Chapter 20

AutoDock and AutoDockTools for Protein-Ligand Docking: Beta-Site Amyloid Precursor Protein Cleaving Enzyme 1(BACE1) as a Case Study

Nehme El-Hachem, Benjamin Haibe-Kains, Athar Khalil, Firas H. Kobeissy, and Georges Nemer

Abstract

Computational docking and scoring techniques have revolutionized structural bioinformatics by providing unprecedented insights on key aspects of ligand-receptor interaction. Docking is used for optimizing known drugs and for identifying novel binders by predicting their binding mode and affinity. AutoDock and AutoDockTools are free of charge techniques that have been extensively cited in the literature as essential tools in structure-based drug design. Moreover, these methods are fast enough to permit virtual screening of ligand libraries containing tens of thousands of compounds. However using Autodock requires some knowledge in programming which creates a limitation for biologists and makes them prone for commercial applications. Here, we selected a relevant target involved in the progression of Alzheimer disease and provided a fully reproducible docking protocol. This example will show how docking techniques would be an important asset to identify new BACE1 inhibitors. The following friendly user tutorial targets both undergraduate and graduate students, allowing them to understand docking as a computational tool for structure-based drug design.

Key words Docking, AutoDock, Alzheimer disease, BACE1

1 Introduction

The main aim of structure-based pharmaceutical drug design is to find a ligand that will bind to its active site with high affinity and specificity. Computational protein-ligand docking has gained popularity in the drug discovery pipeline, since it is used to predict bound conformations and free energies of binding for small-molecule ligands to macromolecular targets [1]. Docking techniques leveraged a plethora of computational platforms, which improved our understanding of complex interactions between a chemical/drug and its corresponding receptor/s (the "key-lock" concept). Protein-ligand docking is a geometric search problem that will show whether molecules will form a complex (interact/

Firas H. Kobeissy and Stanley M. Stevens, Jr. (eds.), *Neuroproteomics: Methods and Protocols*, Methods in Molecular Biology, vol. 1598, DOI 10.1007/978-1-4939-6952-4_20,

bind), will determine the binding affinity, and will predict the 3D structure of the complex. Because of its cost effectiveness and time-saving properties, protein-ligand docking witnessed many success stories [2, 3].

Multiple methods have been developed to resolve the computational docking problems that range from simple point-matching algorithms to explicit physical simulation methods. AutoDock is a molecular modeling simulation software, effective for protein-ligand docking and is among the most accurate docking tools. Moreover, it is open source software, which makes it publicly available at no charge. On the other hand, AutoDockTools (ADT) is the graphical front-end for setting up and running AutoDock. AutoDockTool combines accuracy in determining the binding pose of a small-sized chemical in a corresponding receptor pocket and a free open source solution available to researchers interested in computational docking.

A typical docking protocol consists of four main steps. First, the preparation of the ligand and the target (sometimes this needs sophisticated steps depending on the nature of the ligand and/or the target). Second, the preparation of docking and scoring parameters (the following files should be created for running Autodock: grid parameter file, map files, docking parameter files). Third, the running of the docking program using a graphical interface or a command-line interface terminal (e.g., AutoDock). Fourth, the analysis and evaluation of the results of docking (comparing docking poses to crystallized ligand). Does the docking correlate with the biology?

In the present chapter, we will provide an easy step-by-step and practical docking protocol using an implementation of AutoDock, and AutodockTools [4]. We highly recommend reading the full and rich documentation provided by the AutoDock team (http://autodock.scripps.edu/faqs-help/manual/autodock-4-2-user-guide/AutoDock4.2.6_UserGuide.pdf). Briefly, the AutoDock engine uses a special format termed PDBQT. This file contains all information about atom types and charges for both ligand and protein. This file is created upon conversion from an original PDB file (http://www.wwpdb.org/documentation/file-format). It can be created using AutoDockTools, the preferred graphical user interface for AutoDock (The generation of the PDBQT file prior to docking will be explained in the methods section).

Energy evaluation of the binding site is achieved using a Grid calculation procedure (AutoGrid). Then, the energetics of the ligand are evaluated against values generated from the interaction terms assigned from the affinity grids calculations.

The AutoDock version 4.2 is used in this chapter since it contains several technical improvements over earlier versions. The new semi-empirical force field used by AutoDock 4.2 is enriched with additional terms including updated de-solvation term and an improved unbound state model to estimate the free energy of

ligand-receptor interaction. The final step consists of docking the ligand using several search algorithms. One of the standards and efficient methods to look for optimal ligand binding conformation is the Lamarckian genetic algorithm (LGA) implemented in AutoDock.

To make use of this technique in a clinically relevant case study, we chose Alzheimer disease as an example. It is a chronic and progressive neurodegenerative disorder, which is the most common cause of dementia in the elderly. This disease progresses to death if not treated, though till now there is no efficient treatment for it [5]. A proposed mechanism leading to this pathology is the aggregation of amyloid-β peptides (Aβ) of 36–43 amino acids resulting in neuronal death.

