# Molecular Architect (MolAr) Tutorial

Eduardo Habib Bechelane Maia
Alisson Marques da Silva
Alex Gutterres Taranto




# MolAr Tutorial

# Figures

# Summary

# 1. Install MolAr

One of the great difficulties of using the various existing programs that assist VS is its installation. Often, in addition to installing the program, it is necessary to install libraries used by them. The necessity for these libraries is often not reported in the program manual and the user discovers their need only after performing an extensive internet search for the solution of an error in the program.

In order to overcome this problem, it is intended to make the software installation process simpler for the user. Table 1 and

Table **2** shows the programs and libraries installed automatically by MolAr and why it is needed.

**Table 1 – Software programs installed**

| Program | Necessity |
|---|---|
| Open JDK | Package containing the necessary infrastructure for the development of Java applications. |
| Mopac | Refinement of ligands. |
| MODELLER | Homology Modeling. |
| Pip | Python package manager on Linux. |
| Procheck | Ramachandran Plot. |
| Pdfunite | Join pdf files. |
| Pymol | View a 3D molecule and calculate RMSD. |
| Jmol | View a 3D Molecule. |
| Pdb2pqr | Python package that contains PROPKA, which is used in the protonation state adjustment. |
| MPI | It is used to carry out DOCK 6 in parallel. |
| DOCK 6 | VS with DOCK 6. |
| Sphgen | It is used in DOCK6 to generate sets of overlapping spheres to describe the shape of a molecular surface. |
| Autodock Vina 1.1.2 | Virtual screening with Autodock Vina 1.1.2. |

| | |
|---|---|
| DMS | It is used by DOCK 6 to compute the molecular surface of a molecule. |
| Yad | It is a tool used by MolAr to create graphical dialogs from bash scripts. |
| Autodocktools | It is used in Autodock Vina. |
| Chimera | It is used to configure the ligand for docking. |
| Ambertools | Used to run DOCK 6 with amber force field |
| Evince | It is used to view pdf files. |
| Openbabel | It is used to convert one chemical file format to a different one. |
| mgltools | It is installed to allow use of the adt and Autodock Vina. |

**Table 2 - Libraries installed**

| Library | Necessity |
|---|---|
| biopython | Python scripts |
| python-dev | Python scripts |
| python-matplotlib | Python scripts to generate ROC curve |
| python-sklearn | Python scripts to generate ROC curve |
| libgfortran3 | Procheck |
| numpy 1.8.2 | Autodocktools |
| libgl1-mesa-dev | Autodocktools |
| mesa-common-dev | Autodocktools |
| libstdc++5:i386 | DS Visualizer |
| libstdc++5 | DS Visualizer |

The installation of MolAr automatically installs all the software and libraries used in the whole Virtual Screening process with few user interventions.

So, installing MolAr is very simple (there is a YouTube video of the steps below in the link https://www.youtube.com/watch?v=0npBw-co1TM :

To install MolAr, follow the following steps:

I. Download MolAr in the link http://www.drugdiscovery.com.br/software/;

II. Extract the file downloaded;

III. If MolAr is already installed on the computer to be used, uninstall it by typing the command **./uninstall.sh** into the installation folder extracted in the previous step.

IV. After uninstalling, configure the installation by entering the command **./config.sh**, which will install the most current version of JAVA. Next, config command will show a screen asking for MODELLER [1] and MOPAC [2] license (Figure 1) number .

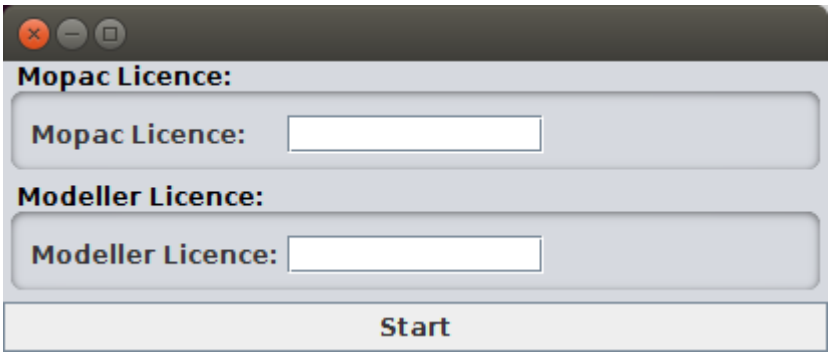



**Figure 1 - Config Screen**

V. Enter these licenses;

VI. Then enter the command ./install.sh. The installation takes a while because several programs and libraries are installed (Table 1 and

VII. Table **2**). It is necessary to follow the installation and confirm the installation of each program;

VIII. After executing the above commands just run the molar command from anywhere at the command prompt. This will open MolAr software (Figure 2);

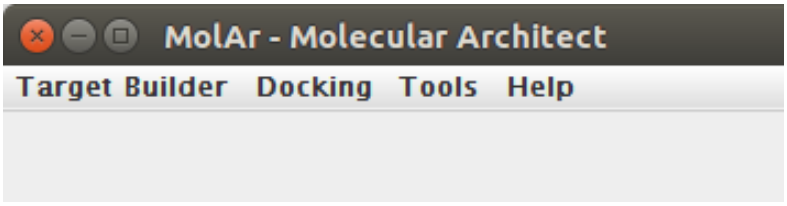



**Figure 2 - MolAr main screen**

IX. You can also access MolAr using the icon that was created on the Desktop;

X. However, in Ubuntu 18, it is necessary an additional step to access MolAr via the Desktop icon. After installation the icon on Figure 3 will appear on the desktop:


**Figure 3 - MolAr desktop windows after installation on Ubuntu 18**

XI. Double-click the icon and the message in Figure 4 will be displayed:

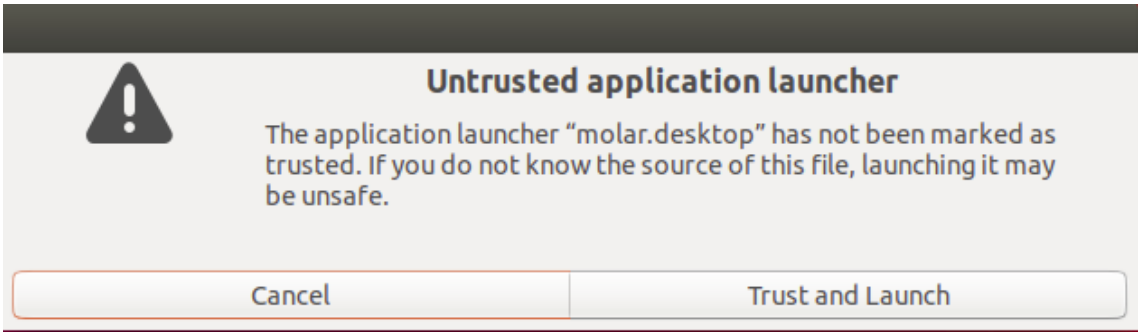



**Figure 4 - Trust and Launch message**

XII. Just click on Trust and Launch and the icon will change to MolAr icon (Figure 5);


**Figure 5 - MolAr Icon after the step above**

XIII. So, to start MolAr just click on the icon on Figure 5.

## 2. Building a Model by Homology Modeling

In Virtual Screening, it is essential the availability of the 3D structure of the protein. However, sometimes it is not possible to obtain the target protein experimentally. Frequently, although there is availability of the 3D structure, the atomic coordinate of high flexible loops can be poorly described by experimental methods. Thereby, regions of gaps are formed, and such regions may be close to the binding site. In another situations, researchers have the amino acid sequence of the target protein, but do not have its 3D structure. So, to use the target in the

VS in a reliable way, the 3D structure must be determined as completely as possible, avoiding the existence of gaps.

The 3D structure of a target protein can be predicted from its amino acid sequence using a methodology called homology modeling.

There are two main situations in which it is necessary to perform homology modeling to generate a good 3D model of the target protein.

a) Generation of the 3D structure from an amino acid sequence.
b) Reconstruction of gap regions in the target protein.

In the next subsections, we will demonstrate examples of how to generate a model by homology modeling in these situations using the MolAr.

## 2.1 Generation of the 3D structure from an amino acid sequence.

a) We will use as an example the 1H00 protein (https://www.rcsb.org/structure/1H00). In the case of this protein, it would be best to model only the gaps regions, but we will use its amino acid sequence to reconstruct it and then compare it to a modeling where only gap regions are modeled;
b) The crystallographic model to be used in this type of modeling should preferably have low resolution (<2 Å) and have a ligand in its structure (the ligand will aid to determine the location of the active site);
c) Copy the FASTA sequence of the 1H00 protein in the PDB database;
d) Open MolAr;
e) Click the Target Menu and then the Homology Modeling submenu;
f) MolAr will open the screen in Figure 6;
g) Figure 6 shows the main interface of homology modeling feature. It is required to the user to fill in one of the following fields: PDB code, PDB File or FASTA sequence of the structure (Figure 6a). In Figure 6b user can inform templates to be used, the resolution of the PDB file (used to generate Ramachandran plot), the number of models that will be generated and the number of refinement loops to be made during the process.

See MODELLER online manual[1] for details. It is noteworthy that the use of several models generally decreases the quality of the generated model [3]. Moreover, user must select witch modeling method will be used. The possible methods are, in order of optimization: very_fast, fast, slow, very_slow, slow_large. The name of the methods indicates that the more we want to optimize, the slower the modeling will be. It is possible to do the loop refinement in each model. So, in this case, user must inform the amount of refinement loops as well as the refinement method, which follows the same nomenclature and meaning of the modeling methods explained above. Finally, user can also indicate if they want to generate the new model with heteroatoms and hydrogen (Figure 6c). If user select the option “Select Best Model Automatically”, MolAr will select the best model based only in DOPE energy.

h) Click on the button “Advanced options”;

i) Paste the fasta sequence below in the “Inform Fasta Sequence” field:

XMENFQKVEKIGEGTYGVVYKARNKLTGEVVALKKIRLDTETEGVPSTAIREISLLKELNHPNIVKLLDVIHTENKLYLV
FEFLHQDLKKFMDASALTGIPLPLIKSYLFQLLQGLAFCHSHRVLHRDLKPQNLLINTEGAIKLADFGLARAFGVPVRTY
THEVVTLWYRAPEILLGCKYYSTAVDIWSLGCIFAEMVTRRALFPGDSEIDQLFRIFRTLGTPDEVVWPGVTSMPDYKPS
FPKWARQDFSKVVPPLDEDGRSLLSQMLHYDPNKRISAKAALAHPFFQDVTKPVPHLRL

j) If you want to use pre-selected templates, just enter their PDB code, separated by commas in the "Enter the PDB code of the template" field. In this example, no template will be used and MolAr itself will search for templates and display them to users for selection;

k) We will generate 5 models in this example modeling. To do so, simply fill in the "Number of Models" field. If no value is placed, MolAr will generate 20 models and the user should choose the best one;

[1] https://salilab.org/modeller/manual/

(a)

(b)

(c)

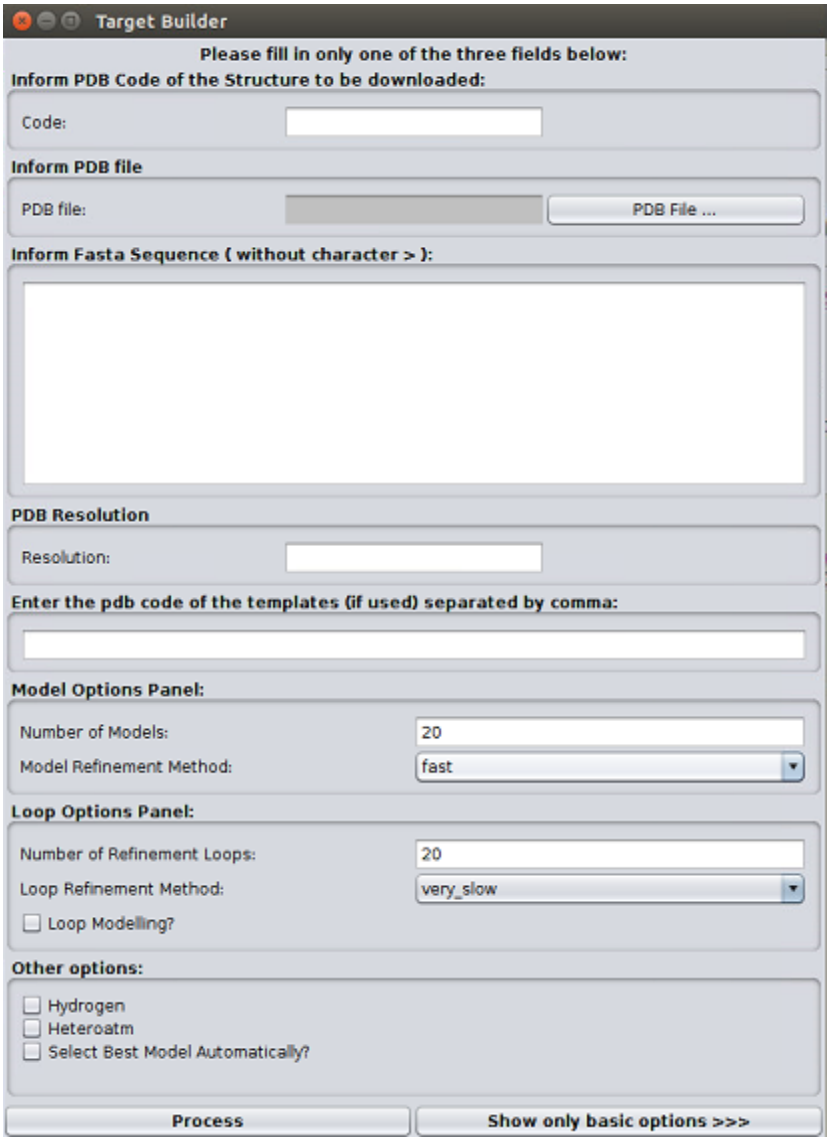



**Figure 6: (a) Sequence Data; (b) Simulation Parameters; (c) Other Options**

l) The screen with the filled options can be verified in Figure 7;

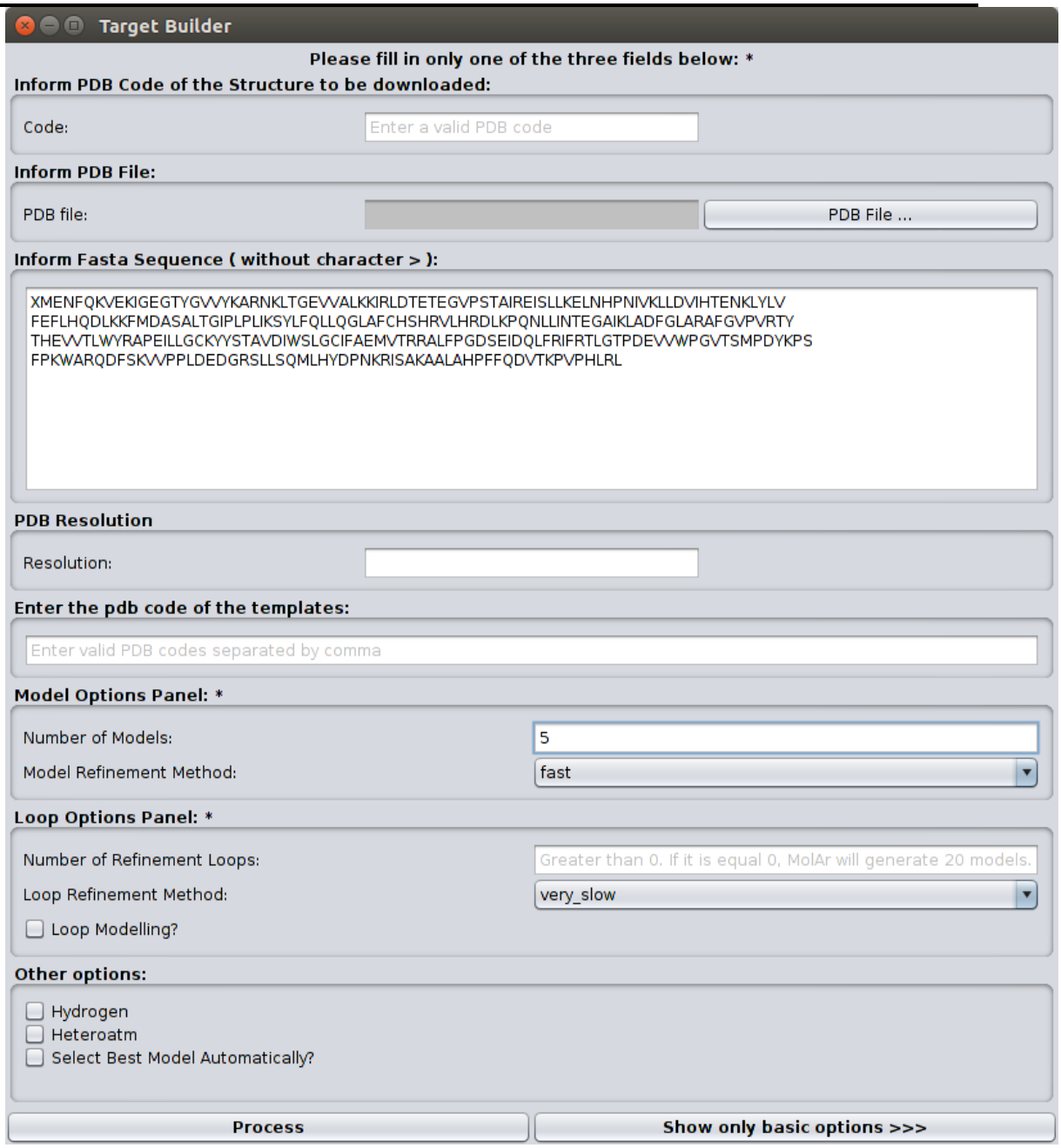



**Figure 7 - Homology Modeling after fill all the required fields**

m) Press the Process button;

n) After a processing time, MolAr will display some suggested templates, as in Figure 8;

