Exercise: Docking of Donepezil in Acetylcholinesterase

Aim

Practice on the use of AutoDock to explore the potential binding mode of a ligand to the biological target and obtain an (approximate) estimate of the best pose according to the score function.

Context

Docking is one of the most important tools used in both academia and pharmaceutical companies to explore the binding mode of small molecules to biological receptors. However, it suffers from many limitations that after several decades remain unsolved, such as the accuracy of the scoring function, the treatment of conformational flexibility, or the description of hydration effects. Thus, a careful inspection of the results of a docking calculation is necessary.

The presence of these limitations justifies the existence of a large variety of docking programs. This is indicative that a perfect solution to the 'docking problem' is still missing.

In spite of these limitations, it can be valuable to formulate hypothesis about the the binding mode of novel chemical scaffolds, especially when some pharmacophore constraints can be imposed to guide the conformational search, or to define a first step to filter out the exploration of large chemical libraries.

1. Starting with Autodock program

1.1. What is AutoDock?

AutoDock is a suite of automated docking tools. It is designed to predict how small molecules, such as substrates or drug candidates, bind to a receptor of known 3D structure.
AutoDock 4 actually consists of two main programs:

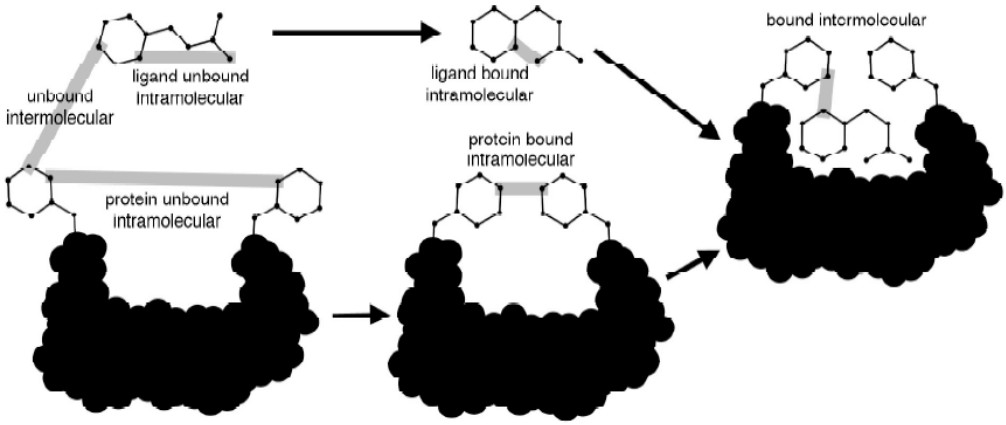
- *autodock* performs the docking of the ligand to a set of grids describing the target protein;
- *autogrid* pre-calculates these grids.

They have also developed a graphical user interface called **AutoDockTools** (ADT; see Annex I) to assist the preparation of the docking calculation.

1.2. The scoring function

AutoDock 4.2 uses a semi-empirical scoring function to evaluate conformations during docking simulations. The function was parameterized using a large number of protein-inhibitor complexes considering both structure and inhibition constants.

The scoring function evaluates the binding affinity in two steps (**Scheme 1**):
i) Starting from unbound states of both ligand and protein, the intramolecular energetics are estimated for the transition from these unbound states to the bound conformation of the ligand-protein complex.
ii) Evaluation of the intermolecular energetics between ligand and protein.



Scheme 1. Evaluation of the scoring function in AutoDock.

This is accomplished through 6 pair-wise evaluations (V) and an estimate of the conformational entropy lost upon binding (ΔS_{conf}) of ligand (L) and protein (P).

$$\Delta G = (V_{bound}^{L-L} - V_{unbound}^{L-L}) + (V_{bound}^{P-P} - V_{unbound}^{P-P}) + (V_{bound}^{P-L} - V_{unbound}^{P-L} + \Delta S_{conf})$$

Each of the pair-wise energetic terms includes evaluations for dispersion/repulsion, hydrogen bonding, electrostatics, and desolvation.

$$V = W_{vdw}\sum_{i,j}\left(\frac{A_{ij}}{r_{ij}^{12}} - \frac{B_{ij}}{r_{ij}^{6}}\right) + W_{hbond}\sum_{i,j}E(t)\left(\frac{C_{ij}}{r_{ij}^{12}} - \frac{D_{ij}}{r_{ij}^{10}}\right) + W_{elec}\sum_{i,j}\frac{q_i q_j}{e(r_{ij})r_{ij}} + W_{sol}\sum_{i,j}\left(S_i V_j + S_j V_i\right)e^{(-r_{ij}^2/2\sigma^2)}$$

where the weighting constants W have been optimized to calibrate the empirical free energy based on a set of experimentally determined binding constants.