A landmark discovery in Alzheimer disease was the identification of BACE1 as a novel class of type I transmembrane aspartic protease. This protease is comprised of 501 amino acids that trigger the production of the toxic Aβ peptides. BACE1 will be thus used as a model for this practical docking tutorial since it is a promising target for the treatment of Alzheimer's disease [6].

Interestingly, a team at AMGEN Inc. have recently submitted a crystal structure of BACE1 in complex with 2-aminooxazoline 3-azaxanthene inhibitor 28 [7]. http://www.rcsb.org/pdb/explore/explore.do?structureId=4XKX. We will perform a docking simulation using the PDB entry 4XKX in order to evaluate the accuracy of AutoDock in placing the inhibitor of BACE1 in the binding site compared to its experimental coordinates.

2 Materials

2.1 Starting with a Set of Preliminary and Important Requirements

This would require friendly user tools to manipulate and visualize protein-ligand complexes such as Discovery Studio http://accelrys.com/products/collaborative-science/biovia-discovery-studio/visualization.html (Windows OS only and need registration) or the Swiss-PdbViewer http://spdbv.vital-it.ch/ [Windows and UNIX systems (Mac OS X)]. These tools have an excellent help section and explain in detail how to manipulate protein-ligand complexes, remove solvent, fix structure, etc. All users are encouraged to be familiar with these tools (extracting raw coordinates data and basic manipulation are out of the scope of this chapter).

2.2 Download the pdb (Protein Database) File

Crystal structure of BACE1 in complex with 2-aminooxazoline 3-azaxanthene inhibitor is downloaded from: http://www.rcsb.org/pdb/download/downloadFile.do?fileFormat=pdb&compression=NO&structureId=4XKX. The current crystal structure has 411 amino acid residues and the monomer has a resolution of 1.8 Angstroms. For convenience, structures of both—target protein and corresponding ligand—were extracted separately from the

original pdb files. Users are highly encouraged to regenerate the target.pdb and ligand.pdb files using the Swiss-PdbViewer http://spdbv.vital-it.ch/. For this exercise, both files were already provided (*see* the supplemental material section) to reproduce the computational experiment.

2.3 Autodock 4.2.6

Download the latest version of Autodock from http://autodock.scripps.edu/downloads/autodock-registration/autodock-4-2-download-page/. The newest AutoDock version runs natively under Windows, see instructions for installation (http://autodock.scripps.edu/downloads/autodock-4-2-x-installation-on-windows). Autodock is also available for Mac OS X and Linux platforms. The main files are Autodock and AutoGrid necessary to run the pre-docking (energy maps), docking, and scoring calculation (*see* **Note 1**).

2.4 MGL/AutoDockTools

AutodockTools can be downloaded from http://mgltools.scripps.edu/downloads. It is a Graphical User Interface used to prepare input, run and analyze dockings generated from Autodock (adding atomic charges, fixing bonds, adding hydrogens, preparing the ligand and the target in convenient PDBQT format compatible with Autodock, creating grids and docking parameter file and visualizing interactively docking results). Please refer to **Note 2**, if running Mac OS X (*see* **Note 2**).

3 Method/Docking Approach

The following steps are critical because they dictate the procedure for running AutoGrid and AutoDock and provide precise docking parameters. The coordinated files and corresponding information should be created in a specific format termed PDBQT, which contains atom/bond types, partial atomic charges, etc. These data types are prepared (typically) using AutoDockTools. In this chapter, we will limit our docking experiment to default settings.

3.1 Converting PDB Files to PDBQT Format

3.1.1 Target Preparation

The process of creating a target PDBQT from a PDB file (from crystal structure) consists of the following steps (*see* **Notes 3** and **4**):

1. Open MGLTools and Autodock tools from your desktop or program files.
2. A Figure of the basic interface is provided in Fig. 1a.
3. Make sure that target.pdb and ligand.pdb files are in the same folder, e.g., ~Desktop\autodock (for Windows users).
4. From the File menu > open > Read Molecule (*see* Fig. 1b).
5. Select target.pdb from folder: ~Desktop\autodock.

