF_type1> and <MMFF_type2>. Bond increment set to 0.`

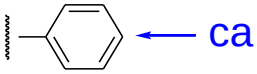
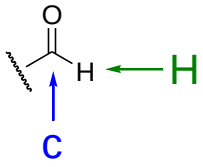
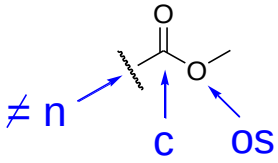
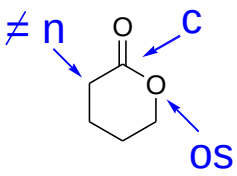
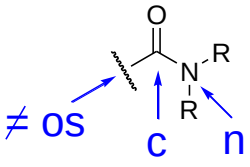
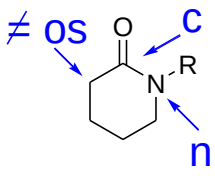
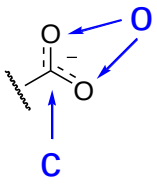
- When automatically assigning the MMFF charges (`-charge` command-line option), this message is output for every pair of atoms for which the bond increment is not parameterized. Add the bond increment in the `forcefield/fitted_ff.txt` file.

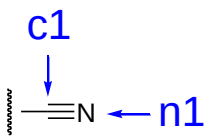
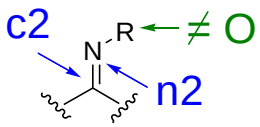
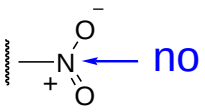
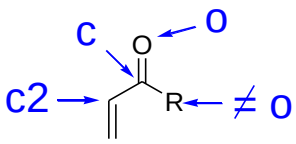
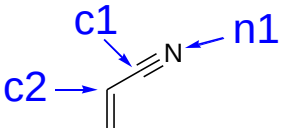
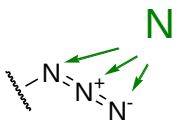
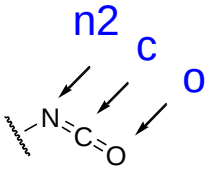
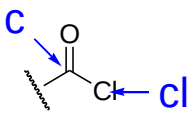
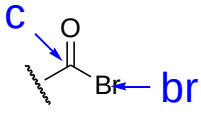
`WARNING: Could not assign charges to molecule`

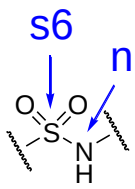
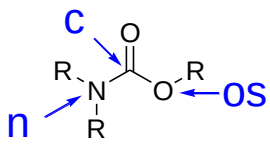
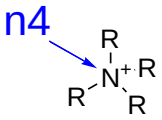
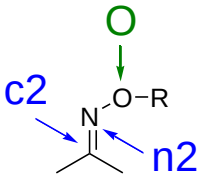
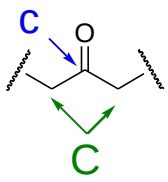
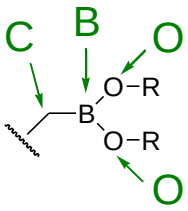
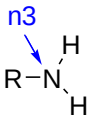
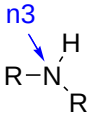
- When automatically assigning the MMFF charges (`-charge` command-line option), this message is indicative of other problems with the charge assignment. Look for warning messages appearing before this one in the log file.

Appendix E: Functional group definitions

Table 1 - Definition of functional groups in SMART (blue = atom type, green = element)

Keyword	Description	
Aromatic		An aromatic group is present if a *ca* atom type is within the molecule
Aldehyde		An aldehyde is present if there is a *c* atom type in the molecule bound to a hydrogen.
Ester		An ester is present if there is a *c* atom type bound to an atom with an *os* atom type with the *c* not bound to an a *n* atom type, with both *c* and *os* atoms being acyclic.
Lactone		A lactone is present there is a *c* atom type bound to an atom with an *os* atom type with the *c* not bound to an a *n* atom type, with *c* and *os* atoms involved in a ring.
Amide		An amide is present if there is a *c* atom type bound to an atom with an *n* atom type with the *c* not bound to an *os* atom type, with both *c* and *n* atoms being acyclic.
Lactame		A lactame is present if there is a *c* atom type bound to an atom with an *n* atom type with the *c* not bound to an a *os* atom type. With both *c* and *n* atoms being cyclic.
Acid		An acid (carboxylate) is present if an atom with a *c* atom type is bound to two atoms with *o* atom types.

Nitrile		A nitrile is present if an atom with a *c1* atom type is bound to an atom with an *n1* atom type.
Imine		An imine is present if an atom with a *c2* atom type is bound to an atom with an *n2* atom type, both acyclic; R cannot be an oxygen atom.
Nitro		A nitro is present if there is an atom with an *no* atom type within the molecule.
Acceptor	 	A Michael acceptor is present if an atom with an atom type of *c2* is bound to either 1) an atom with a *c* atom type which is not a carboxylate, or 2) a nitrile group. The bond between *c2* and *c/c1* must be acyclic.
Azide		An azide is present if there are three acyclic nitrogens in a linear formation.
Isocyanate		An isocyanate is present if an atom with an atom type of *c is* bound to 2 atoms, one with an atom type of *n2* and another with an atom type of *o*, where the *c – n2* bond is acyclic.
Acyl_Chloride	 	An acyl chloride is present if an atom with a atom type of *c* is bound to an atom with an atom type of *cl* or *br.*

Sulphonamide		A sulphonamide is present when an atom with an atom type of *s6* is bound to an atom with an atom type of *n*.
Carbamate		A carbamate is present when an atom with an atom type of *c* is bound to an atom with an *n* atom type and an atom with an *os* atom type.
Ammonium		An ammonium is present if there is an atom with an *n4* atom type.
Oxime		An oxime is present if there is an atom with a *c2* atom type bound to an atom with an *n2* atom type which in turn is bound to an oxygen atom.
Ketone		A ketone is present if an atom with a *c* atom type *c* is bound to 2 carbon atoms.
Boronate		A boronate is present if there is a boron atom bound to a carbon and two oxygens.
Primary_Amine		A primary amine is present if there is an atom with an atom type of *n3* bound to two hydrogens.
Secondary_Amine		A secondary amine is defined as an atom with an atom type of *n3* bound to a single hydrogen.