1.3. What do we need to run AutoDock?

To run a docking with AutoDock, we need 4 different files:

i) one file for our target, which obviously does not have to contain any ligand or wàter molecules and has to contain (polar) hydrogen atoms.
ii) one file for the ligand.
iii) one file with the grid maps
iv) one file with the parameters chosen for the docking computation.

The docking calculation implies 4 steps:

i) preparation of coordinate files
ii) AutoGrid calculation
iii) docking with AutoDock
iv) Analysis of results

We can prepare all these files through ADT.

2. Computational protocol in AutoDock

2.1. Preparing coordinates

We need **PDBQT** files for both ligand and protein. PDBQT files contain the coordinates and information about the partial charges, atom types, polar hydrogen atoms, and flexibility (in cases where flexibility of ligand and/or side chains in the protein are explored).

Pre-processing the protein (PDB file).

Here we remove the crystallographic water molecules (co-solutes, ions) and add all hydrogen atoms. [1]

- Opening file: In Menu Bar → File → Read Molecule: protein.pdb
- Color by atom: In Dashboard → Click ◊ under "Atom"→ By atom type
- Eliminating waters: In Menu Bar → Select → SelectFromString → type HOH* in "Residue" → type * in "Atom" → Add → Dismiss → Edit → Delete → Delete Selected Atoms
- Find missing atoms and repairing them: In Menu Bar → Edit → Misc → Check for missing atoms → Edit → Misc → Repair for missing atoms
- Add hydrogens: In Menu Bar → Edit → Hydrogens → Add → choose "All hydrogens", "no bond order", and "Yes" to renumbering
- Hide protein: In Dashboard → Click on ⬭ the red in display Lines column

Preparing the ligand

Here we prepare the ligand for AutoDock that has to be in format pdbqt. If the ligand has assigned charges, these are used. But, if they are absent, ADT computes Gasteiger charges for the entire ligand.

- **Make sure to add all hydrogens by MOE or GaussView (or any other visualization program) before working with ADT**.[2]
- Opening file: Ligand → Input → Open: ligand.pdb
- Define torsions: Ligand → Torsion Tree → Detect Root => ADT identifies the central atom in the ligand for using as the root (the rigid part) and marks it with a green sphere.
- Ligand → Torsion Tree → Choose Torsions => This widget displays the number of currently active bonds. The maximum number can be 32. Amide bonds should NOT be active (pink color) → Done

[1] *Before running any docking calculation, be sure of the protonation state of the residues included in the binding site that you selected with the Grid box. Use Propka or experimental evidence in order to perform a docking calculation with fewer errors.*

[2] *Be cautious with the hydrogen atoms in the ligand, it will affect the charges that Autodock assigns, and also missing hydrogens or, on the other hand, more hydrogens will bias the results of the docking calculations.*

- Save ligand file: Ligand → Output → Save as PDBQT: **ligand.pdbqt**
- Hide the ligand (same as protein)

2.2 Preparing the grids

Here, we will prepare the 3D area to be searched during the AutoDock experiment. The search space is defined by specifying a center, the number of points in each dimension and the spacing between points.

To make the docking, AutoDock does not use the receptor, but it uses a set of "pre-calculated" maps generated by Autogrid. They include a map for each atom type of the ligand plus 2 extras: a "d" map for desolvatation and "e" map for electrostatics. In this way ADT can check if the molecule has assigned charges and also can determine the types of atoms in the macromolecule.

- Grid → Macromolecule → Choose: protein.pdb → Select Molecule → *type-in*: **protein.pdbqt** → Save
-
- Grid → Grid Box: Choose the space, size and the position of the box depending on the binding site of the chosen receptor.
- File → Close saving current
- Grid → Set Map types → Choose Ligand → ligand → Select ligand
- Grid → Output → Save GPF → type-in: **protein_grid.gpf**

2.3 Preparing the AutoDock Parameter File

Here we establish the search method together with other parameters. The DPF file contains these docking parameters and the instructions for the search algorithm docking.

- Docking → Macromolecule → Set Rigid Filename... → protein.pdbqt → Open
- Docking → Ligand → Choose: ligand → Select Ligand → Accept
- Docking → Search Parameters → Genetic Algorithm: normal, GA runs= 100 → Accept
- Docking → Output → Lamarckian GA: **ligand_docking.dpf** → Save

2.4 Running the docking calculation

We will launch the Grid and the Docking calculation via command line. Go to the folder that contains all the inputs for the calculation[3].