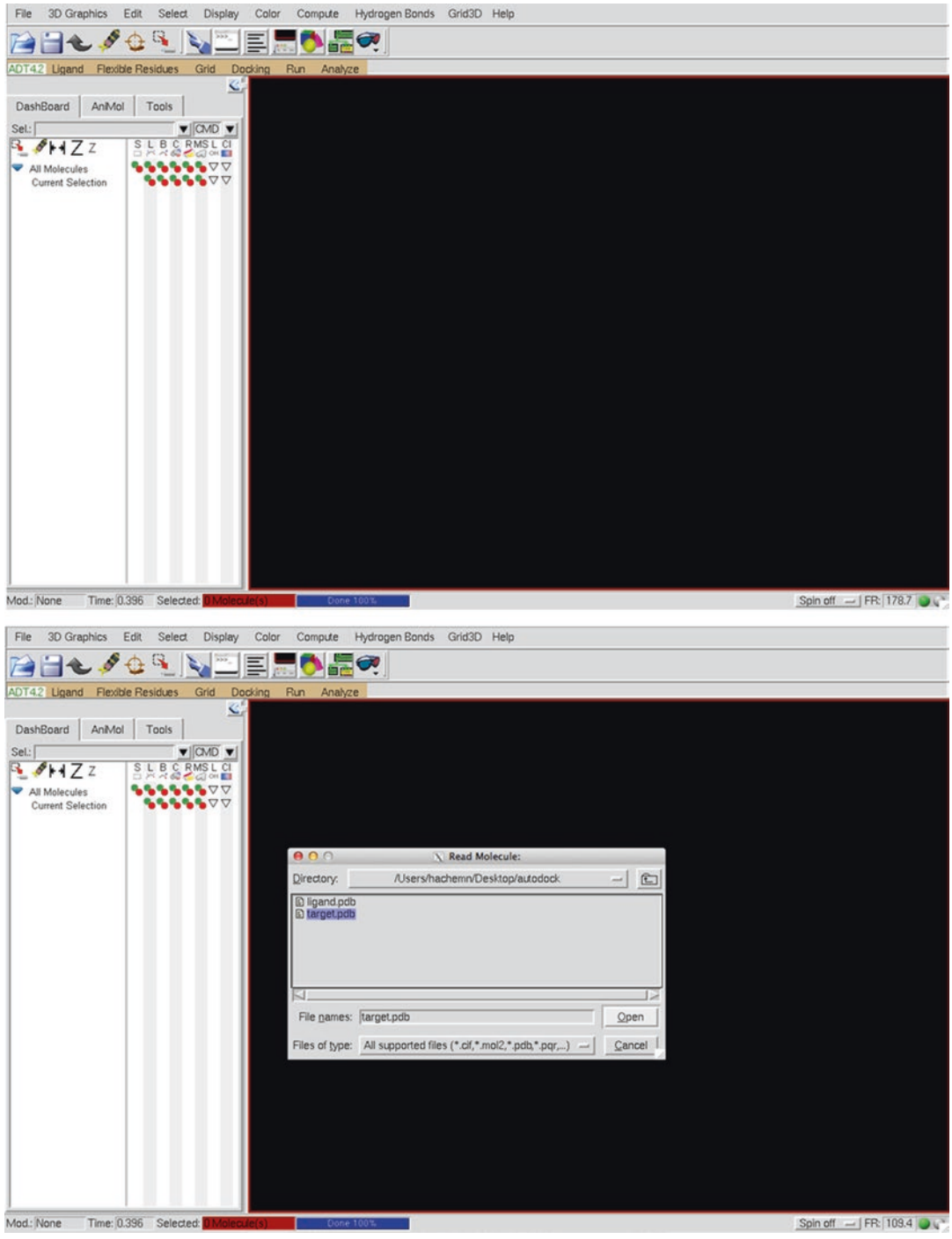

Fig. 1 The AutoDock Tools main window and the read molecule widget

6. The crystal 3D structure of BACE1 will appear on screen (3D viewer, *see* Fig. 2a).
7. To visualize, select, or color the protein, make use of the dashboard located in the left side of the 3D viewer.
8. Click on Edit menu > add hydrogens: add polar hydrogens, fix bond order, and renumber residues including the newly added hydrogen atoms (*see* Fig. 2b).

