Automated Docking

Introduction

AutoDock is an automated procedure for predicting the interaction of ligands with biomacromolecular targets. The motivation for this work arises from problems in the design of bioactive compounds, and in particular the field of computer-aided drug design. Progress in biomolecular x-ray crystallography continues to provide important protein and nucleic acid structures. These structures could be targets for bioactive agents in the control of animal and plant diseases, or simply key to the understanding of fundamental aspects of biology. The precise interaction of such agents or candidate molecules with their targets is important in the development process. The goal has been to provide a computational tool to assist researchers in the determination of biomolecular complexes.

In any docking scheme, two conflicting requirements must be balanced: the desire for a robust and accurate procedure, and the desire to keep the computational demands at a reasonable level. The ideal procedure would find the global minimum in the interaction energy between the substrate and the target protein, exploring all available degrees of freedom (DOF) for the system. However, it must also run on a laboratory workstation within an amount of time comparable to other computations that a structural researcher may undertake, such as a crystallographic refinement. In order to meet these demands a number of docking techniques simplify the docking procedure. AutoDock combines two methods to achieve these goals: rapid grid-based energy evaluation and efficient search of torsional freedom.

The current version of AutoDock, using the Lamarckian Genetic Algorithm and empirical free energy scoring function, typically will provide reproducible docking results for ligands with approximately 10 flexible bonds. A related software, AutoDock Vina (http://vina.scripps.edu), uses a simpler scoring function that allows a faster search method, and provides reproducible results for larger systems with upwards of 20 flexible bonds.

AutoDock calculations are performed in several steps: 1) preparation of coordinate files using **AutoDockTools**, 2) precalculation of atomic affinities using **AutoGrid**, 3) docking of ligands using **AutoDock**, and 4) analysis of results using **AutoDockTools.**

Step 1—Coordinate File Preparation. AutoDock4.2 is parameterized to use a model of the protein and ligand that includes polar hydrogen atoms, but not hydrogen atoms bonded to carbon atoms. An extended PDB format, termed **PDBQT**, is used for coordinate files, which includes atomic partial charges and atom types. The current AutoDock force field uses several atom types for the most common atoms, including separate types for aliphatic and aromatic carbon atoms, and separate types for polar atoms that form hydrogen bonds and those that do not. PDBQT files also include information on the torsional degrees of freedom. In cases where specific sidechains in the protein are treated as flexible, a separate PDBQT file is also created for the sidechain coordinates. AutoDockTools, the Graphical User Interface for AutoDock, may be used for creating PDBQT files from traditional PDB files.

Step2—AutoGrid Calculation. Rapid energy evaluation is achieved by precalculating atomic affinity potentials for each atom type in the ligand molecule being docked. In the AutoGrid procedure the protein is embedded in a three-dimensional grid and a probe atom is placed at each grid point. The energy of interaction of this single atom with the protein is assigned to the grid point. AutoGrid affinity grids are calculated for each type of atom in the ligand, typically carbon, oxygen, nitrogen and hydrogen, as well as grids of electrostatic and desolvation potentials. Then, during the AutoDock calculation, the energetics of a particular ligand configuration is evaluated using the values from the grids.

Step 3—Docking using AutoDock. Docking is carried out using one of several search methods. The most efficient method is a Lamarckian genetic algorithm (LGA), but traditional genetic algorithms and simulated annealing are also available. For typical systems, AutoDock is run several times to give several docked conformations, and analysis of the predicted energy and the consistency of results is combined to identify the best solution.

Step 4—Analysis using AutoDockTools. AutoDockTools includes a number of methods for analyzing the results of docking simulations, including tools for clustering results by conformational similarity, visualizing conformations, visualizing interactions between ligands and proteins, and visualizing the affinity potentials created by AutoGrid.