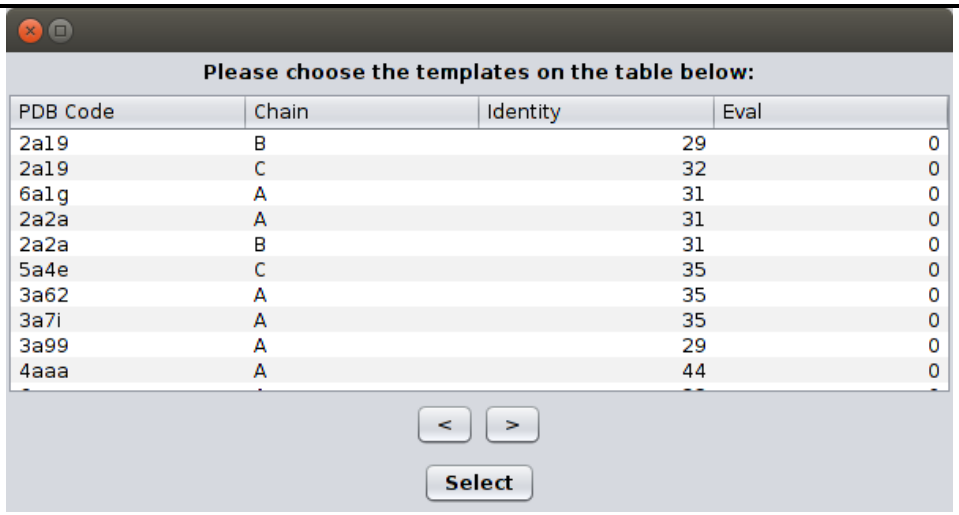



**Figure 8 - Select template screen**

o) To facilitate choosing a better template, we will click on the Identity column so that the templates are sorted by identity Figure 9;

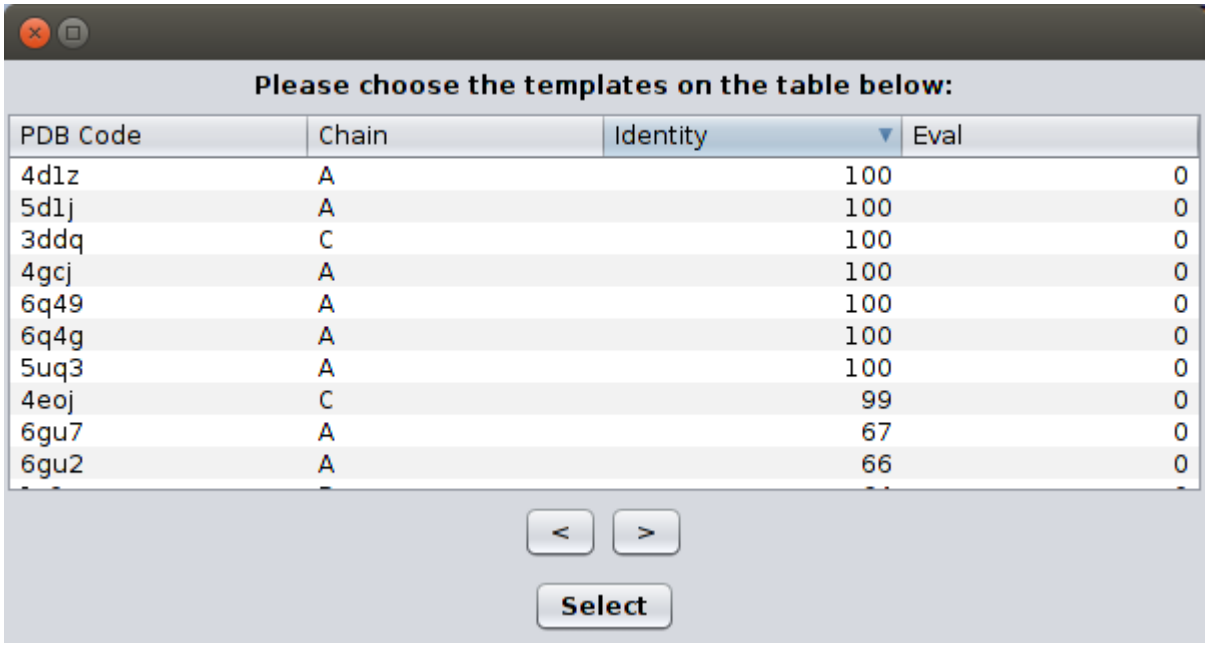



**Figure 9 - Select template screen after sort**

p) Using the Ctrl key and the mouse, we will select the 3 best templates for this modeling Figure 10;

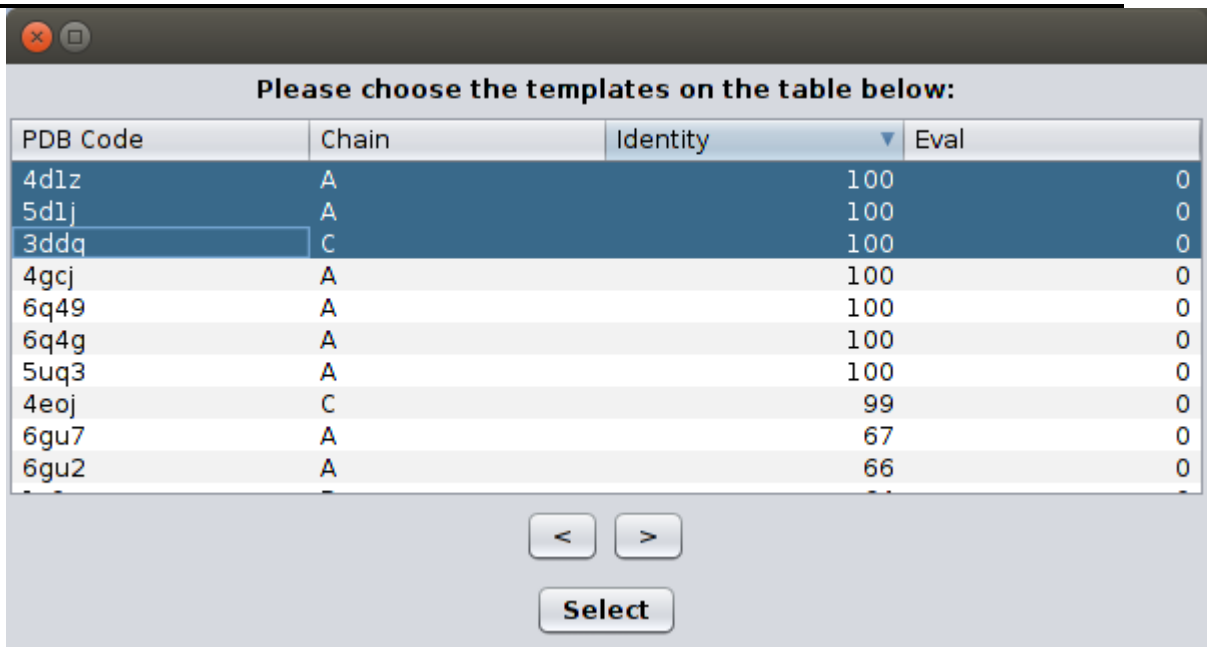



**Figure 10 - Choose the templates screen after selection**

q) Press the select Button;

r) After a processing time, MolAr will display the generated models, ordered by DOPE energy Figure 11;

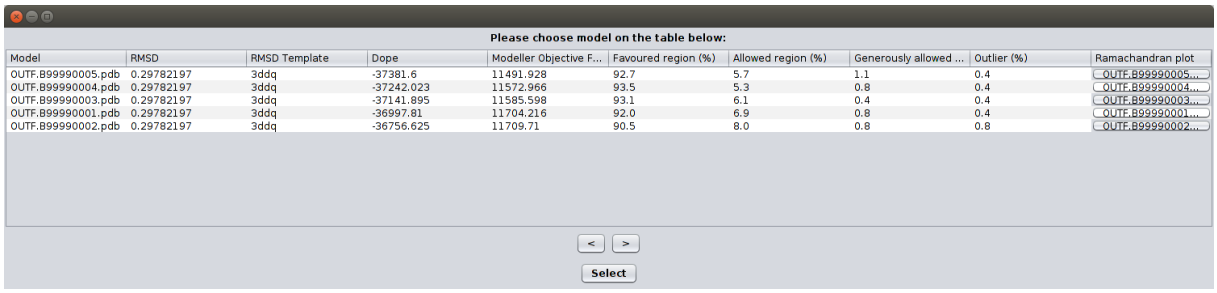



**Figure 11 - Generated models**

s) We will choose the first model, because it is the one with the best DOPE energy. In addition, its RMSD is low (approximately 0.23) and more than 90% of the atoms in the Ramachandran chart are in the more favorable region;

t) Finally, after selecting the first template, we will call the select command and save it to a directory (Figure 12);

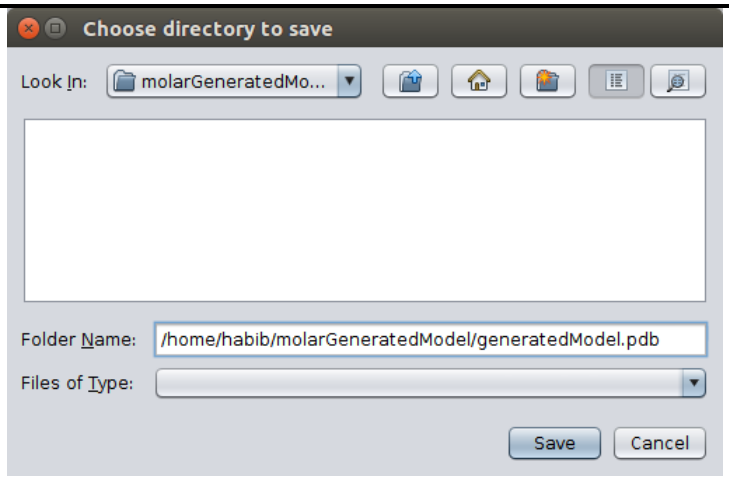



**Figure 12 - Save model screen**

u) The Ramachandran plot can be saved by clicking the command in the Ramachandran Plot column of the selected model (Figure 13);

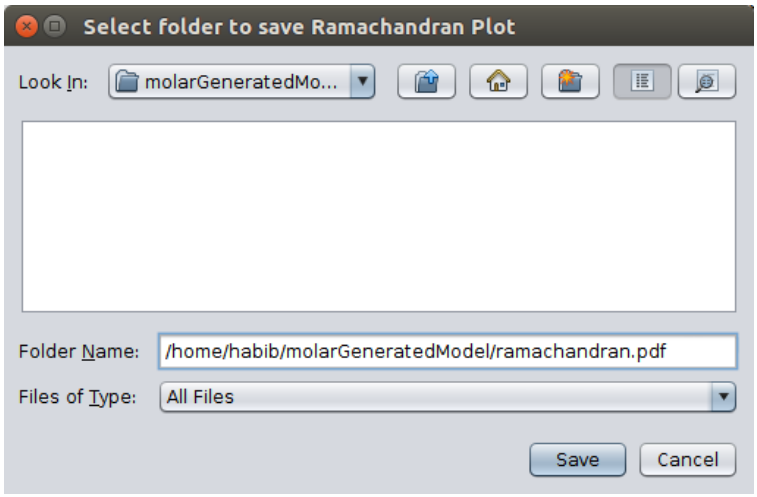
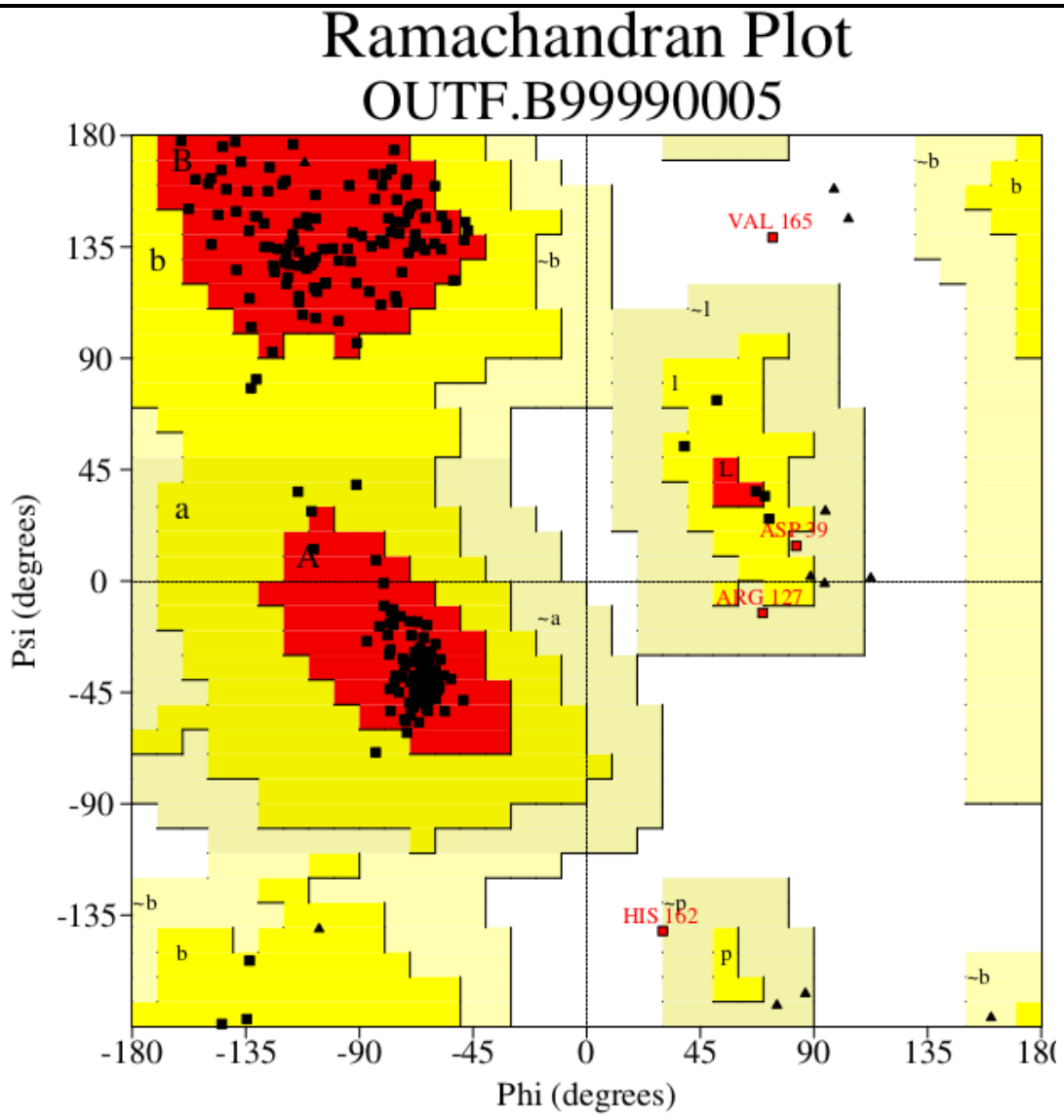



**Figure 13 - Save Ramachandran screen**

v) The generated Ramachandran plot of the selected template can be viewed in Figure 14;




Plot statistics

| | | |
|---|---|---|
| Residues in most favoured regions [A,B,L] | 243 | 92.7% |
| Residues in additional allowed regions [a,b,l,p] | 15 | 5.7% |
| Residues in generously allowed regions [~a,~b,~l,~p] | 3 | 1.1% |
| Residues in disallowed regions | 1 | 0.4% |
| Number of non-glycine and non-proline residues | 262 | 100.0% |
| Number of end-residues (excl. Gly and Pro) | 2 | |
| Number of glycine residues (shown as triangles) | 16 | |
| Number of proline residues | 19 | |
| Total number of residues | 299 | |

Based on an analysis of 118 structures of resolution of at least 2.0 Angstroms and R-factor no greater than 20%, a good quality model would be expected to have over 90% in the most favoured regions.