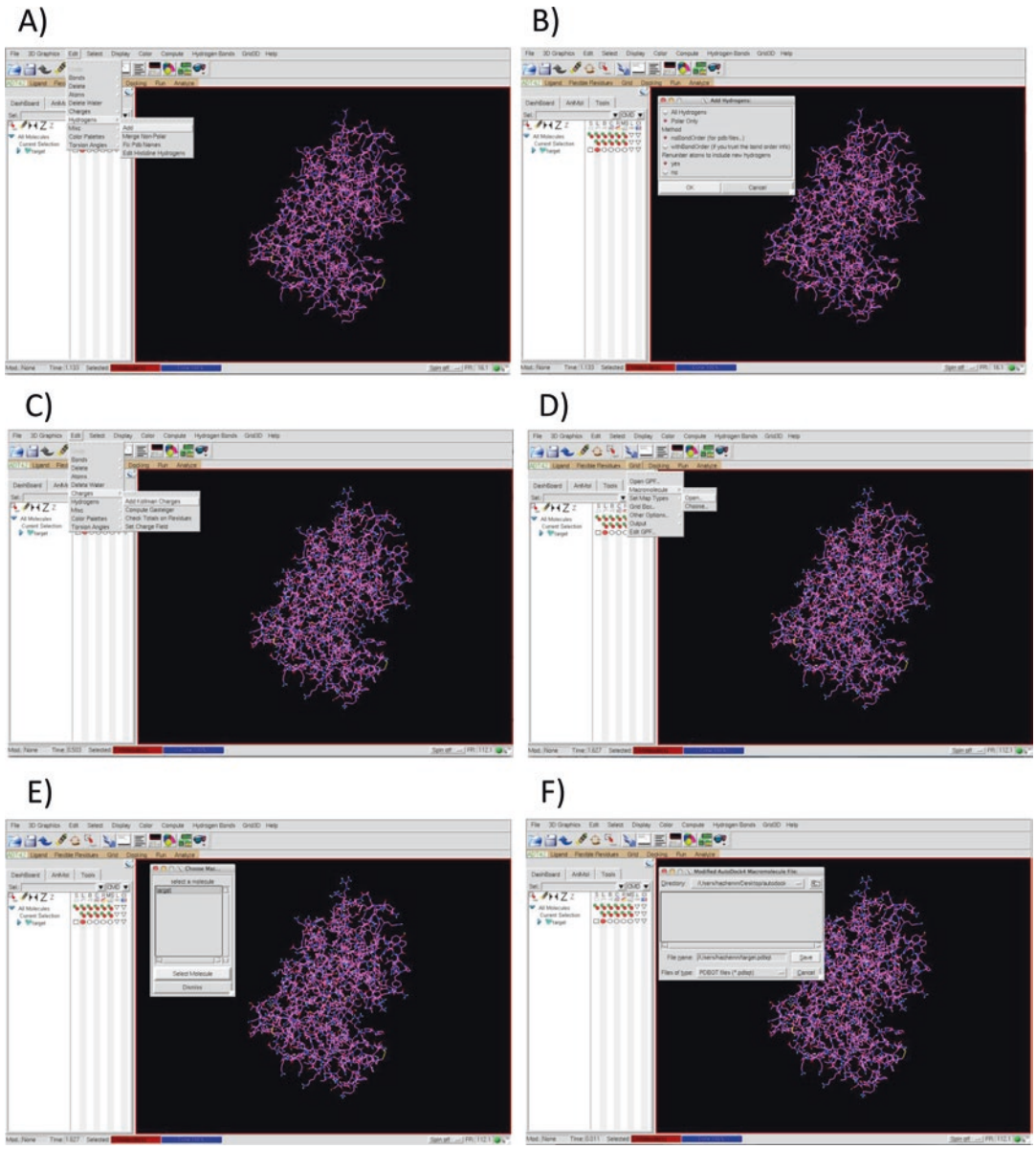

Fig. 2 Panels (**a**–**f**) represent the needed steps to prepare the protein before grid computing. The major steps involve adding hydrogen and appropriate charges to the protein and finally saving the file in ".pdbqt" format, which is compatible with autodock

9. Click on Edit menu > add charges: add Kollman charges. These are derived from quantum mechanics (*see* Fig. 2c). This adds convenient partial charges to the protein.
10. Click on Grid menu > Macromolecules > open and select target molecule and save it under target.pdbqt in the same folder where you created target.pdb (*see* Fig. 2d–f). This stores the partial charges and Autodock atom types that are compatible with Autodock grid computing.

3.1.2 Ligand Preparation

The process of creating a ligand.pdbqt from a ligand.pdb file (from crystal structure) consists of the following steps (*see* **Notes 3** and **4**).

From the File menu > open > Read Molecule (*see* Fig. 1b)

1. Select ligand.pdb from folder: ~Desktop\autodock.
2. The crystal 3D structure of the ligand will appear on screen (same as for target preparation section).
3. Click on Edit menu > atoms > assign AD4 types (to get atom types for Autodock 4).
4. Click on Edit menu > add hydrogens: add polar hydrogens, fix bond order, and renumber.
5. Click on Edit menu > add charges: select Gasteiger charges.
6. Click on Ligand menu > input > choose > select molecule for Autodock 4 (select the ligand in the box, *see* Fig. 3a).
7. Click on Ligand menu > output > save as PDBQT: write ligand.pdbqt in the same folder along with target.pdb (make sure to copy them if not found in the same folder).