Theory

Overview of the Free Energy Scoring Function

AutoDock 4.2 uses a semi-empirical free energy force field to evaluate conformations during docking simulations. The force field was parameterized using a large number of protein-inhibitor complexes for which both structure and inhibition constants, or K_i, are known.
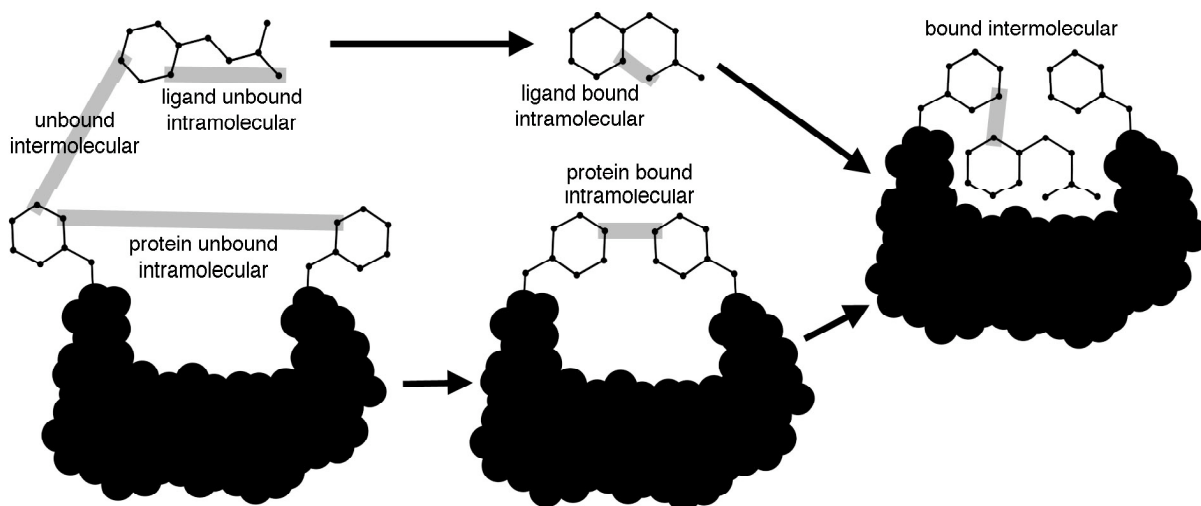
The force field evaluates binding in two steps. The ligand and protein start in an unbound conformation. In the first step, the intramolecular energetics are estimated for the transition from these unbound states to the conformation of the ligand and protein in the bound state. The second step then evaluates the intermolecular energetics of combining the ligand and protein in their bound conformation.

The force field includes six pair-wise evaluations (V) and an estimate of the conformational entropy lost upon binding (ΔS_{conf}):

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

where L refers to the "ligand" and P refers to the "protein" in a ligand-protein docking calculation.

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

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

The weighting constants W have been optimized to calibrate the empirical free energy based on a set of experimentally determined binding constants. The first term is a typical 6/12 potential for dispersion/repulsion interactions. The parameters are based on the Amber force field. The second term is a directional H-bond term based on a 10/12 potential. The parameters C and D are assigned to give a maximal well depth of 5 kcal/mol at 1.9Å for hydrogen bonds with oxygen and nitrogen, and a well depth of 1 kcal/mol at 2.5Å for hydrogen bonds with sulfur. The function E(t) provides directionality based on the angle t from ideal H-bonding geometry. The third term is a screened Coulomb potential for electrostatics. The final term is a desolvation potential based on the volume of atoms (V) that surround a given atom and shelter it from solvent, weighted by a solvation parameter (S) and an exponential term with distance-weighting factor σ=3.5Å. For a detailed presentation of these functions, please see our published reports, included in Appendix IV.

By default, AutoGrid and AutoDock use a standard set of parameters and weights for the force field. The `parameter_file` keyword may be used, however, to use custom parameter files. The format of the parameter file is described in Appendix I.

Several methods for estimating the contribution of the unbound state are implemented in AutoDock. In Autodock 3.0 and earlier versions, an assumption is made that the unbound form of the ligand (V^{L-L}_{bound} in the equation above) is the same as the final docked conformation of the ligand ($V^{L-L}_{unbound}$), yielding a final contribution V^{L-L}_{bound} -$V^{L-L}_{unbound}$ = 0. AutoDock 4.1 introduced a method of generating an extended form of the ligand to model the unbound state. Reports from users, however, revealed that the method caused significant problems with sterically-crowded molecules, and the default method was changed to the bound=unbound assumption in AutoDock 4.2 and later. In addition, there is an option for a user-defined unbound state.

Using AutoDock

STEP 1: Preparing Coordinates

The first step is to prepare the ligand and receptor coordinate files to include the information needed by AutoGrid and AutoDock. These coordinate files are created in an AutoDock-specific coordinate file format, termed PDBQT, which includes:

1) Polar hydrogen atoms;

2) Partial charges;

3) Atom types;

4) Information on the articulation of flexible molecules.

For a typical docking calculation, you will create a file of coordinates for the receptor, and a separate file of coordinates for the ligand. In dockings where selected amino acids in the receptor are treated as flexible, you will create a third file that includes the coordinates of the atoms in the flexible portions of the receptor.

In a typical study, the user prepares coordinate files in several steps using AutoDockTools. The first two steps may be performed using the tools in the Edit menu of AutoDockTools, or with other molecular modeling programs:

1) Add hydrogen atoms to the molecule.