- autogrid4 –p protein_grid.gpf –l protein_grid.glg &
- autodock4 –p ligand_docking.dpf –l ligand_docking.dlg &

[3] *If you try to launch the Grid or the Docking calculations with AutoDock Tools, maybe (depending on the installation protocol) you will find a problem related to the working directory, because it may only accept the default. You can launch the calculations via command to avoid this.*

2.5 Visualizing AutoDock results with AutodockTools

The Docking log file (.dlg) contains all the results and we can visualize the docked poses, as well as the energies corresponding to each pose (see Annex II).

- Analyze → Docking→ Open: lig_docking.dlg → Open
- Analyze → Conformations → Load
- Analyze → Conformations → Play, ranked by energy
- Analyze → Docking → Show as Spheres
- Analyze → Docking → Show Interactions
- File → Exit

2.6 Visualizing AutoDock Results with Pymol

The Docking log file (.dlg) can be converted into different PDB files using the fromdlgtopdb.py script. You can analyze them with Pymol to see the interactions, distances, etc. The files, which will be created, are:

- A PDB file that contain all the poses.
- A result summary.txt, where you can see the different clusters, the ranking by energy, etc.
- Different PDB and PDBqt files containing the best pose for each cluster.

3. Docking exercise: Docking of donepezil in acetylcholinestase

The material contains a folder named **Exercise_Docking**, which contains two folders:

Donepezil_1EVE
Donepezil_1ACJ

We want to explore the potential binding mode of donepezil in the binding site of acetylcholinesterase. To this end, we plan to use to distinct X-ray crystallographic structures of this enzyme, 1EVE and 1ACJ, in order to examine the effect of structural changes on the predicted binding mode.

The two files contain 'cleaned' structures of the protein and the prepared (with addition of hydrogens according to the expected bioactive species at physiological pH) structure of donepezil.

The goal is to execute the docking in the two target templates using the protocol outline above and analyze the predicted poses using Pymol.

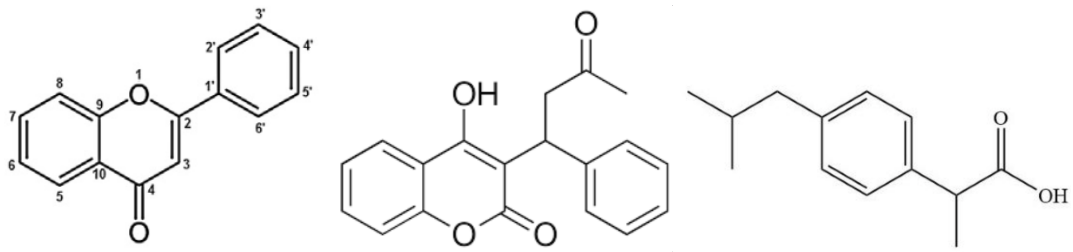
Do you get the same pose?

If not, can you infer the reason why the poses are dofferent in the two targets?

4. Homework

Human serum albumin (HSA) is the main extracellular protein, and is highly concentrated, in blood plasma. HSA is a monomeric globular protein composed of three structurally similar domains (I, II and III). Aromatic and heterocyclic ligands bind to HSA primarily within two hydrophobic pockets named Sites 1 and 2, respectively. Site 1 is the primary binding site for drugs like warfarin (Scheme 2), whereas ibuprofen (Scheme 2) is bound primarily to Site 2.

We want to explore whether 2-Phenylchromone (2Phe; Scheme 2), a flavone found in cereals and herbs indispensable in the human diet, can be transported through binding to sites 1 and 2.



Scheme 2. Structure of (left) 2-phenylchromone, (middle) warfarin, and (right) ibuprofen.

1) Dock warfarin in site 1 of HSA using the X-ray structure of the warfarin-HSA complex (PDB ID 2BXD).

Can the redocking with AutoDock reproduce the experimental binding mode?

Is the experimental binding mode found in the best scored pose?

What are the main interactions that mediate the binding for the predicted pose?

2) Now dock 2Phe in HSA using the X-ray structure 2BXD.

Do you think the predicted binding mode is acceptable?

How does the predicted affinity of 2Phe compare with the affinity of warfarin?

Do the binding affinities of warfarin and 2Phe reflect in your view the interactions formed by these compounds?

3) Finally dock 2Phe in HSA in site 2 using the X-ray structure of the ibuprofen-HSA complex (PDB ID 2BXG).

Do you thing 2Phe may compete with ibuprofen for binding at site 2?