3.2 Docking and Scoring Approach

3.2.1 Preparation of the Grid Parameter File (.GPF) to Run with Autogrid4.exe

1. Open the Grid menu in AutoDockTools to prepare the parameters for Autogrid calculations (*see* Figs. 3b, c, 4a, b, and 5a, b).
2. Click Set Map Types > Open Ligand: select and open the ligand.pdbqt saved previously (AutoGrid calculates grid maps of interaction energies for various atom types. This is important to calibrate docking procedure).
3. Click again on Grid menu > Grid Box (This will pick the coordinates of the binding site for the search engine. In our case, it should be centered on the ligand because the site is known).
4. Center the X, Y, Z atomic coordinates on the ligand (In our case study, the ligand binding site is known, however in other cases the grid box can be approximated).
5. In the grid options box > File > close saving current option. This will save the center grid box.
6. Click again on the Grid menu > Output > save gpf. Save as **dock.gpf**. This creates one grid map for each atom type in the

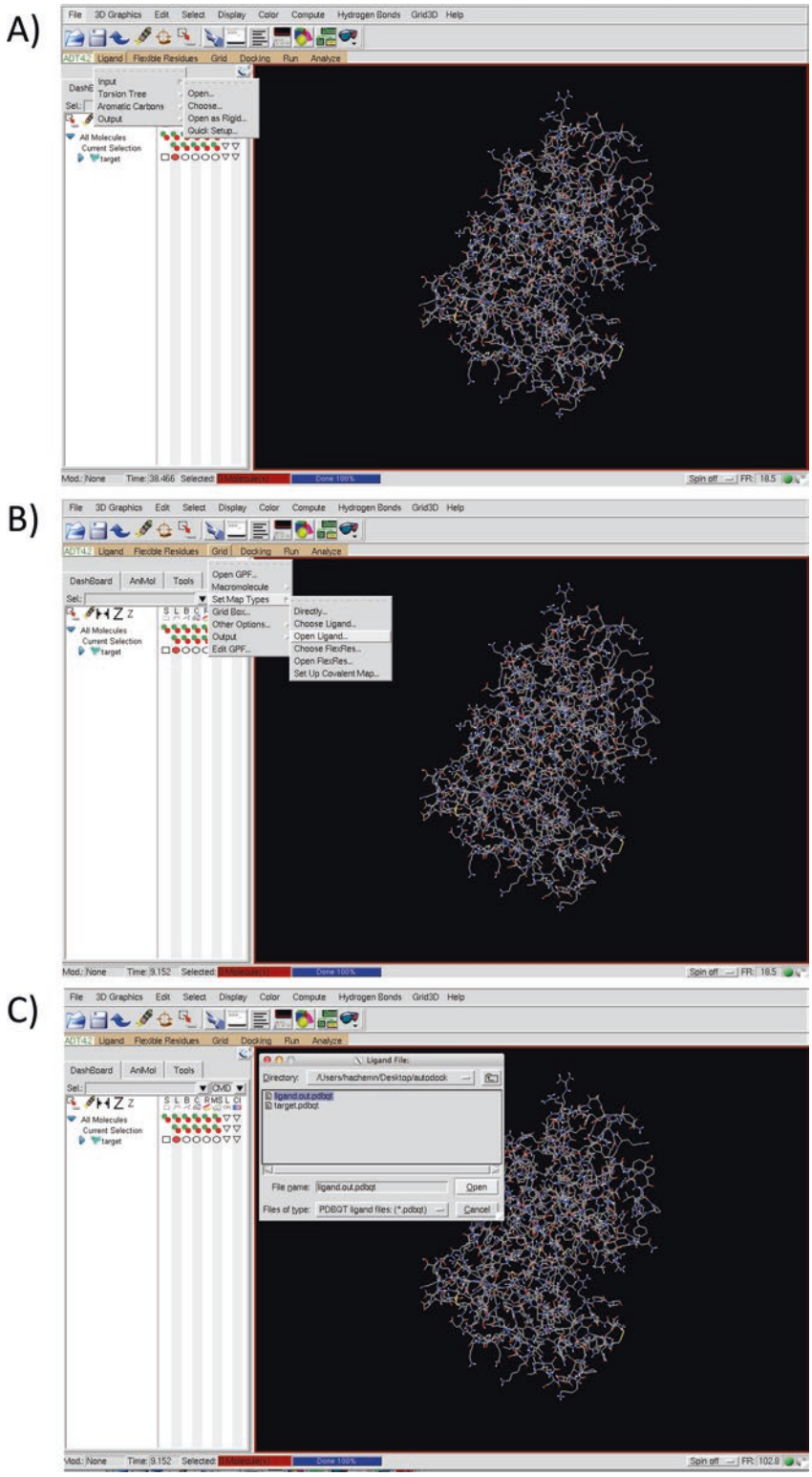

Fig. 3 Panels (**a**–**c**): the ligand is prepared with the quick setup option from AutoDockTools. Then AutoDockTools generates the energy maps for the ligand/receptor from the ligand.pdbqt file