**Figure 14 - Generated Ramachandran plot**

w) The RMSD of approximately 0.30 is relative to the template in which it achieved the best modeling. In relation to the original protein used in this example (1H00), the RMSD is about 0.51. However, in a typical situation where only the amino acid sequence is present, it is not possible to know the RMSD in relation to the original protein, since it does not exist.

## 2.2 Reconstruction of gap regions in the target protein.

a) Download 1H00 Protein (https://www.rcsb.org/structure/1H00). Open the downloaded PDB file in a text editor. You can check that 20 residues in the crystallized protein were not represented in the 3D structure provided. The PDB file contain the information of which residue is missing (Figure 15).

```
REMARK 465
REMARK 465 MISSING RESIDUES
REMARK 465 THE FOLLOWING RESIDUES WERE NOT LOCATED IN THE
REMARK 465 EXPERIMENT. (M=MODEL NUMBER; RES=RESIDUE NAME; C=CHAIN
REMARK 465 IDENTIFIER; SSSEQ=SEQUENCE NUMBER; I=INSERTION CODE.)
REMARK 465
REMARK 465   M RES C SSSEQI
REMARK 465     GLY A    13
REMARK 465     THR A    14
REMARK 465     ARG A    36
REMARK 465     LEU A    37
REMARK 465     ASP A    38
REMARK 465     THR A    39
REMARK 465     GLU A    40
REMARK 465     THR A    41
REMARK 465     GLU A    42
REMARK 465     GLY A    43
REMARK 465     PHE A   152
REMARK 465     GLY A   153
REMARK 465     VAL A   154
REMARK 465     PRO A   155
REMARK 465     VAL A   156
REMARK 465     ARG A   157
REMARK 465     THR A   158
REMARK 465     TYR A   159
REMARK 465     THR A   160
REMARK 465     HIS A   161
```

**Figure 15 - Missing residues**

b) Thus, when there are gap regions, a MolAr option can be used to build only these regions, without disturbing the rest of the protein structure. To access this option:

   a. Open MolAr;
   b. Click the Target Menu and then the Homology Modeling Missing residues submenu;
   c. MolAr will open the missing gaps screen (Figure 16);

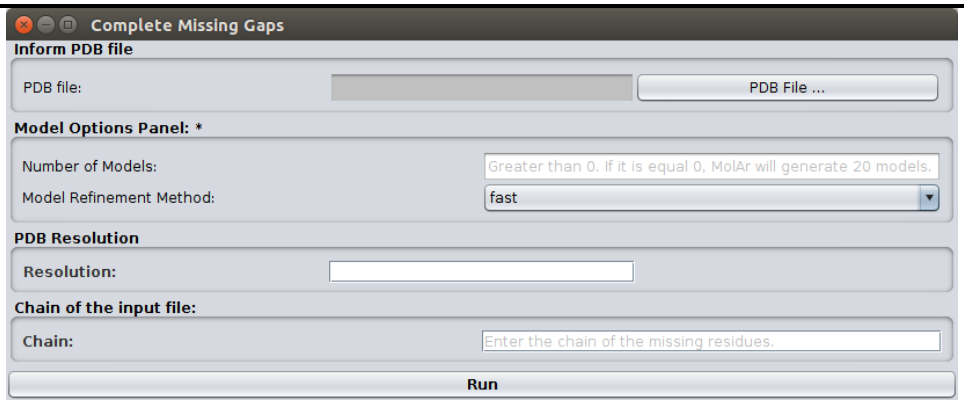



**Figure 16 - Missing gaps screen**

c) So, in this case you will need to fill in the fields: PDB file, number of models, model refinement method, resolution of the PDB file used and chain;

   a. PDB file: PDB file where the gaps will be filled.
   b. Number of models: Number of models to be generated during the modeling.
   c. Model refinement method: refinement method. The slower the method, the more accurate it is.
   d. Resolution of the PDB file: Resolution of the PDB file downloaded.
   e. Chain: chain where the gaps will be filled.

d) The screen with the filled options can be verified in Figure 17.

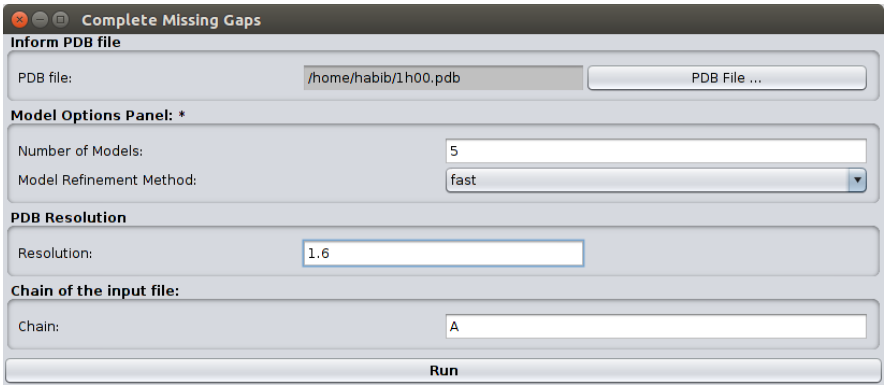



**Figure 17 - Filled missing gaps screen**

e) Press the Run button;

f) After a processing time, MolAr will open the Result screen (Figure 18):

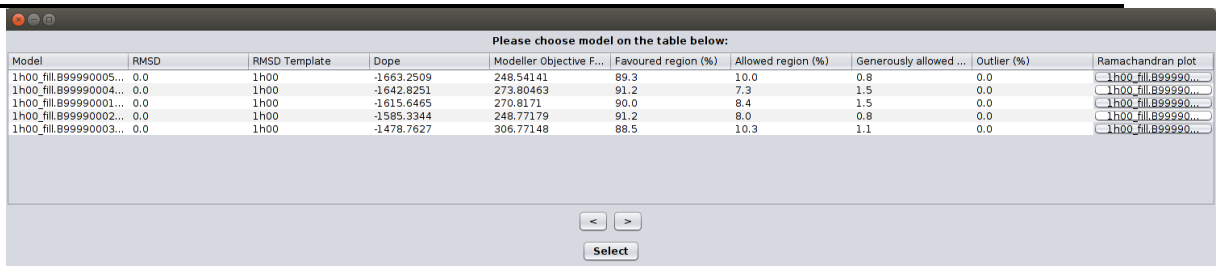



**Figure 18 – Result screen**

g) So, you can choose the best model and click in the select button to save it. We will choose the second model. Although this model doesn't have the lowest DOPE energy (it has the second lowest) it has more than 90% of the atoms in the Ramachandran chart are in the more favorable region. The first model does not meet this requirement. Finally, as we reconstruct only the GAP regions and there are few amino acids in the gap region, the RMSD was very low and since the precision is only 2 decimal places, it was displayed as 0;

h) Finally, after selecting the first template, we will call the select command and save it to a directory (Figure 19);

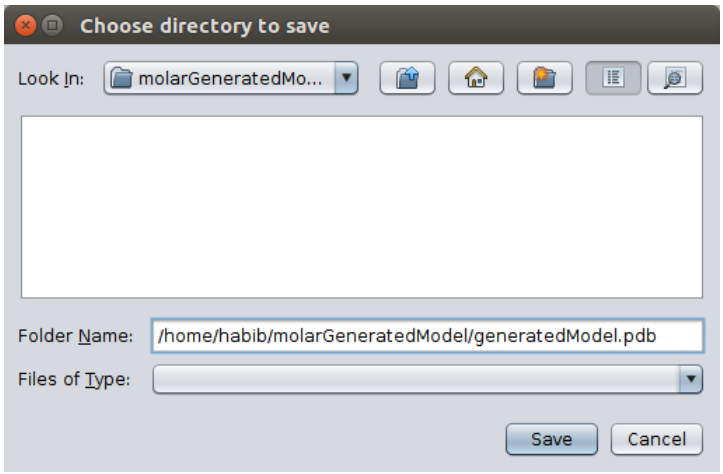



**Figure 19 - Save model screen**

i) The Ramachandran plot can be saved by clicking the command in the Ramachandran Plot column of the selected model (Figure 20);

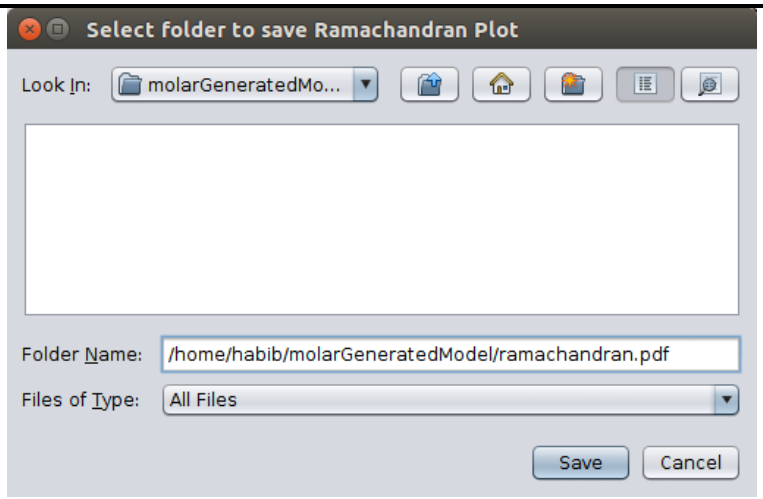
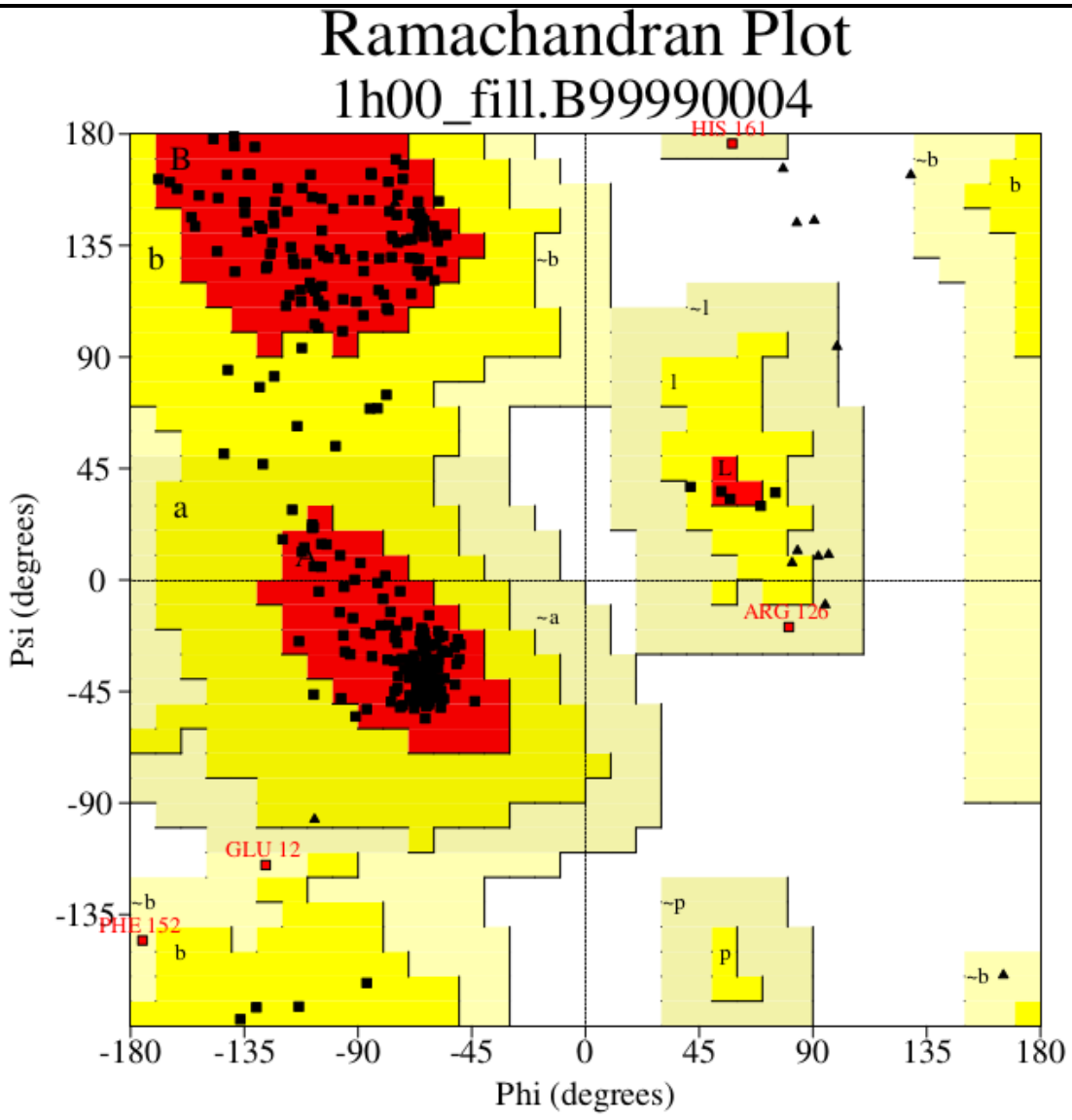



**Figure 20 - Save Ramachandran screen**

j) The generated Ramachandran plot of the selected model can be viewed in Figure 21;




Plot statistics

| | | |
|---|---|---|
| Residues in most favoured regions [A,B,L] | 238 | 91.2% |
| Residues in additional allowed regions [a,b,l,p] | 19 | 7.3% |
| Residues in generously allowed regions [~a,~b,~l,~p] | 4 | 1.5% |
| Residues in disallowed regions | 0 | 0.0% |
| Number of non-glycine and non-proline residues | 261 | 100.0% |
| Number of end-residues (excl. Gly and Pro) | 4 | |
| Number of glycine residues (shown as triangles) | 16 | |
| Number of proline residues | 19 | |
| Total number of residues | 300 | |

Based on an analysis of 118 structures of resolution of at least 2.0 Angstroms and R-factor no greater than 20%, a good quality model would be expected to have over 90% in the most favoured regions.

**Figure 21 - Generated Ramachandran plot**

# 3. Docking

## 3.1 Docking Menu

The docking menu has the features that allow the realization of Molecular Docking and Virtual Screening. The developed platform allows the realization of the Virtual Screening through Autodock Vina ([4]), DOCK 6 [5] or through a Consensus Docking between them.

### 3.1.1 Octopus Submenu

Octopus Submenu performs VS using Autodock Vina [4]. It allows the execution of the four main functionalities (Figure 22), which are: Virtual Screening with Mopac, Virtual Screening without Mopac, Run Mopac and Database Manager.