Annex I. AutoDockTools

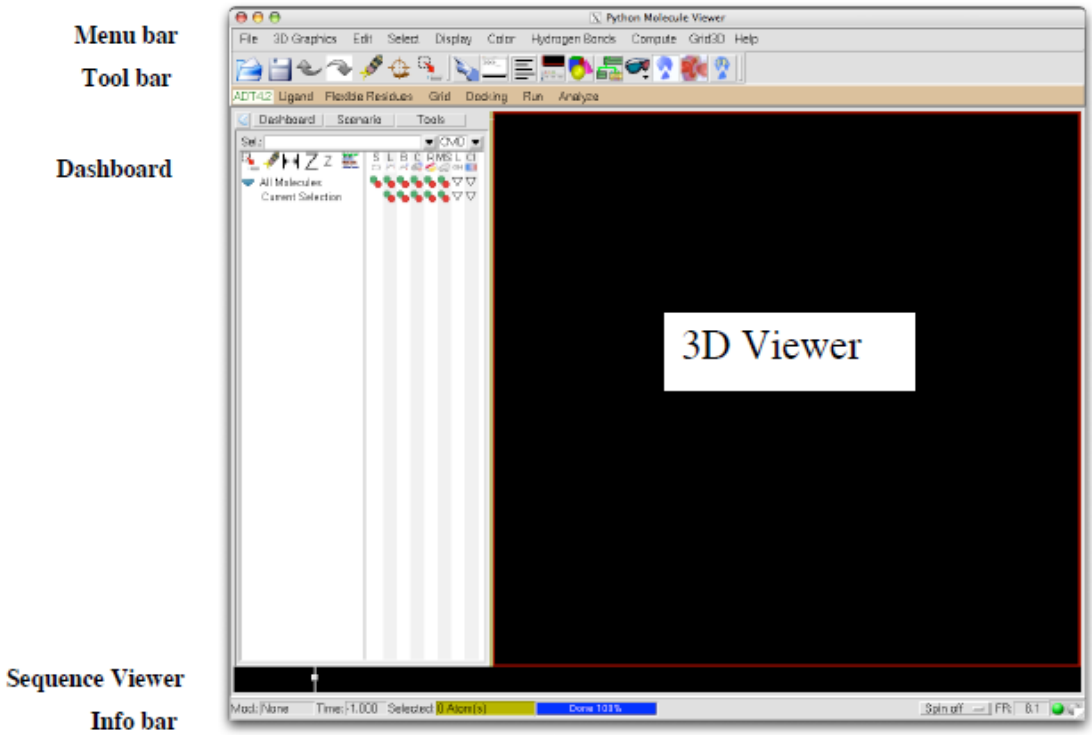
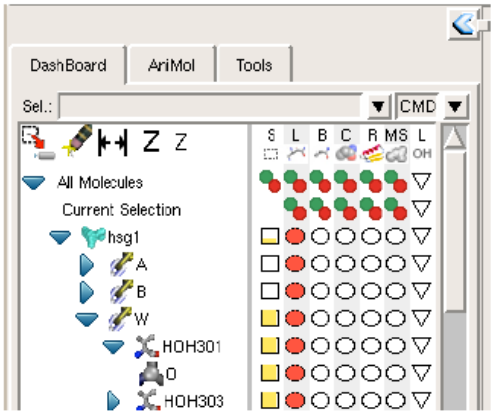


Figure 1. Initial configuration of ADT program with the different menus indicated.



The **Tree Widget** on the left lists all molecules currently loaded in PMV. Click on the arrows to navigate between molecules, chains, residues and atoms. Clicking on a **shape** in one of the columns in the right section executes the PMV command corresponding to the label at the top of the column on the group of nodes corresponding to the row. There 16 different commands that can be executed this way - gray rectangle (Show/Hide), select/unselect (**Sel.**), display lines (**Lines**), display CPK (**CPK**), display sticks and balls (**S&B**), display secondary structure (**Rib.**), display molecular surface (**MS**), display labels (**Lab.**), color by atom type (**Atom**), color by molecule (**Mol**), color by chain (**Chain**), color by residue according Rasmol (**RAS**), color by residue according Shapely (**SHA**), color according to David Goodsell colors (**DG**), color by secondary structure element type (**Sec.Str.**) and color by instance (**Inst**).

Figure 2. Dashboard widget explanations.

Annex II: Definition of parameters obtained for each docking pose.

- **Binding energy** is the sum of the intermolecular energy and the torsional free-energy penalty
- **Docking energy** is the sum of the intermolecular energy and the ligand's internal energy
- **Inhib_constant** is calculated in AutoDock as follows:

$$K_i = \exp((\Delta G*1000)(R*TK))$$

where ΔG is docking energy, R is 1.98719 $cal \cdot K^{-1} \cdot mol^{-1}$ and TK is 298.15

- **refRMS** is rms difference between current conformation coordinates and current reference structure. By default the input ligand is used as the reference.
- **clRMS** is rms difference between current conformation and the lowest energy conformation in its cluster
- **torsional_energy** is the number of active torsions * 0.3113 (0.3113 is autoDock 3 force field torsional free energy parameter)
- **rseed1** and **rseed2** are the specific random number seeds used for current confomation's docking run.