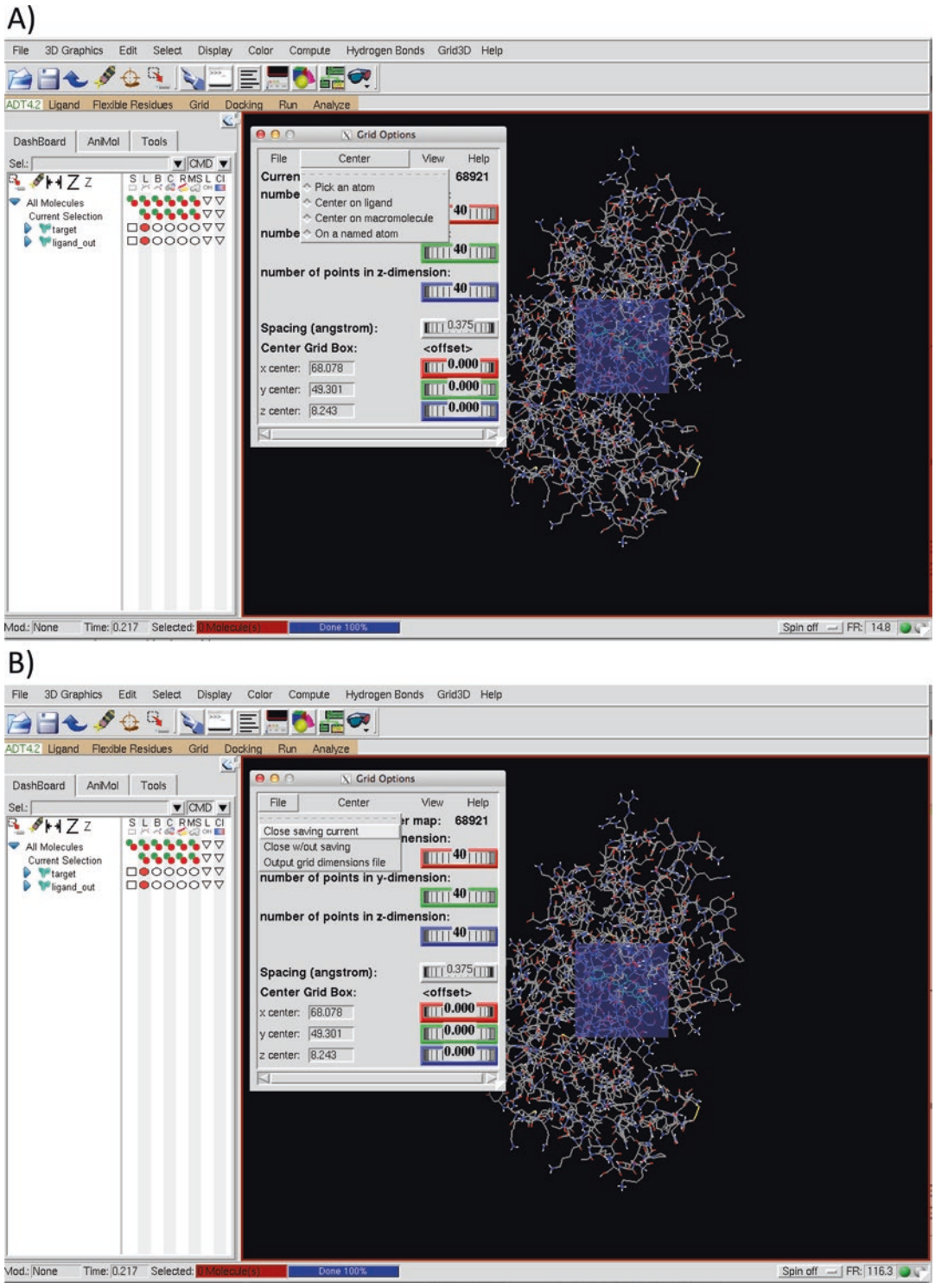

Fig. 4 Panels (**a**–**b**): The grid box and the grid options widget. The grid is centered on the ligand in the Beta-site amyloid precursor protein cleaving enzyme 1(BACE1) case study