2) Add partial charges.

Then, read the molecule into AutoDockTools using the `Ligand` (for the ligand) or `Grid` (for the receptor) menus, and create the PDBQT file:

3) Delete non-polar hydrogens and merge their charges with the carbon atoms.

4) Assign atom types, defining hydrogen bond acceptors and donors and aromatic and aliphatic carbon atoms.

5) Choose a root atom that will act as the root for the torsion tree description of flexibility.

6) Define rotatable bonds and build the torsion tree.

There are a few things to keep in mind during this process:
Be Critical. AutoDockTools and PMV currently use a modified version of Babel to add hydrogen atoms and assign charges. Unfortunately the method has trouble with some molecules. In those cases, hydrogen positions and charges may be assigned by the user's preferred method, *e.g.* using Reduce, InsightII, Quanta, Sybyl, AMBER or CHARMm.

Check Your Hydrogen Positions. In addition, most modeling systems add polar hydrogens in a default orientation, typically assuming each new torsion angle is 0° or 180°. Without some form

of refinement, this can lead to spurious locations for hydrogen bonds. One option is to relax the hydrogens and perform a molecular mechanics minimization on the structure. Another is to use a program like "pol_h" which takes as input the default-added polar hydrogen structure, samples favorable locations for each movable proton, and selects the best position for each. This "intelligent" placement of movable polar hydrogens can be particularly important for tyrosines, serines and threonines.

Watch for Disordered Residues. Care should be taken when the PDB file contains disordered residues, where alternate location indicators (column 17) have been assigned. For each such atom, the user must select only one of the possible alternate locations, making sure that a locally consistent set is chosen.

Randomize Starting Coordinates When Redocking. In redocking experiments, where coordinates from a known protein-ligand complex are separated and docked, it is important to randomize the conformation of the ligand before it is docked. The techniques used to implement rotation of the ligand may be biased to prefer values of zero rotation, so to obtain an unbiased result, create a pdbqt file with the randomized ligand coordinates using the "Randomize" command in the ADT Ligand menu, then use this randomized coordinate file for docking experiments.

Please note: coordinate preparation is the most important step in the docking simulation. The quality and accuracy of the docked results will only be as good as the quality of the starting coordinates. Be critical and carefully examine hydrogen positions, atom type assignments, partial charges, and articulation of the molecules to ensure that they make sense chemically. If you are using the Babel method within AutoDockTools to add charges and hydrogens, carefully check the results and make corrections if necessary—it often has trouble with molecules such as nucleotides.


Creating PDBQT files in AutoDockTools

Overview of AutoDockTools

AutoDockTools is a set of commands implemented within the Python Molecular Viewer (PMV), providing a Graphical User Interface for AutoGrid and AutoDock.

The AutoDockTools window has several parts:

1) at the top are menus that access the general methods available in PMV. These include tools for reading and writing coordinates and images, for modifying coordinates, for selection, and for visualization.

2) a row of buttons at the top allows quick access to the most popular tools of PMV

3) below the buttons, there are a series of menus that access the AutoDock-specific tools of AutoDockTools.

4) the 3-D molecular viewer is at center right.

5) the Dashboard, located to the left of the viewer, allows quick selection, visualization, and coloring of molecules currently displayed in the viewer.
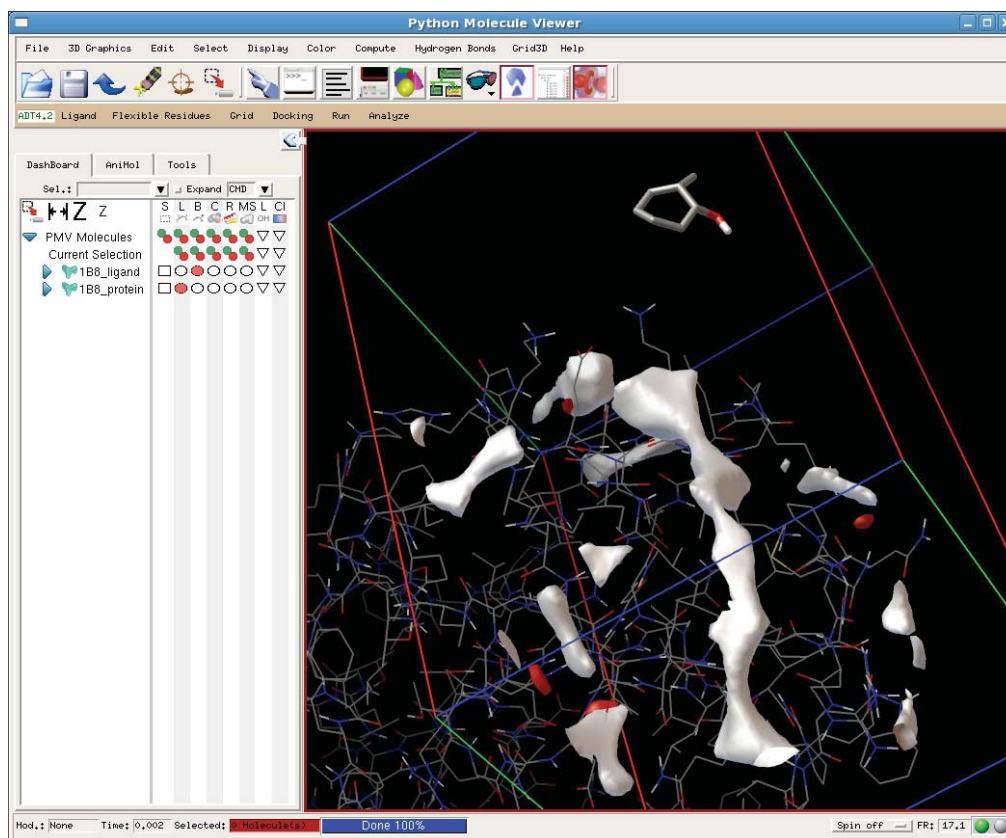
File> Preferences>Set>Startup Directortory > C:\Users\User Name\Desktop \Tutorial_Files