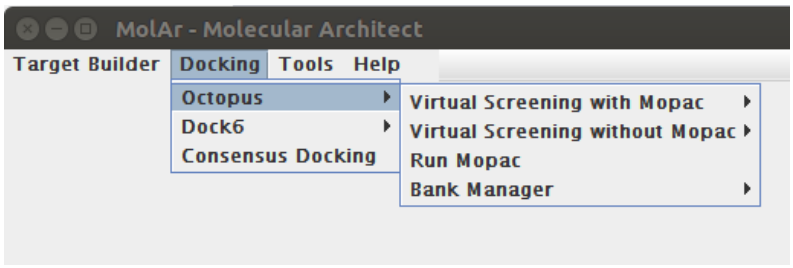



**Figure 22 - Octopus Menu**

An overview of the Octopus workflow can be seen in Figure 23. First, directories of ligands and targets are chosen. Ligands must be in the PDB format and the files in targets directory must be in the Autodock Vina format. The target database has a configuration file with X, Y and Z coordinates, a grid box size delimiting the region for molecular docking simulations and the reference binding energy according to the re-docking of crystallographic ligand. These files are explained in detail in section (section 3.1.1.2). If user choose to refine ligands, MolAr will perform the refinement using the Run MOPAC [2] software (explained in section 3.1.1.1). Next, ligands are converted from PDB to PDBQT file format while assigning the rotatable bonds, the Gasteiger-Marsili net atomic charges [6] and only the hydrogens on polar atoms (oxygen and

nitrogen) are kept, while other hydrogens atoms are removed. Then, visual inspection of the geometries of the ligands can be performed using PyMOL [7]. In the next step, the docking is performed using the Autodock Vina, which runs until all the ligands have been docked on a set of targets. Finally, the binding energy results for the complex target/ligand are generated. The standard crystallographic values for the binding energies between ligands and targets are also displayed. In the next subsections, we will carry out an example in MolAr.

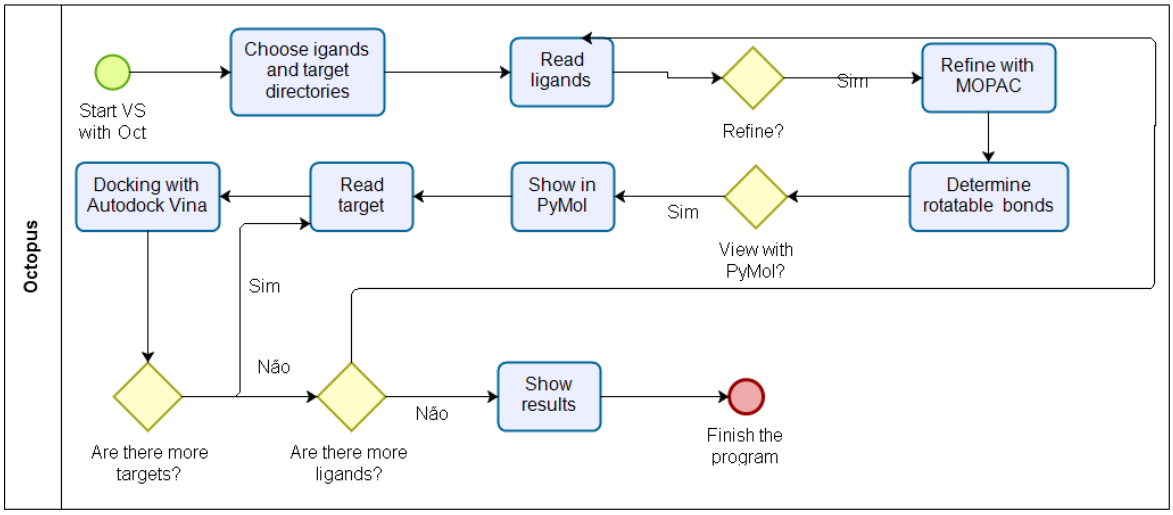



**Figure 23 - Octopus Workflow**

### 3.1.1.1 RUNMOPAC

RUNMOPAC is a Python software developed by our group and registered in INPI. MolAr integrates RUNMOPAC software within the Octopus submenu. This software calculates the net atomic charges for each atom in each molecule avoiding a massive work by the user. RUNMOPAC refine the ligands, in PDB file format, through MOPAC2016 [2] using the Parametric Method 7 (PM7) [8] and EF routine [9] to search for the structure of local minimum.

There is an example of how to refine a ligand using MolAr in the link:

https://www.youtube.com/watch?v=PiMh_PVzHZE&feature=youtu.be

### 3.1.1.2 Virtual Screening with Mopac and Without Mopac

Octopus performs the VS process using Autodock Vina in 2 different forms, depending on which menu option was chosen (Figure 22), which are: with a previous execution of Mopac (choosing Virtual Screening with Mopac) or without running MOPAC (choosing Virtual Screening without Mopac). For both the Screen displayed is the same as in Figure 24. The only difference is whether Mopac will be carry out.

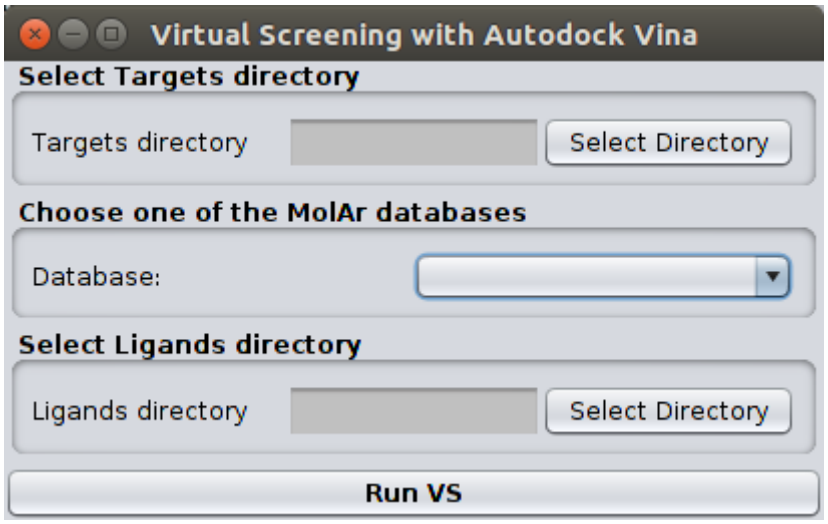



**Figure 24 - Octopus main screen**

In the main Octopus screen, you can select the targets database where the VS should be performed and the ligands directory. The ligands directory must have all ligands to be used in VS, in the PDB format.

The target database, on the other hand, must be in the PDBQT format, which is used by Autodock Vina, and it is necessary to place the configuration files with the necessary data. The platform developed comes with two previously registered databases that are the Our Own Molecular target (OOMT) ([10]) and the Brazilian Malaria Molecular Target (BRAMMT). The OOMT database comprises various receptors from the Protein Data Bank (PDB), and it includes specific targets for cancer, dengue, and malaria. The BRAMMT database comprise receptors for *Plasmodium falciparum*. To illustrate how the target database should be, we will use the OOMT database structure as an example.

The OOMT database has 42 targets and each one is placed in a specific directory, identified by the PDB code of the target protein and a file containing the reference binding energy values between each target and its crystallographic ligand (Figure 25).

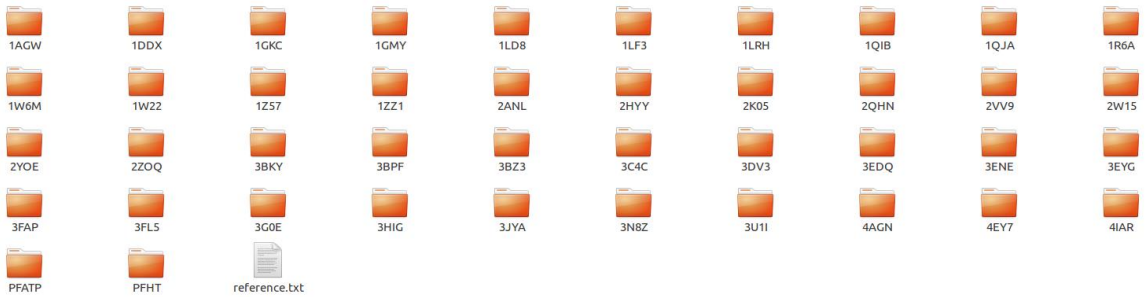



**Figure 25 - OOMT database**

Within each of the target directories, there should initially be 2 files. The target protein file in the PDBQT format and the conf file. Figure 26 shows an example for the 1AGW target protein (Crystal structure of the tyrosine kinase domain of fibroblast growth factor receptor 1 in complex with su4984 inhibitor). The 1AGW.pdbqt file has the 1AGW protein in the PDBQT format.

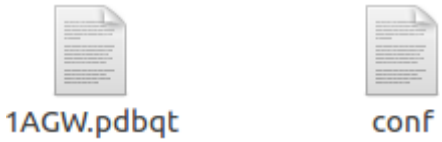



**Figure 26 – example directory for 1AGW Target**

The conf file has the information about the binding site. Figure 27 shows the conf file used for the 1AGW protein. It is important to fill in only the coordinates for the center of the binding site (center_x, center_y and center_z) and the dimensions of the box to be used (size_x, size_y, and size_z).

```
exhaustiveness = 24

center_x = 9.877
center_y = 3.592
center_z = 23.95

size_x = 20
size_y = 20
size_z = 20
```

**Figure 27 - conf file for 1AGW protein**

Below, we will use the 1H00 protein to demonstrate how to determine the binding site information using adt, which is installed with MolAr.

The steps to determine the binding site information are:

- First, it is necessary to save the protein / ligand complex. Let's save it with the name 1H00_complex.pdb. So, we must reconstruct the GAPS regions according to section 1.2, delete the water molecules of the 1H00_complex.pdb file and leave only the target and the crystallographic ligand, which in this case we will use the ligand FCP.
- Then we must separate the ligand and the target into 2 different files. Let's save the ligand in a file named FCP.pdb and the protein in file 1H00.pdb.
    - Add the polar hydrogens to the ligand using some tool like Discovery Studio or Chimera. In Chimera, this must be done in 2 steps. So, click Tools, General Controls, Command Line. The Chimera command prompt will appear at the bottom of the Screen.

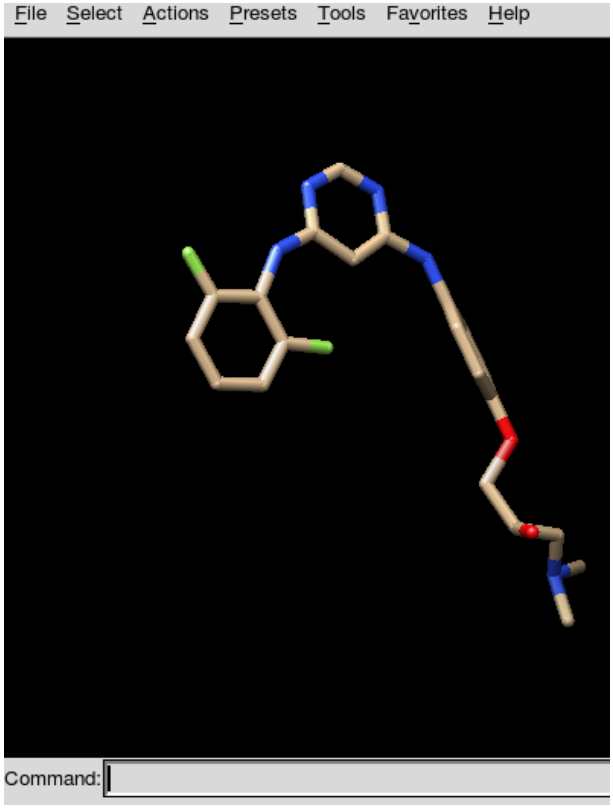



**Figure 28 - Chimera screen before adding polar hydrogens**

- Then enter “addh” to add all the hydrogens to the ligand and press ENTER.
- Next, type “delete HC” to delete the non-polar hydrogens and press ENTER. The ligand will look like the image below. Figure 29 shows the ligands after adding the polar hydrogens.

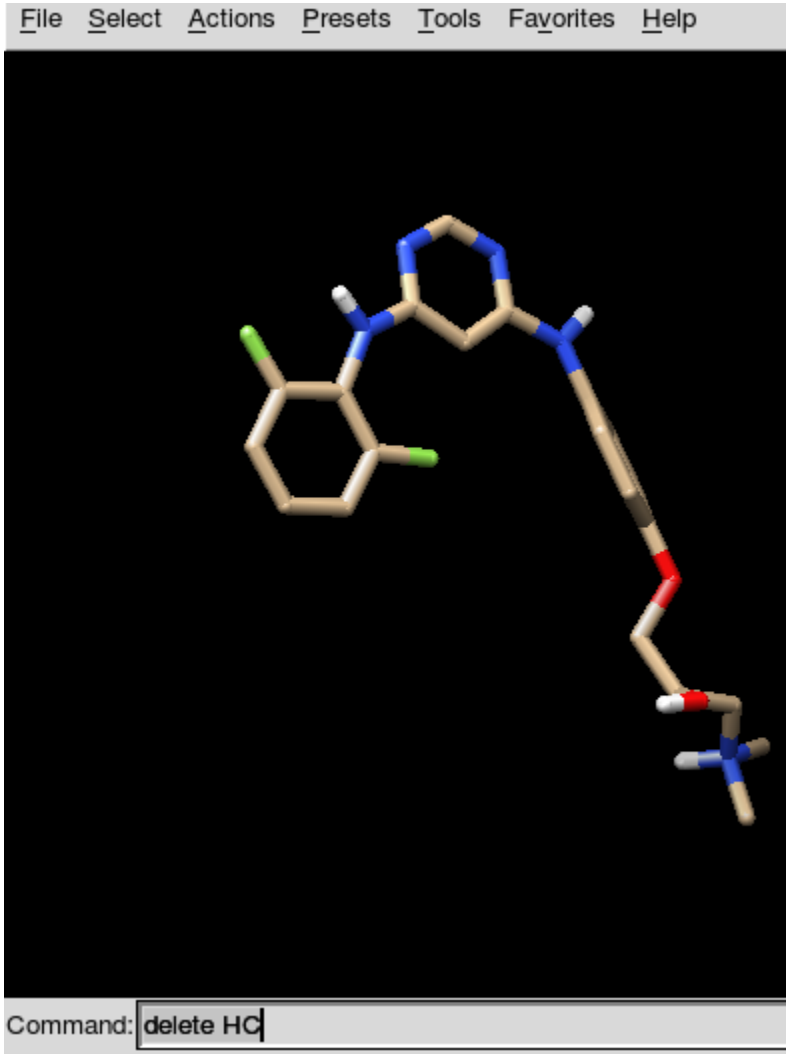



**Figure 29 - Chimera after adding the polar hydrogens**

Next, it is necessary to determine the position of the binding site. Therefore:

- Open ADT by typing adt at the Linux command prompt.
- Open the ligand in ADT by clicking ligand -> Input -> Open.
- Click on Ligand -> Torsion Tree -> Choose Torsions and then, press the done button.

- Save the ligand in PDBQT format by clicking in Ligand -> Output -> Save as PDBQT
- Open the target in ADT by clicking in Grid -> Macromolecule -> Open
- The ADT will save the target in the PDBQT format. Confirm the save.
- Uncheck the target in the ADT so that only the ligand is displayed (Figure 30).

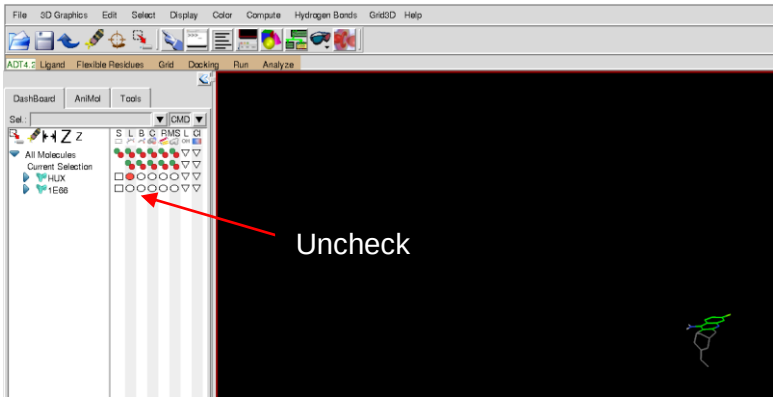



**Figure 30 - Unchecking target**

- Next, the position of the center of the Ligand must be determined. To do so, click on Grid -> Set map type -> Choose Ligand and choose the ligand FCP (Figure 31)


**Figure 31 - Choose ligand screen**

- Click on Grid -> Gridbox and ADT will open the screen in Figure 32.