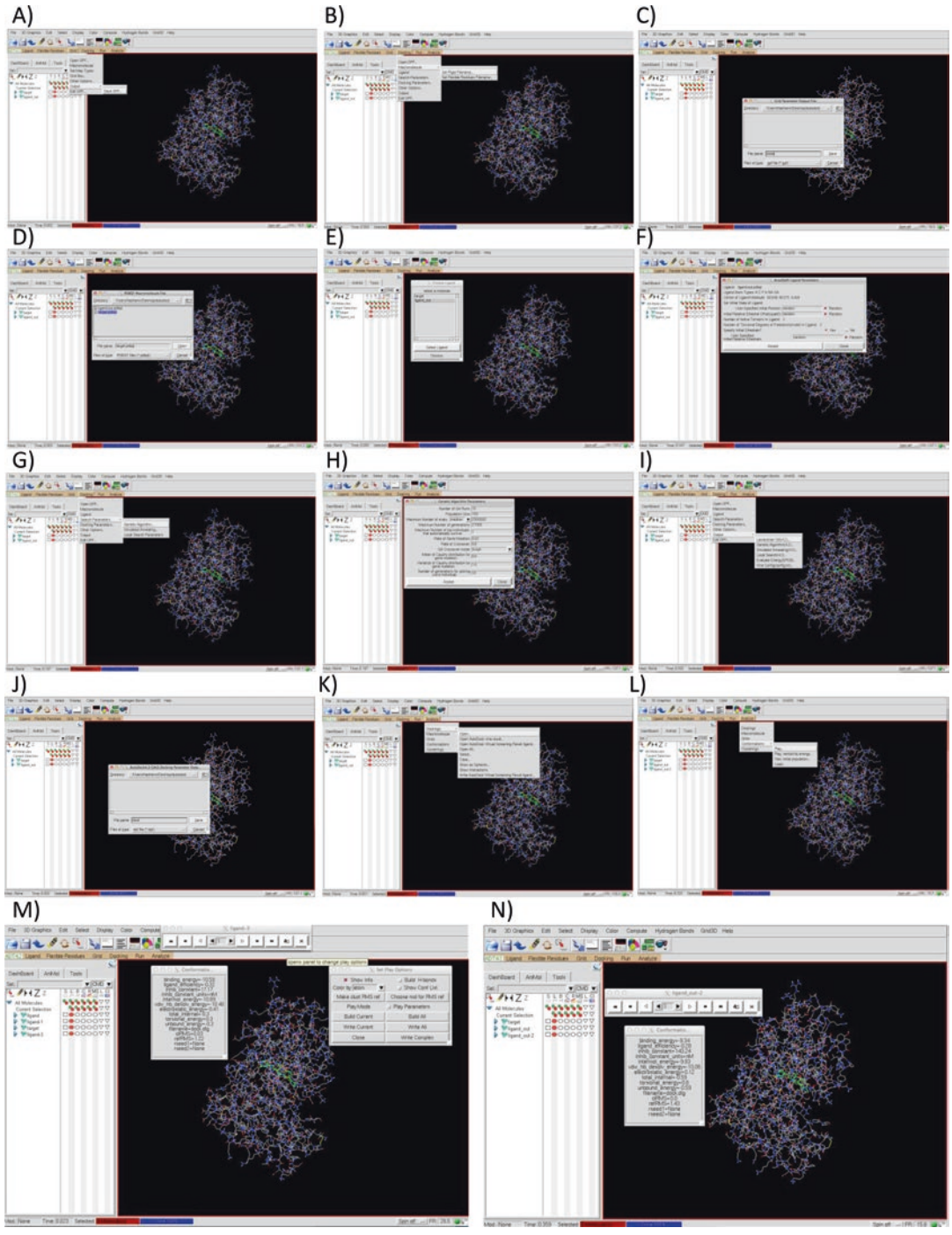

Fig. 5 Panels (**a**–**n**): The run autogrid and autodock widgets, the grid and docking files, the docking parameters used (genetic algorithm), the results of the docking (estimated binding energy of the ligand and generated conformations)

ligand plus an electrostatics and a desolvation map (Save in the same directory as the other pdbqt files).

3.2.2 Preparation of the Docking Parameter File (.DPF) to Run with Autodock4. exe

Open the Docking menu in AutoDockTools to prepare the parameters for Autodock (*see* Fig. 5c–j).

1. Open Docking menu > Macromolecule > Set rigid Filename > select Target.pdbqt.
2. Click on Docking > Choose > ligand (if the ligand in pdbqt format is open in the viewer).
3. If not in the viewer, click on Docking > open: choose the location of the ligand.pdbqt file.
4. In the Autodpf4 parameters box, click Accept Ligand Parameters.
5. Click again on Docking > Search Parameters > Genetic Algorithm > Accept (We are keeping the default parameters but advanced users can play with the settings).
6. Click Output > LamarkianGA algorithm > Save the file as **dock.dpf** in the same directory (~Desktop\autodock).

Now you have all the required files for docking (target.pdbqt, ligand.pdbqt, dock.gpf, dock.dpf).

3.3 Running Autogrid4 and Autodock4

After downloading and installing AutoDock 4.2.6 from http://autodock.scripps.edu/downloads/autodock-registration/autodock-4-2-download-page/.

For Windows users, Start > Run and type "cmd.exe" then type the command: C:\Program Files\The Scripps Research Institute\Autodock\autodock4.exe"

For windows, the user should see this message:

```
C:\Users\mgl>"C:\Program Files\The Scripps
Research Institute\Autodock"\autodock4.exe
usage: AutoDock -p parameter_filename
-l log_filename
-k (Keep original residue numbers)
-i (Ignore header-checking)
-t (Parse the PDBQT file to check torsions,
then stop.)
-d (Increment debug level)
-C (Print copyright notice)
--version (Print autodock version)
--help (Display this message)
C:\Users\mgl>
```

For Unix-like operating environment, users should copy the executable to usr/local/bin folder.

3.3.1 Running Autogrid4

Start > Run and type "cmd.exe", change your working directory to ~Desktop\autodock (using the cd command).