Hydrogen Atoms and Charges

The tools available in PMV are used to read coordinates in PDB and other formats, to add hydrogens, to select portions of the molecule, and to add partial charges. These functions are all accessed through menus at the top of the PMV window. A few useful commands will be described here—for more information on the many other functions of PMV, please see the PMV documentation.

`File>ReadMolecule:` opens a browser that allows reading of PDB coordinate files (receptor.pdb).

`Edit>Delete Water:` to delete the water molecules.

`Edit>Hydrogens>Add:` options for adding all hydrogens or polar hydrogens using Babel.

`Edit>Charges>Compute Gasteiger :` options for computing Gasteiger charges for

arbitrary molecules using Babel. Now hide the receptor moleule from the dashboard.

Ligand PDBQT Files – the "Ligand" Menu

Once ligand coordinates are created with hydrogen atoms and charges, they can be processed in the "Ligand" menu to create the ligand PDBQT file.

`Ligand>Input>Open:` reads coordinates from a file (lig.pdb).

`Ligand>TorsionTree>ChooseRoot:` manual selection of the root atom.

`Ligand>TorsionTree>DetectRoot:` automatic detection of the root that provides the smallest largest subtree.

`Ligand>TorsionTree>ChooseTorsions:` launches an interactive browser for choosing rotatable bonds. Rotatable bonds are shown in green, and non-rotatable bonds are shown in red. Bonds that are potentially rotatable but treated as rigid, such as amide bonds and bonds that are made rigid by the user, are shown in magenta. Rotation of rotatable bonds may be switched on and off by clicking on the bonds.

Ligand>TorsionTree>SetNumberOfTorsions: sets the number of rotatable bonds in the ligand by leaving the specified number (10) of bonds as rotatable. The two options will choose the torsions that rotate either the fewest atoms in the ligand or the most atoms in the ligand.

Ligand>Output: opens a browser to write the formatted PDBQT file.

Rigid Receptor PDBQT Files – the "Grid" Menu

For docking calculations using rigid receptor coordinates, add the hydrogen atoms and charges in PMV, then read the coordinates into AutoDockTools using the "Grid" menu.

Grid>Macromolecule>Choose: chooses a molecule that has been previously read into PMV. It will merge non-polar hydrogen atoms and charges, assign aromatic carbons, and prompt the user to write a PDBQT file.

Creating grid parameter files in AutoDockTools

The tools available in "grid" menu of AutoDockTools may be used to create grid parameter files.

`Grid>Macromolecule:` has options for opening an existing PDBQT file or choosing a molecule that has been read using PMV.

`Grid>SetMapTypes:` tools to define the atom types for the grids that will be calculated. Grids must be calculated for each type of atom in the ligand, and if flexible sidechains are used in the receptor, their atom types must also be included. The option "Directly" allows the user to input the list of atom types directly. Other options allow the user to define the atom types based on a ligand or flexible residue that has been read by PMV, or to open ligand or flexible residue PDBQT and use the atom types in these files.