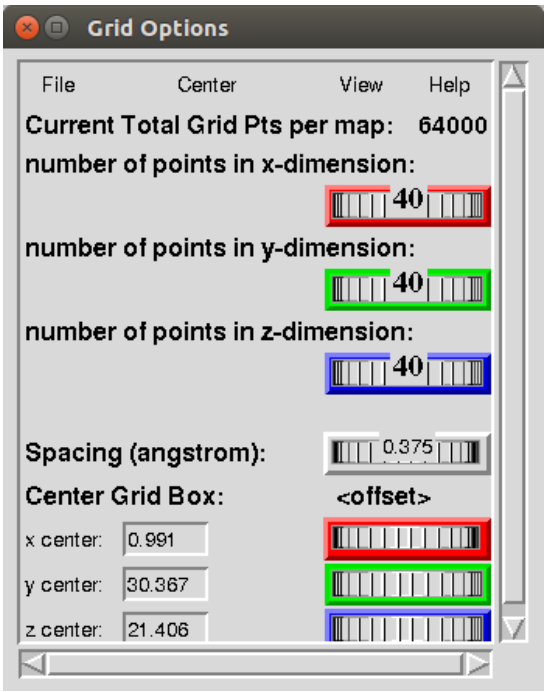



**Figure 32 - Grid Options Screen**

- Then Press Center -> Center on Ligand (Figure 33)

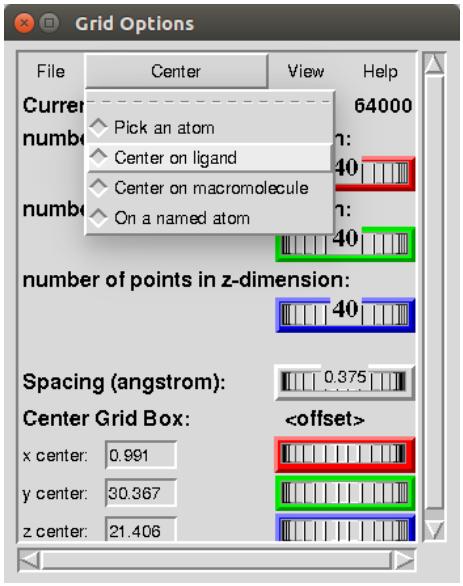



**Figure 33 - Center on ligand**

Finally, with the X, Y and Z coordinates we can create da database using MolAr.

### 3.1.1.3 Database Manager

The Database Manager tool (Figure 34) allows to manage the Octopus database. With this functionality it is possible to the user create a new database and to verify whether targets databases used by Octopus are corrected. Targets databases are constantly changing, since new molecules are inserted or modified continually. If the database updates do not follow a standard, VS could fail, and a precious time is spent trying to identify and correct the problem. In order to solve this problem, Database Manager was developed. It is a feature that allows the creation of the databases to be used by Octopus in the correct format and allows the correction of inconsistencies in the databases used by Octopus. The tool has 3 basic functions that allow the creation of a new database (Figure 34(a) and Figure 34(b)), to fix problems in the database (Figure 34(c)) or to edit the data stored in an existing database (Figure 34(d)).


**Figure 34 – Database Manager Screen: (a) and (b) Create database Screen; (c) Fix database screen; (d) Edit database screen**

The steps below will show how to create a new database using MolAr.

- First, copy the generated PDBQT file of the target to a new folder. If you have more than one target copy all the targets to the same folder.
- Then, in MolAr, choose the Ban Manager option in Docking->Octopus->Database Manager->Create a new database.
- MolAr will open the screen in Figure 35

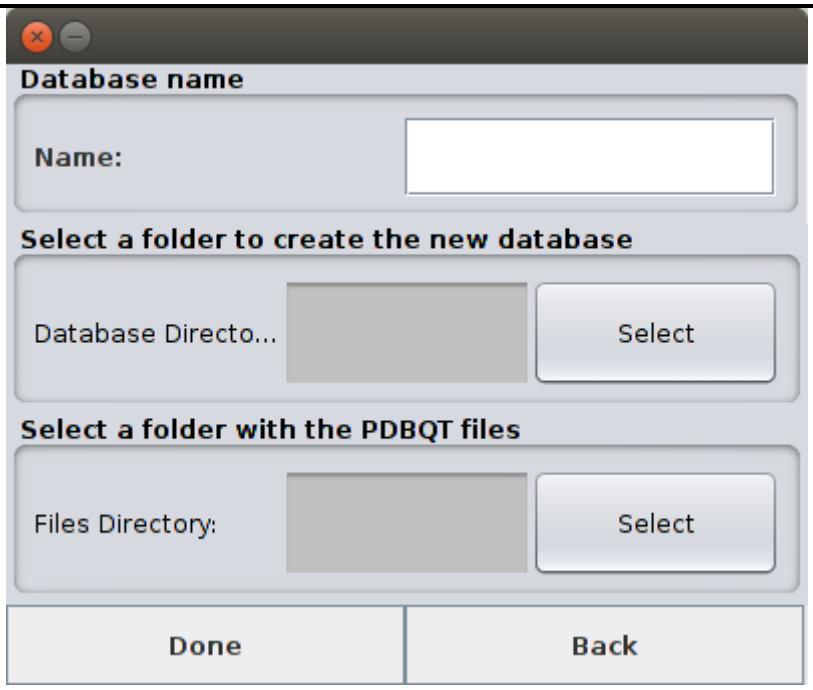



**Figure 35 - Create a new Database screen**

- Fill the Name field Figure 35 with the name of the database being created.
- Select a folder where the database will be created.
- Select the folder containing the target PDBQT files. In our example, only the PDBQT of the target 1H00 were saved.
- The Figure 36 shows the filled home screen of the database manager.

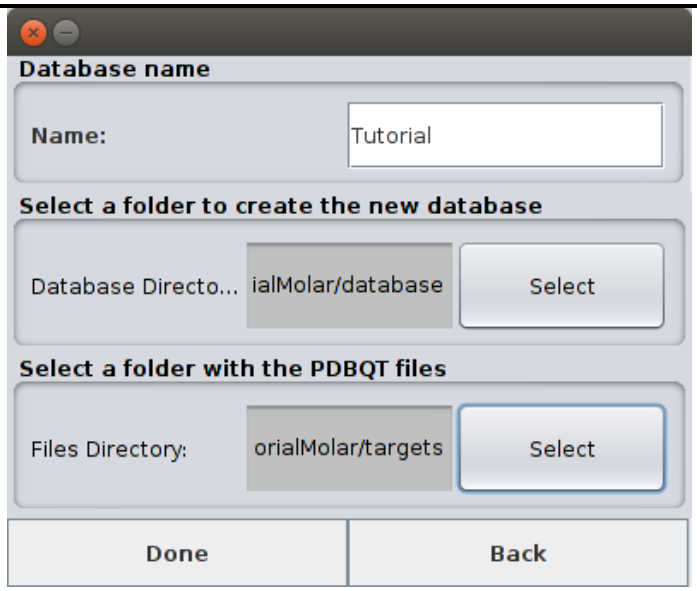



**Figure 36 - Filled home screen of database manager**

- Next, press the Done Button.
- The, in the next screen fill the X, Y and Z coordinates of the center of the ligand (Figure 33).
- The X, Y, and Z positions may be slightly different from those in this example, depending on how the entire process was done.

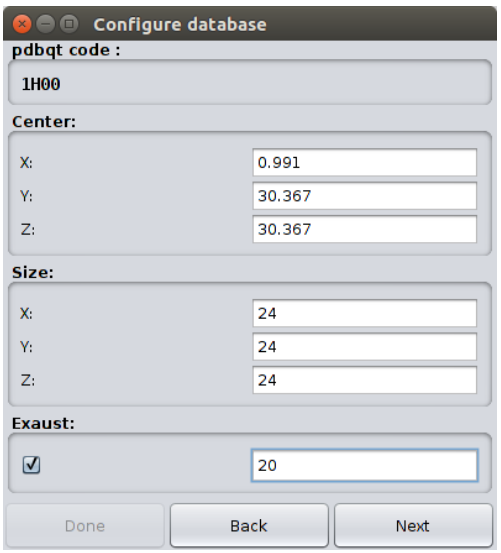



**Figure 37 - configure database screen**

- If there were more targets in our database, the fields should be populated for all targets. As there are no more targets, the Done button becomes enabled. So just press it.

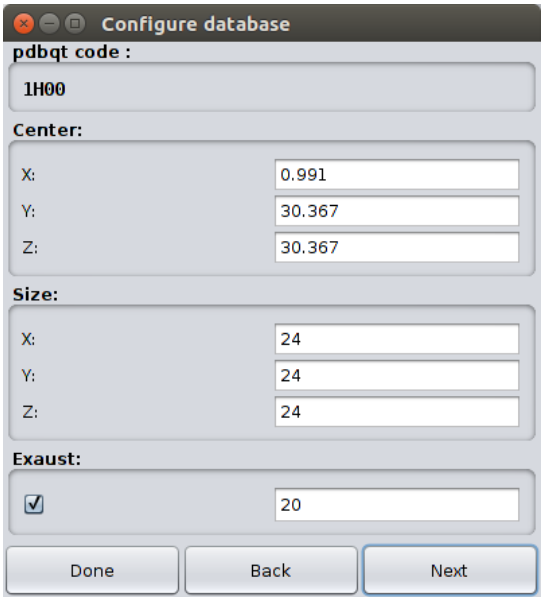



**Figure 38 - Configure database screen with the Done button enabled**

- So, the database was created on the selected folder.
- In each target directory two files are created. The target PDBQT file and the conf file containing the binding site settings completed in the previous steps (Figure 39).

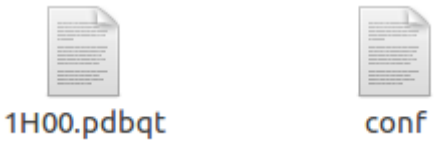



**Figure 39 - Content of the 1H00 folder**

### 3.1.2 DOCK 6

The DOCK 6 program was created in the 1980s by Irwin and Kuntz's Group of Pharmaceutical Chemistry Laboratory of University of California. It was the first docking program [11] [5]. Even though DOCK 6 is a powerful program with multiple usage options, it does not have a graphical interface.

Thus, MolAr implements a graphical interface to access the main features of DOCK 6. Before creating this interface, to do a VS or a simple redocking using DOCK

6, it was necessary to use several scripts, modify several configuration files and execute several programs manually.

The graphical interfaces developed allow the use of DOCK 6 to execute a redock or to perform a Virtual Screening. Figure 40 illustrates the execution of DOCK 6.

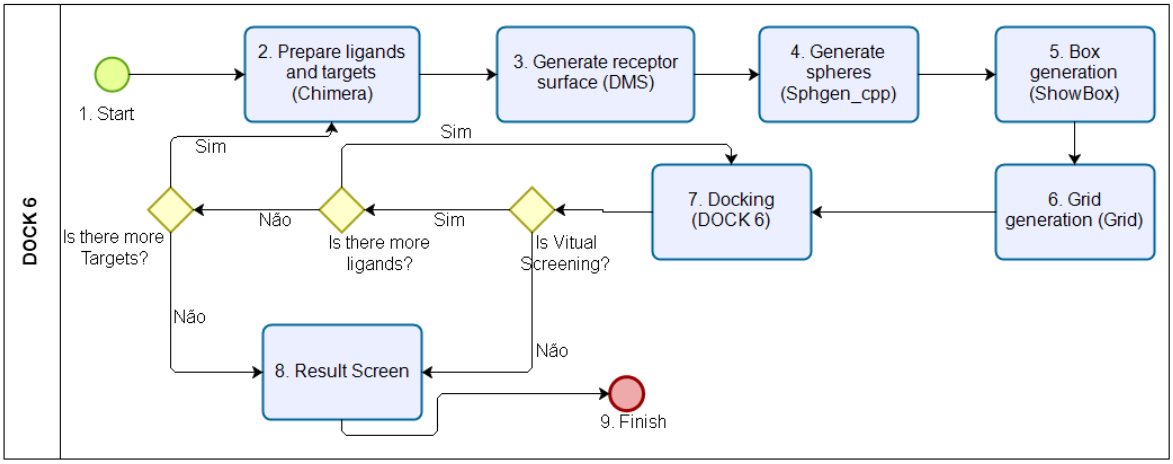



**Figure 40 - DOCK 6 Workflow**

### 3.1.2.1 Redocking with DOCK 6

Redocking aims to restore the initial position of a ligand in a crystallographic structure of a ligand/target complex. Usually redocking is performed to verify if the docking parameters for the program used can predict the structure and interactions of a known complex.

Figure 41 shows the MolAr Redock Screen using DOCK 6. As can be observed, for the default redocking it is necessary to inform the PDB file containing the protein-ligand complex which the redock will be performed and the PDB code of the ligand to be considered during the redocking process.

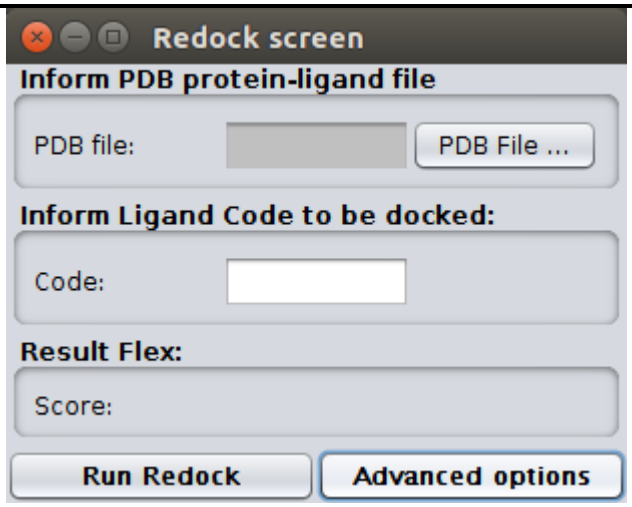



**Figure 41 - Redock Basic Screen**

We will perform the redocking using 1H00 target used in Octopus:

- Click in the Advanced Option Button. MolAr will show the advanced Redock screen (Figure 42).