Type in the console: autogrid4.exe -p dock.gpf -l dock.glg &.

3.3.2 Running Autodock4

Type in the console: autodock4.exe -p dock.dpf -l dock.dlg &.

This will take some time depending on your CPU and memory capacity.

The dlg file contains all information about the docking runs, the estimated binding energy in Kcal/mol, and other information such as the RMSD vs. crystal binding pose.

3.4 Analyzing Docking Results

To analyze docking results, open the Analyze menu (*see* Fig. 5k–n).

1. Docking results are found in the .dlg log file.
2. Open the Analyze menu > Docking > Open > dock.dlg.
3. Open Analyze menu > Conformations > Play.

This shows the conformation from 1 to 10 of the ligand bound to BACE1.

The best conformation has a binding energy (G) of −10.59 kcal/mol and inhibition constant (Ki) of 17.17 nM (nanomolar) and a RMSD (root-mean-square deviation of atomic positions) from reference structure of 1.22 Å. This shows that the results from Autodock are reliable and accurate (in the nanomolar range for a known inhibitor). Docking and virtual screening would be an important asset to identify new BACE1 inhibitors.

4 Notes

1. We recommend checking the manual of Autodock for technical details.
2. You should install XQuartz http://www.xquartz.org/ on a Mac OS X platforms to run AutoDockTools.
3. Be aware that Autodock takes a molecule at a time, so to run virtual screening experiments using a library of small molecules (provided in 3D format), you should use autodock vina instead http://vina.scripps.edu/.
4. Be critical since coordinate preparation, protein and ligand check is an important step to ensure good docking results. You should be aware that the Babel tool is the default engine to add charges and hydrogens. However, some more refinement such as energy minimization, protonation states should be taken into consideration for some special types of targets and ligands (metalloproteins, etc.). In such case, one should consider more sophisticated methods to prepare ligand and target protein. If you have a limited expertise in command-line tools you can always check commercial tools http://accelrys.com/products/collaborative-science/biovia-discovery-studio/structure-based-design.html or https://www.biosolveit.de/.

5 Conclusion

The aim of this step-by-step docking protocol is to bridge the gap between molecular biologists and bioinformaticians. We provided a simple and user-friendly exercise by docking an inhibitor against a therapeutic target in Alzheimer"s disease.

Furthermore, we showed that docking results from Autodock are reliable and accurate (in the nanomolar range). Docking and virtual screening will be helpful in identifying the potential leads to design novel BACE1 inhibitors for AD therapy.

References

1. Forli S, Huey R, Pique ME, Sanner MF, Goodsell DS, Olson AJ (2016) Computational protein-ligand docking and virtual drug screening with the AutoDock suite. Nat Protoc 11:905–919
2. Siddiquee K, Zhang S, Guida WC, Blaskovich MA, Greedy B, Lawrence HR, Yip ML, Jove R, McLaughlin MM, Lawrence NJ, Sebti SM, Turkson J (2007) Selective chemical probe inhibitor of Stat3, identified through structure-based virtual screening, induces antitumor activity. Proc Natl Acad Sci U S A 104:7391–7396
3. Li C, Xu L, Wolan DW, Wilson IA, Olson AJ (2004) Virtual screening of human 5-aminoimidazole-4-carboxamide ribonucleotide transformylase against the NCI diversity set by use of AutoDock to identify novel nonfolate inhibitors. J Med Chem 47:6681–6690
4. Morris GM, Huey R, Lindstrom W, Sanner MF, Belew RK, Goodsell DS, Olson AJ (2009) AutoDock4 and AutoDockTools4: Automated docking with selective receptor flexibility. J Comput Chem 30:2785–2791
5. Javaid FZ, Brenton J, Guo L, Cordeiro MF (2016) Visual and ocular manifestations of Alzheimer's disease and their use as biomarkers for diagnosis and progression. Front Neurol 7:55
6. Yan R, Vassar R (2014) Targeting the beta secretase BACE1 for Alzheimer's disease therapy. Lancet Neurol 13:319–329
7. Chen JJ, Liu Q, Yuan C, Gore V, Lopez P, Ma V, Amegadzie A, Qian W, Judd TC, Minatti AE, Brown J, Cheng Y, Xue M, Zhong W, Dineen TA, Epstein O, Human J, Kreiman C, Marx I, Weiss MM, Hitchcock SA, Powers TS, Chen K, Wen PH, Whittington DA, Cheng AC, Bartberger MD, Hickman D, Werner JA, Vargas HM, Everds NE, Vonderfecht SL, Dunn RT 2nd, Wood S, Fremeau RT Jr, White RD, Patel VF (2015) Development of 2-aminooxazoline 3-azaxanthenes as orally efficacious beta-secretase inhibitors for the potential treatment of Alzheimer's disease. Bioorg Med Chem Lett 25:767–774