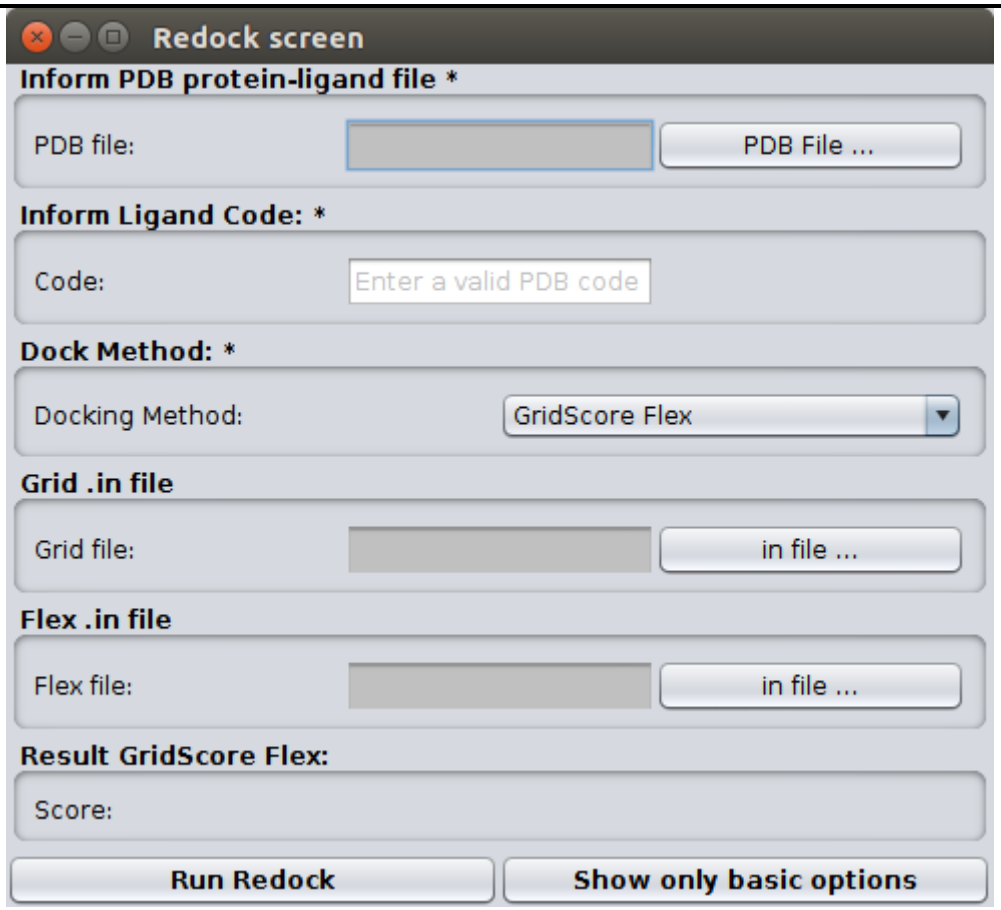



**Figure 42 – Advanced Redock Screen**

- Select the target / ligand complex for the 1H00 protein.
- Put the ligand code (FCP) in the correct field;
- Select Gridscore Flex in the docking method
- Click the Run Redock button.
- After a processing time (Figure 43), MolAr will show the redock energy (Figure 44)

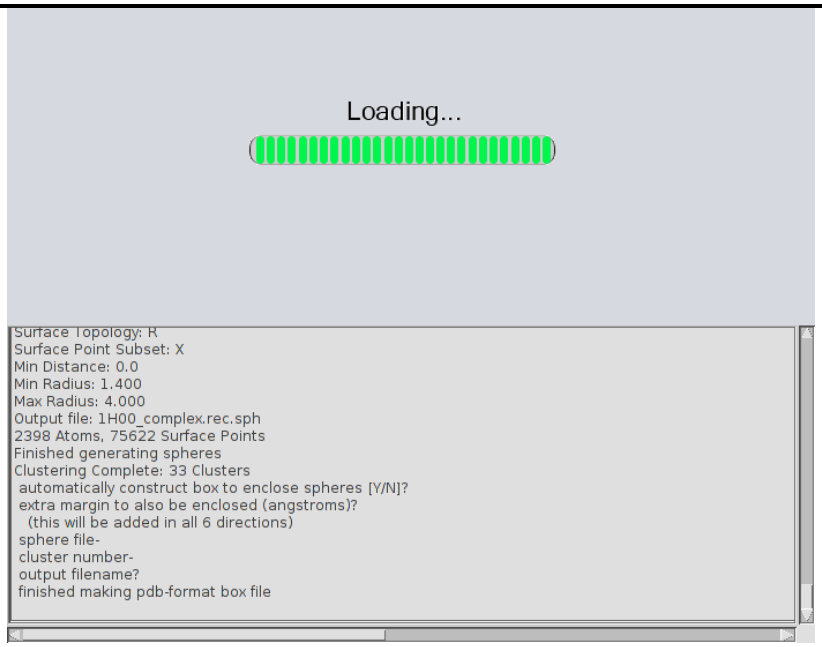



**Figure 43 - Loading screen**

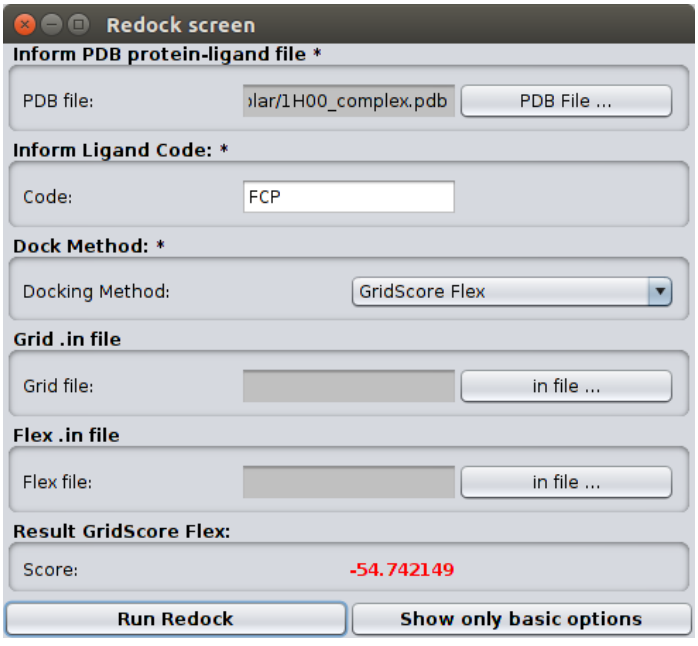



**Figure 44 - Redock result**

Docking can be accomplished using any set of configurations allowed by dock6. If you want any different configuration, simply enter the desired configuration files in the fields for grid.in and flex.in.

### 3.1.2.2 Virtual Screening with DOCK 6

VS can identify the most promising compounds for biological assays and decrease the costs associated with drug development [12]. Figure 45 shows the VS screen using DOCK 6. It is necessary to inform only the target/ligand complex directory and the ligands directory that will be used in the VS process.

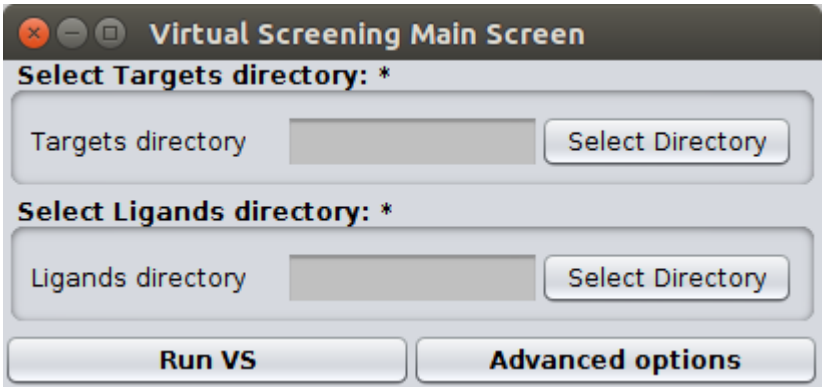



**Figure 45 - Virtual Screening using DOCK 6**

If user press the Advanced options button, MolAr will allow the docking method to be changed (Figure 46). Gridscore flex is the default if user does not change this option.

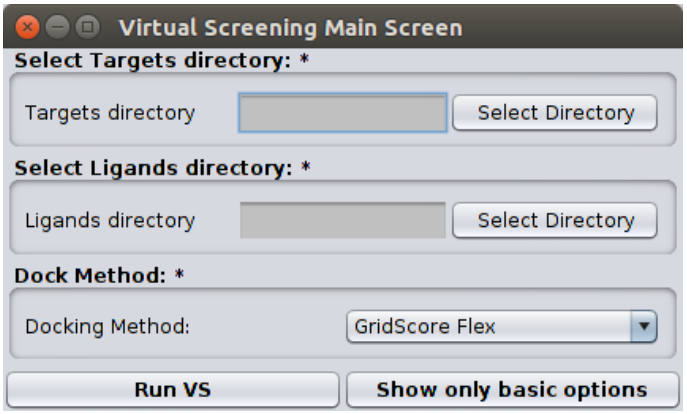



**Figure 46 - Virtual Screening main screen after pressing the Advanced options button**

We will perform a DOCK6 VS using some 1H00 ligands selected from Dud ([13]):

- First, put the target / ligand complex in a separated folder.
- Rename the target file to the 1H00.pdb, just to make the result cleaner.

- Select the folder in the field Targets Folder.
- Select the folder with the Dud ligands in the Ligands folder field.
- Figure 47 shows an example of the VS with DOCK 6 screen with the filled fields. In this example, we use the GridScore flex with amber docking method. This choice promises better results, as it performs some molecular dynamics simulations during docking with DOCK 6, but it leads to a much longer execution time than GridScore flex without amber

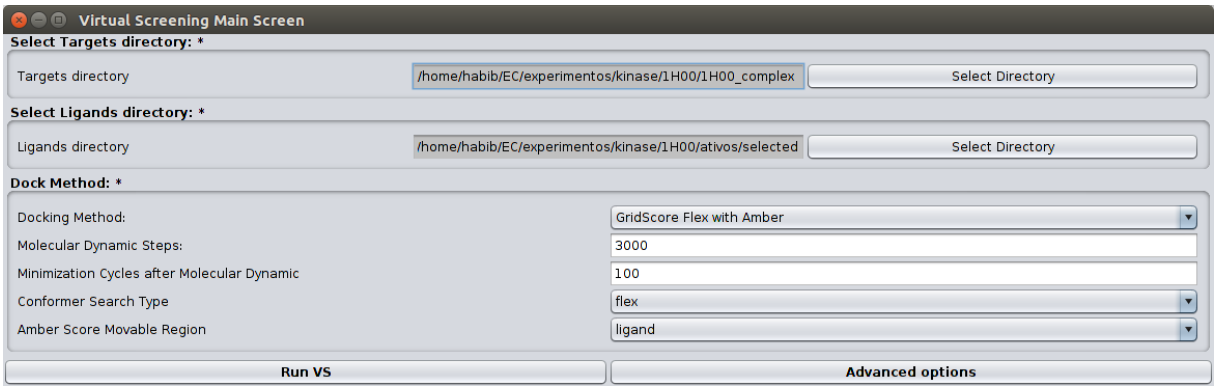



**Figure 47 - VS with DOCK 6 screen with all fields filled**

- Click on the Run VS Button.
- After performing the VS, a result screen is displayed (Figure 48), with the VS result. This screen shows the result of the VS and allows to order the result by PDB code, ligand used or by the Grid Score. The results screen also allows the data to be filtered according to the data of each column (Figure 49).

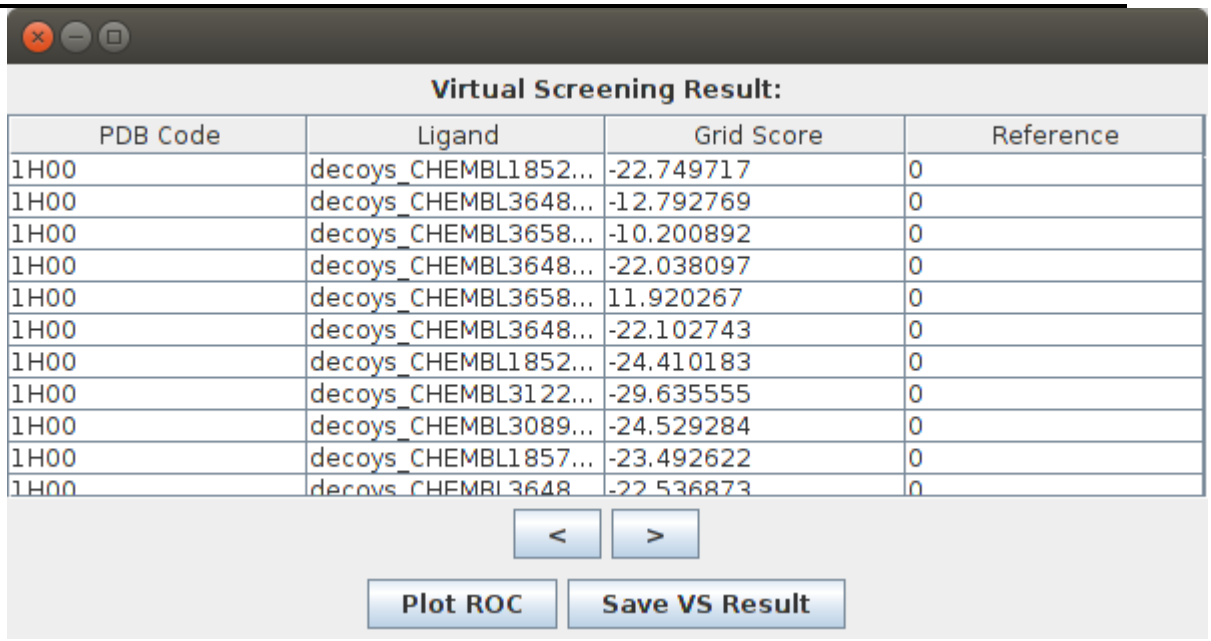



**Figure 48 - DOCK 6 result screen**

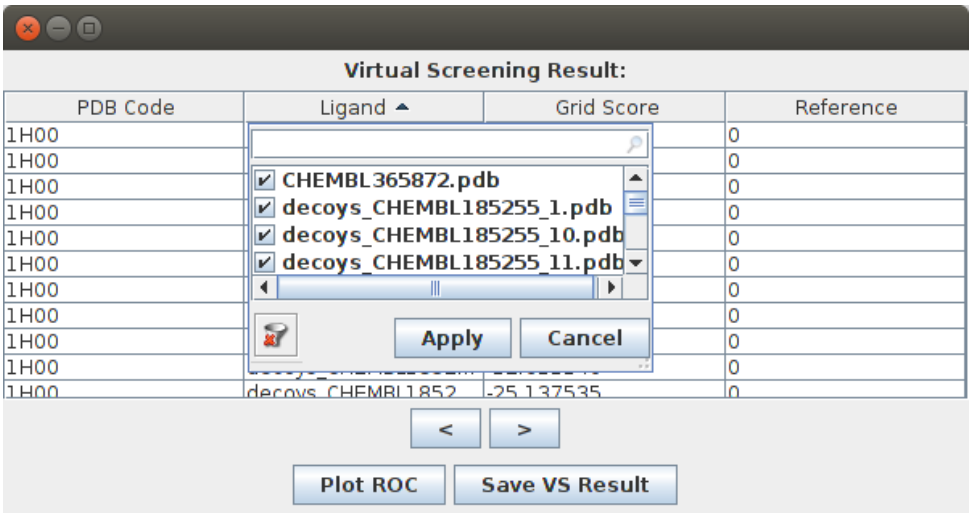



**Figure 49 - Filter being performed by the Ligand Column**

Finally, when filtering the screen by the target PDB code, selecting the active ligands (use Ctrl key to select more than one) and clicking on the Plot ROC command (Figure 50), the ROC curve is generated and AUC is calculated, as shown in Figure 51.

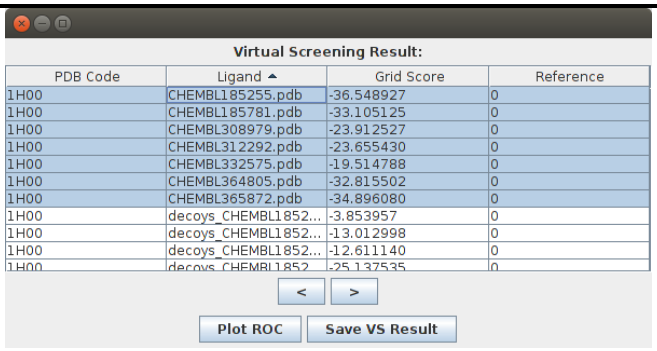



**Figure 50 - Active ligands selected**

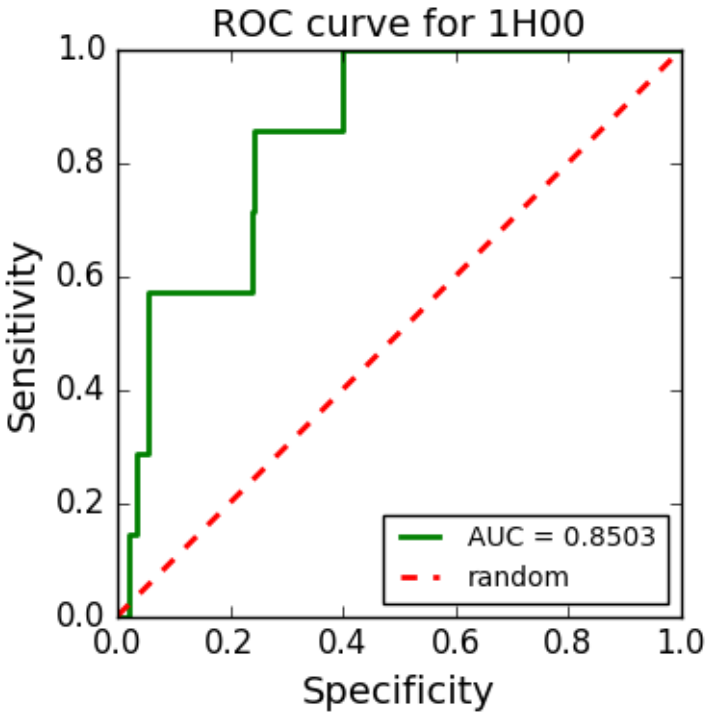



**Figure 51 - ROC Curve for 4O1Z VS**

### 3.1.3 Consensus Docking (CD)

CD assumes that two distinct VS approaches combined yield a better result than a single approach alone. Thus, by combining the results of two distinct approaches, it aims to improve the reliability of VS results [14].

It is a difficult approach to handle, because it involves managing entries in different formats and using different programs. MolAr implements virtual consensus

screening between DOCK 6 and Autodock Vina and it handles all the file conversion needed in this process. In addition, MolAr compares and validates the results. Figure 52 shows a workflow of the Consensus Docking approach implemented by MolAr. After performing the Consensus docking, the result is displayed on the screen (Figure 54).

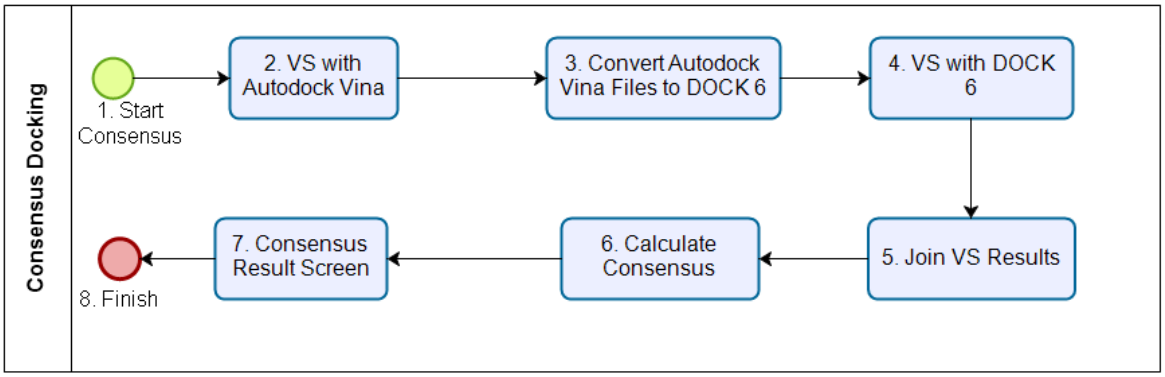



**Figure 52 - Consensus Docking Workflow**

As can be seen in Figure 53, for the realization of the Consensus Docking approach, users only need to inform MolAr the directory of the molecular target database and the ligands directory. The Target data must be in Autodock Vina format (section 3.1.1.2) while the ligands must be in PDB format.

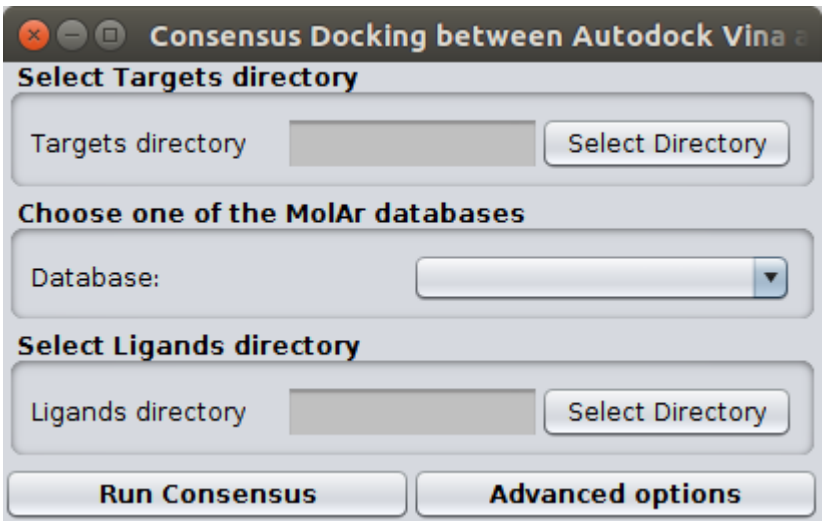



**Figure 53 - Consensus Docking Screen**

The Advanced Options button can be pressed to choose the docking method to be used in DOCK 6 (as well as virtual screening with DOCK 6 - 3.1.2.2)

Once the screen on Figure 53 is filled, user must press the Run consensus button to carry out the VS. MolAr will perform all the necessary conversions and run the VS using both, the Autodock Vina and DOCK 6. At the end of the simulation, MolAr will display DOCK 6 and Autodock Vina binding energy and in a third column, a classification according to the consensus docking which is performed. The result of the scoring function displayed by Autodock Vina and DOCK 6 is normalized to values between 0 and 10. The Consensus result calculated by MolAr corresponds to the average between these 2 values. Thus, the consensus docking scoring function is calculated according to the following equation:

$$consensus = \frac{(normalized\ DOCK6\ result) + (normalized\ autodock\ vina\ result)}{2}$$

Then the list is ordered, and the lowest consensus number is changed to 1 (corresponding to the first position), the second smaller is changed into 2 (second position) and so on. Figure 54 shows an example of a CD result in MolAr.

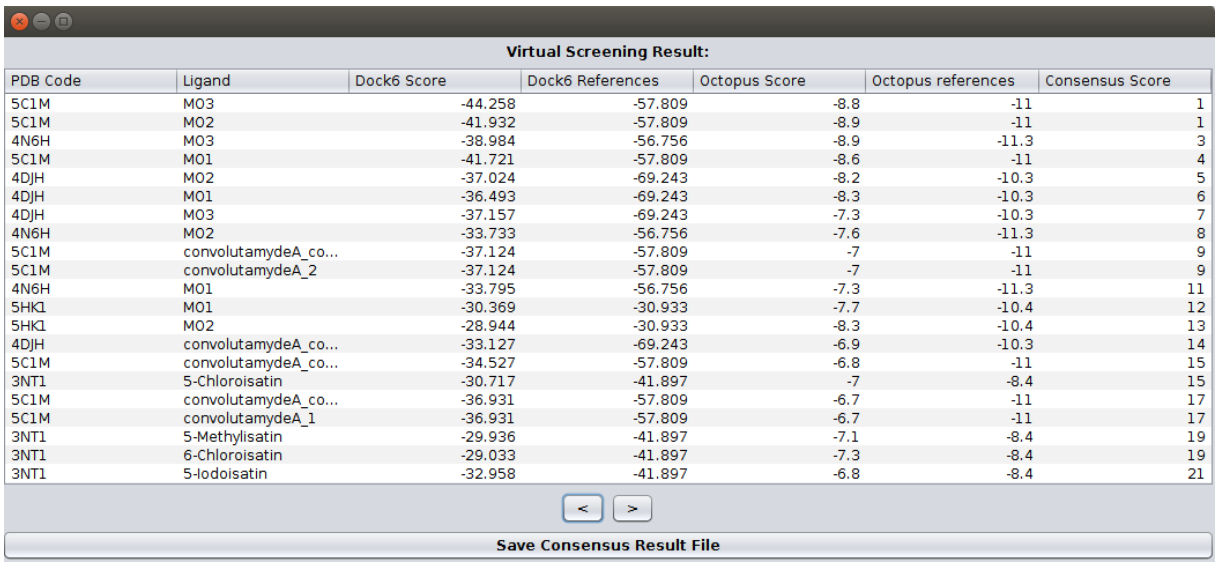



**Figure 54 - Consensus Docking Result Screen**

We will perform a Consensus VS Between Autodock Vina and DOCK 6 using the same input previous used to 1H00 target:

- First, select the target folder used in Octopus (section 3.1.1.2).
- Then, select the ligand folder used in VS with DOCK 6 (section 3.1.2).

- Click on the Run Consensus Button.
- After a processing time, MolAr will show the results.

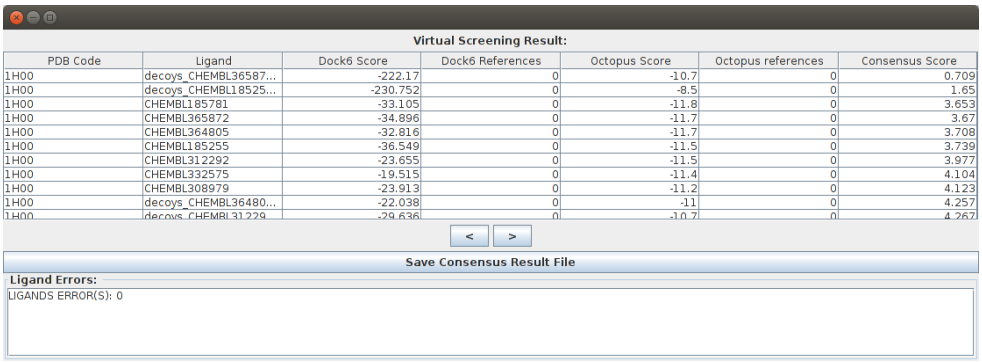


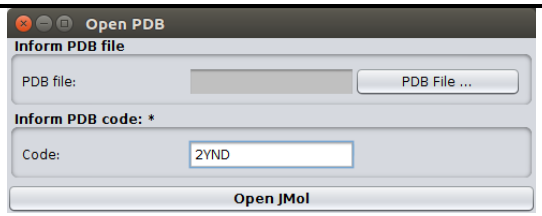

**Figure 55 - Consensus docking result screen**

# 4. Tools Menu

The platform developed has a set of tools to support the realization of the entire VS process. With them, it is possible to visualize the three-dimensional structure of a molecule, to make the quality analysis of a structure, to carry out the adjustment of the protonation state or to generate the ROC curve of a previous VS performed in the platform. These features have been grouped in the Tools menu and in the following subsections, they will be described in detail.

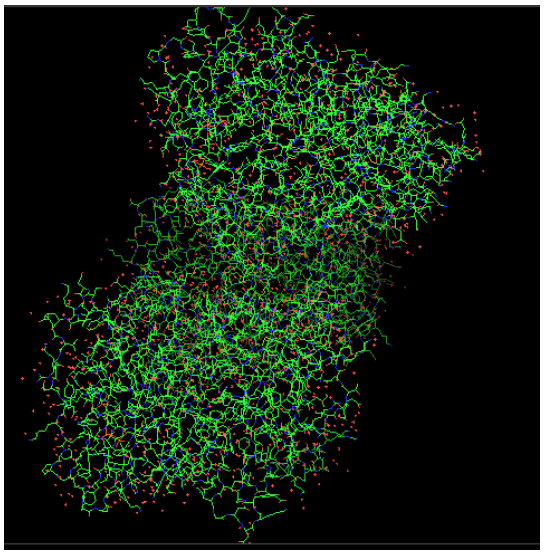
#### 4.1.1.1 View sub menu

It was integrated to the developed platform the possibility of visualizing the 3D structure of a PDB file (Figure 56a). Therefore, Jmol [15] and PyMol [7] tools were integrated into the developed platform. Figure 56a shows the screen that allows opening a protein with JMol or Pymol. It is necessary to inform the PDB file of the protein whose 3D structure is to be visualized or its PDB code. If the PDB file is informed, the protein is displayed. If the PDB code of the protein is reported, the PDB file is downloaded and then displayed in JMol. Figure 56b shows the 3D structure of the 2YND protein using JMol and Figure 56c shows the same protein using Pymol.




(a)

(b)



(c)

**Figure 56 – (a) Open PDB file screen with Jmol; (b) Visualization of the 3D structure of 2YND protein in JMol; (c) Visualization of the 3D structure of 2YND protein in Pymol**

#### 4.1.1.2 Structure analysis submenu

The developed framework allows the analysis of the quality of a structure through the RMSD and the Ramachandran Plot. Figure 57 shows the screen that allows the generation of the Ramachandran Plot. In this screen, it is necessary to inform the PDB file that contains the 3D structure and the resolution of the protein to which the Ramachandran Plot will be generated. Based on this information, MolAr uses Procheck [16] to generate the Ramachandran Plot. Figure 58 shows an example of the Ramachandran Plot generated by the Platform for the protein 1H00.

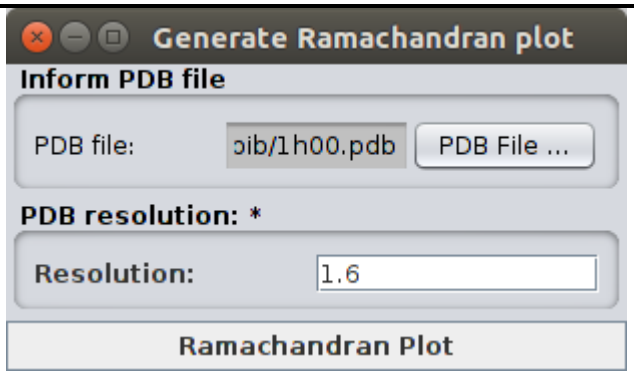
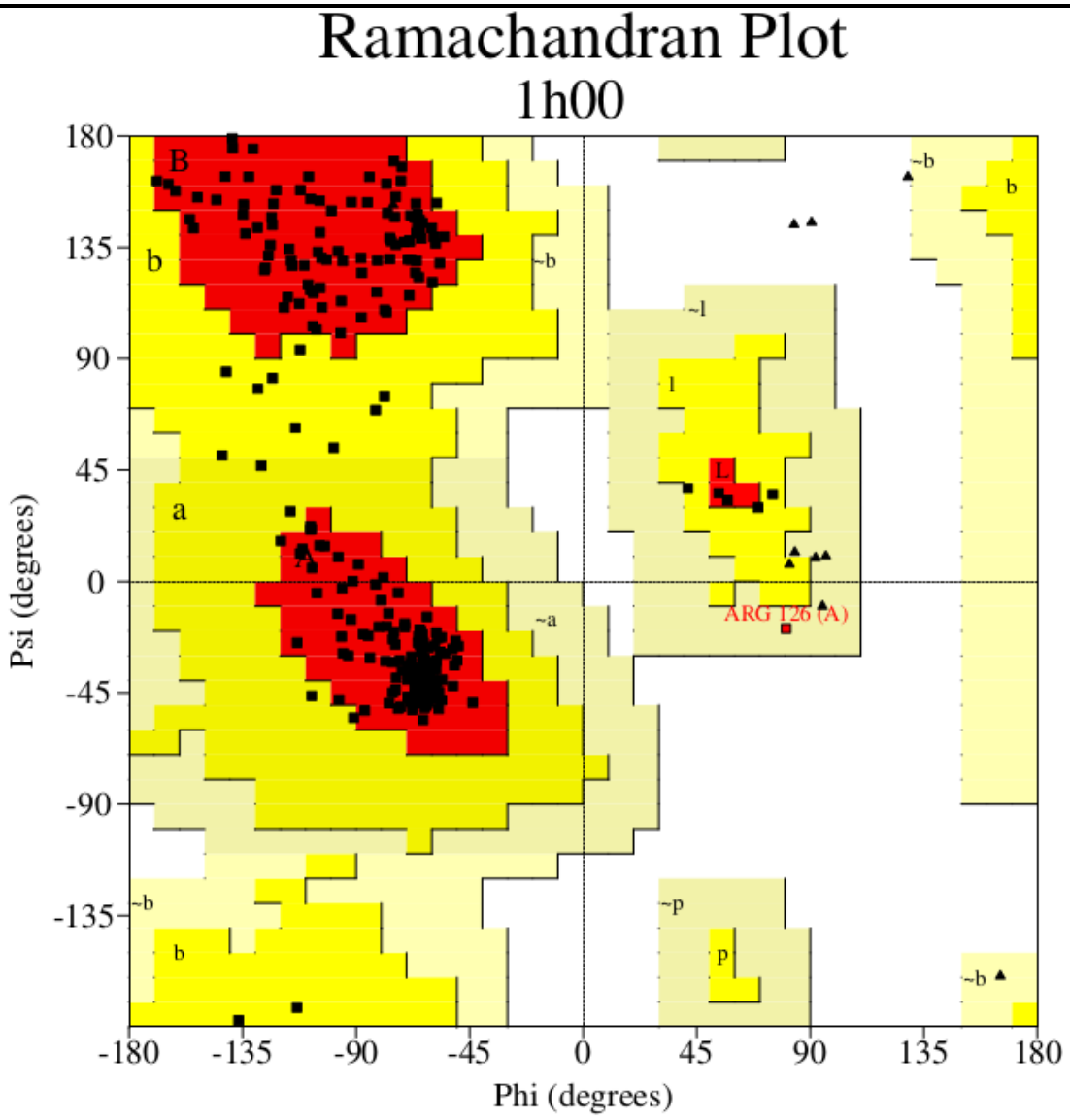



**Figure 57 - Ramachandran plot screen**




Plot statistics

| | | |
|---|---|---|
| Residues in most favoured regions [A,B,L] | 222 | 92.9% |
| Residues in additional allowed regions [a,b,l,p] | 16 | 6.7% |
| Residues in generously allowed regions [~a,~b,~l,~p] | 1 | 0.4% |
| Residues in disallowed regions | 0 | 0.0% |
| Number of non-glycine and non-proline residues | 239 | 100.0% |
| Number of end-residues (excl. Gly and Pro) | 8 | |
| Number of glycine residues (shown as triangles) | 13 | |
| Number of proline residues | 18 | |
| Total number of residues | 278 | |

Based on an analysis of 118 structures of resolution of at least 2.0 Angstroms and R-factor no greater than 20%, a good quality model would be expected to have over 90% in the most favoured regions.

**Figure 58 - Ramachandran plot of the 1H00 protein using MolAr**

The RMSD's calculation is done through a python script that uses Pymol library. To carry out this calculation, just inform the 2 molecules to be compared. Figure 59 shows an example of a RMSD calculation between the 1H00 protein and a model generated in MolAr by homology modeling in section 2.1.

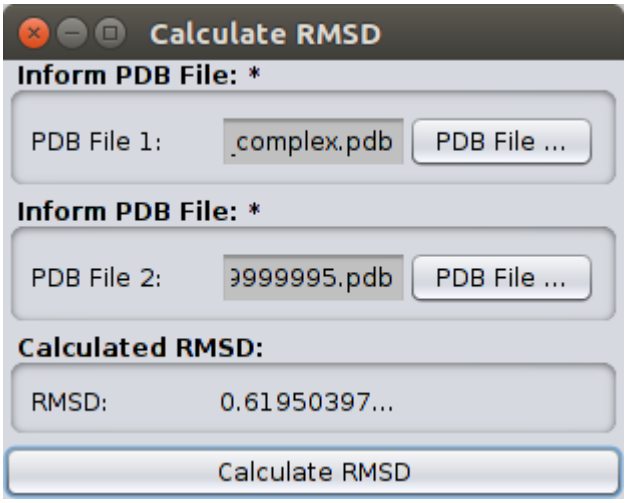



**Figure 59 - RMSD calculation between two proteins after homology modeling**

#### 4.1.1.3 Protonation Submenu

Before performing Docking and Virtual Screening, it is necessary to adjust the protonation state of the 3D structures involved. It is necessary to adjust the protonation state of both the protein and the ligands involved. MolAr uses Propka [17] to adjust the protonation state of proteins and Babel [18] to adjust ligands.

**Protein Protonation**

PROPKA requires the execution of a command line in the Linux prompt. Figure 60 shows a simple interface created in MolAr for the adjustment of the protonation state using PROPKA.

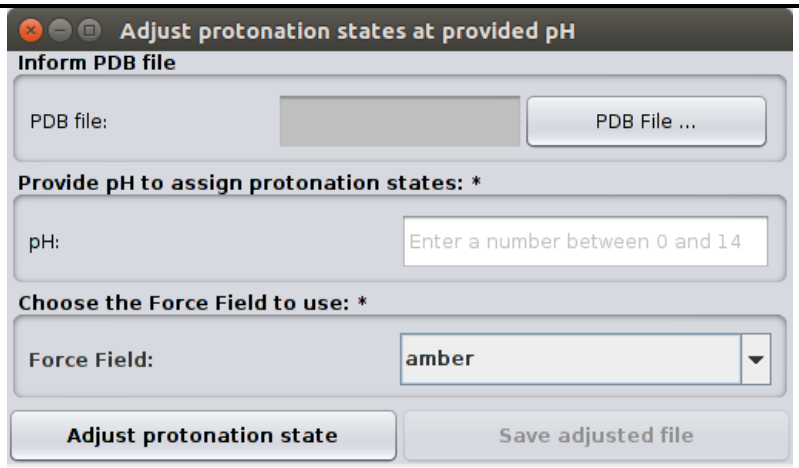



**Figure 60 - Adjust the Protonation State Screen**

In the interface of Figure 60, the PDB file, the pH and the force field to be used in this procedure must be reported for the adjustment of the protonation state. It is important to emphasize that the choice of the force field is very important in this process. This choice defines, among other things, the nomenclature of residues, which is important in the next stages of molecular modeling studies, and the generation of topology files. The interface of the Figure 60, created to automatically adjust the protonation state, is quite intuitive and therefore does not require prior knowledge of how to use the PDB2PQR program. After press the Adjust protonation state button, MolAr will perform the protonation according with pH and Force Field and then the Save adjusted file button will be enabled Figure 61.

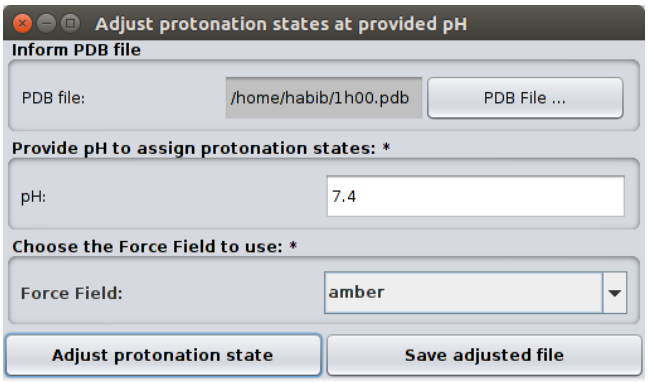



**Figure 61 - Adjust protonation state screen after press the run button**

Then, just click on Save adjusted file button to save the PDB file.

### Ligand Protonation

MolAr allows the protonation state of the ligands to be adjusted using Babel (O'BOYLE *et al.*, 2011). If you want to adjust the protonation state of the ligands using MolAr, access the Tools -> Protonation -> Ligand protonation submenu. MolAr will display Figure 62. Then, after filling in the fields of the scree (folder containing the ligands that will have the protonation state adjusted and the pH field), and pressing the "Adjust protonation state" button, MolAr will adjust the protonation state.

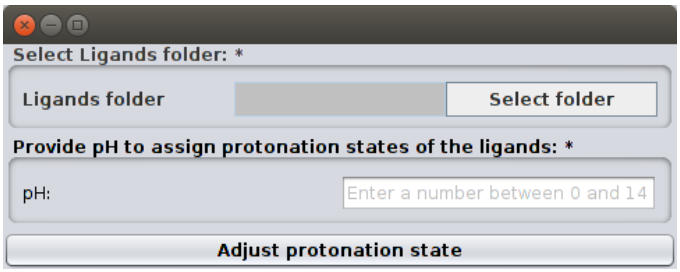



**Figure 62 - Adjust ligand protonation state**

#### 4.1.1.4 Generate Roc Curve submenu

After performing a virtual screening, one of the most important points to be observed is if the program used was able to separate active compounds from inactive ones. The developed program allows the generation of the ROC curve of a given VS, in order to verify if it can separate these compounds, by just pressing the ROC Curve command from the Tools Menu. So, Molar will show the screen of Figure 63. Then, select the file with the result of the Virtual Screening, generated after its execution, and press the button Plot ROC.

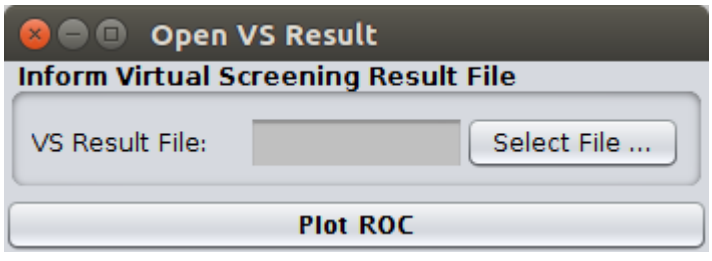



**Figure 63 - ROC Curve screen**

After this button is pressed, the ROC curve is generated. Figure 64 shows an example of a ROC curve. It can be verified that the generated curve also shows the

value of the Area Under the Curve (AUC). If AUC is higher than 0.7, we can verify that the program was able to correctly distinguish between active and inactive compounds [26].

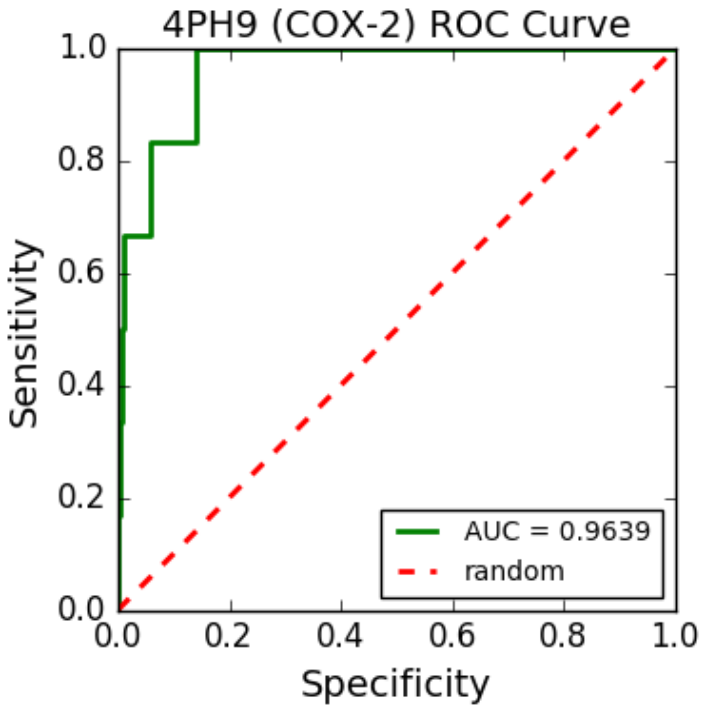



**Figure 64 - ROC Curve generated by the ROC curve screen**

## 5. References


[1] a Šali, “MODELLER: A Program for Protein Structure Modeling Release 9.12, r9480,” *Rockefeller Univ.*, pp. 779–815, 2013.

[2] J. J. P. Stewart, “MOPAC2016™,” 2016. .

[3] D. M. Nikolaev *et al.*, “A Comparative Study of Modern Homology Modeling Algorithms for Rhodopsin Structure Prediction,” *ACS Omega*, vol. 3, no. 7, pp. 7555–7566, Jul. 2018.

[4] O. Trott and A. J. Olson, “AutoDock Vina: Improving the speed and accuracy of docking with a new scoring function, efficient optimization, and multithreading,” *J. Comput. Chem.*, vol. 31, no. 16, p. NA-NA, 2009.

[5] W. J. Allen *et al.*, “DOCK 6: Impact of new features and current docking performance,” *J. Comput. Chem.*, vol. 36, no. 15, pp. 1132–1156, 2015.

[6] J. Gasteiger and M. Marsili, “Iterative partial equalization of orbital electronegativity-a rapid access to atomic charges,” *Tetrahedron*, vol. 36, no. 22, pp. 3219–3228, 1980.

[7] W. L. DeLano, “The PyMOL Molecular Graphics System, Version 1.8,” *Schrödinger LLC*, p. http://www.pymol.org, 2002.

[8] J. J. P. Stewart, “Optimization of parameters for semiempirical methods VI: more modifications to the NDDO approximations and re-optimization of parameters,” *J. Mol. Model*, vol. 19, no. 1, pp. 1–32, 2013.

[9] J. J. P. Stewart, S. C. C. J.J.P. Stewart, J. J. P. Stewart, and S. C. C. J.J.P. Stewart, “MOPAC2012.” Stewart Computational Chemistry, Colorado Springs, CO, USA, CO, USA, 2012.

[10] A. P. Carregal, F. V. Maciel, J. B. Carregal, B. dos Reis Santos, A. M. da Silva, and A. G. Taranto, “Docking-based virtual screening of Brazilian natural compounds using the OOMT as the pharmacological target database,” *J. Mol. Model.*, vol. 23, no. 4, p. 111, Apr. 2017.

[11] I. D. Kuntz, J. M. Blaney, S. J. Oatley, R. Langridge, and T. E. Ferrin, “A geometric approach to macromolecule-ligand interactions,” *J. Mol. Biol.*, vol. 161, no. 2, pp. 269–288, Oct. 1982.

[12] Y. Westermaier, X. Barril, and L. Scapozza, “Virtual screening: An in silico tool for interlacing the chemical universe with the proteome,” *Methods*, vol. 71, pp. 44–57, Jan. 2015.

[13] M. M. Mysinger, M. Carchia, J. J. Irwin, and B. K. Shoichet, “Directory of useful decoys,

enhanced (DUD-E): Better ligands and decoys for better benchmarking," *J. Med. Chem.*, vol. 55, no. 14, pp. 6582–6594, 2012.

[14] D. R. Houston and M. D. Walkinshaw, "Consensus docking: Improving the reliability of docking in a virtual screening context," *J. Chem. Inf. Model.*, vol. 53, no. 2, pp. 384–390, 2013.

[15] A. Herráez, "Biomolecules in the computer: Jmol to the rescue," *Biochem. Mol. Biol. Educ.*, vol. 34, no. 4, pp. 255–261, Jul. 2006.

[16] R. A. Laskowski, M. W. MacArthur, D. S. Moss, and J. M. Thornton, "PROCHECK: a program to check the stereochemical quality of protein structures," *Journal of Applied Crystallography*, vol. 26, no. 2. pp. 283–291, 1993.

[17] T. J. Dolinsky *et al.*, "PDB2PQR: Expanding and upgrading automated preparation of biomolecular structures for molecular simulations," *Nucleic Acids Res.*, vol. 35, no. SUPPL.2, pp. 522–525, 2007.

[18] N. M. O'Boyle, M. Banck, C. A. James, C. Morley, T. Vandermeersch, and G. R. Hutchison, "Open Babel: An Open chemical toolbox," *J. Cheminform.*, vol. 3, no. 10, 2011.

[19] J. Wang, P. Cieplak, and P. A. Kollman, "How well does a restrained electrostatic potential (RESP) model perform in calculating conformational energies of organic and biological molecules?," *J. Comput. Chem.*, vol. 21, no. 12, pp. 1049–1074, Sep. 2000.

[20] A. D. MacKerell *et al.*, "All-Atom Empirical Potential for Molecular Modeling and Dynamics Studies of Proteins †," *J. Phys. Chem. B*, vol. 102, no. 18, pp. 3586–3616, Apr. 1998.

[21] D. Sitkoff, K. A. Sharp, and B. Honig, "Accurate calculation of hydration free energies using macroscopic solvent models," *J. Phys. Chem.*, vol. 98, no. 7, pp. 1978–1988, 1994.

[22] C. L. Tang, E. Alexov, A. M. Pyle, and B. Honig, "Calculation of pKas in RNA: On the Structural Origins and Functional Roles of Protonated Nucleotides," *J. Mol. Biol.*, vol. 366, no. 5, pp. 1475–1496, 2007.

[23] C. Tan, L. Yang, and R. Luo, "How well does Poisson-Boltzmann implicit solvent agree with explicit solvent? A quantitative analysis," *J. Phys. Chem. B*, vol. 110, no. 37, pp. 18680–18687, 2006.

[24] P. Czodrowski, I. Dramburg, C. A. Sotriffer, and G. Klebe, "Development, validation, and application of adapted PEOE charges to estimate pKa values of functional groups in

protein-ligand complexes," *Proteins Struct. Funct. Bioinforma.*, vol. 65, no. 2, pp. 424–437, Aug. 2006.

[25] J. M. J. Swanson, J. A. Wagoner, N. A. Baker, and J. A. McCammon, "optimizing the poisson dielectric boundary with explicit solvent forces and energies: Lessons learned with atom-centered dielectric functions," *J. Chem. Theory Comput.*, vol. 3, no. 1, pp. 170–183, 2007.

[26] A. Hamza, N.-N. Wei, and C.-G. Zhan, "Ligand-Based Virtual Screening Approach Using a New Scoring Function," *J. Chem. Inf. Model.*, vol. 52, no. 4, pp. 963–974, Apr. 2012.