`Grid>GridBox:` launches interactive commands for setting the grid dimensions and center. To enter numbers on the thumbwheel, place the cursor over the thumbwheel and type in the new value. Make it 126, 126 , 126 to cover the entire molecule. Right clicking on the thumbwheel gives more options. IMPORTANT: when finished, use the "close saving current" option in the "File" menu on the Grid Options Panel. Options in the "Center" menu on the browser provide different methods to choose the center of the grid box.

`Grid>Output:` writes a new grid parameter file (grid.gpf).

STEP 2: Running AutoGrid

AutoDock requires pre-calculated *grid maps*, one for each atom type present in the ligand being docked. This helps to make the docking calculations fast. These maps are calculated by **AutoGrid**. A grid map consists of a three-dimensional lattice of regularly spaced points, surrounding (either entirely or partly) and centered on some region of interest of the macromolecule under study. This could be a protein, enzyme, antibody, DNA, RNA or even a polymer or ionic crystal. Typical grid point spacing varies from 0.2Å to 1.0Å, and the default is 0.375Å (roughly a quarter of the length of a carbon-carbon single bond). Each point within the grid map stores the potential energy of a '*probe*' atom or functional group that is due to all the atoms in the macromolecule.

AutoGrid requires a grid parameter file to specify the files and parameters used in the calculation. The grid parameter file usually has the extension ".gpf". As described below, AutoDockTools may be used to create the grid parameter file. A full description of the grid parameter file is included in Appendix I.

Run>Run AutoGrid> Set the program path name as
C:/Program Files (x86)/The Scripps Research Institute/Autodock/4.2.6/autogrid4.exe
Set the parameter file name as
C:\Users\User Name\Desktop\Tutorial_Files\grid.gpf
Set the log file name as
C:\Users\User Name\Desktop\Tutorial_Files\grid.glg

So, finally to run AutoGrid, the command is issued as follows:
C:/Program Files (x86)/The Scripps Research Institute/Autodock/4.2.6/autogrid4.exe -p C:\Users\User Name\Desktop\Tutorial_Files\grid.gpf -l C:\Users\User Name\Desktop\Tutorial_Files\grid.glg

where '-p grid.gpf' specifies the grid parameter file, and '-l grid.glg' specifies the log file written during the grid calculation. If no log file is specified, the output is written to the terminal.

As well as the grid maps, AutoGrid creates two files, with the extensions '.fld', and '.xyz'. The former is a *field file* summarizing the grid maps, and the latter describes the spatial extent of the grids in Cartesian space.

Creating docking parameter files in AutoDockTools

The tools available in the "Docking" menu of AutoDockTools may be used to create docking parameter files.

`Docking>Macromolecule>SetRigidFilename:`

`Docking>Ligand>Choose`

`Docking>SearchParameters>GeneticAlgorithmParameters:` Set maximum number of evaluation to short.

`Docking>DockingParameters:` opens a panel for setting the parameters used during the docking calculation, including options for the random number generator, options for the force field, step sizes taken when generating new conformations, and output options. Keep all the values default for the tutorial.
For details of each parameter, see the full description in the Appendix I.

`Docking>Output>LamarckianGA:` This command writes the docking parameter file using one of the four available search methods.

Save the file as docking.dpf

STEP 3: Docking with AutoDock

AutoDock uses one of several conformational search algorithms to explore the conformational states of a flexible ligand, using the maps generated by AutoGrid to evaluate the ligand-protein interaction at each point in the docking simulation. In a typical docking, the user will dock a ligand several times, to obtain multiple docked conformations. The results may be clustered to identify similar conformations—this is described in more detail in the section on Analysis (Step 4, below).

AutoDock requires: 1) grid maps for each atom type in the ligand, calculated by AutoGrid, 2) a PDBQT file for the ligand, and 3) a docking parameter file that specifies the files and parameters for the docking calculation. AutoDockTools may be used to generate the docking parameter file, as described below, which typically has the extension ".dpf". A full description of the docking parameter file is included in Appendix I. AutoDock writes the final docked coordinates into the docking log file. As described in Step 4 below, these docked conformations may be viewed using AutoDockTools, they may be written as PDBQT files using AutoDockTools, or they may be taken directly from the docking log file using a text editor.

Run>Run AutoDock> Set the program path name as
C:/Program Files (x86)/The Scripps Research Institute/Autodock/4.2.6/autodock4.exe
Set the parameter file name as
C:\Users\User Name\Desktop\Tutorial_Files\docking.dpf
Set the log file name as
C:\Users\User Name\Desktop\Tutorial_Files\docking.dlg

So, finally to run AutoGrid, the command is issued as follows:
C:/Program Files (x86)/The Scripps Research Institute/Autodock/4.2.6/autodock4.exe -p C:\Users\User Name\Desktop\Tutorial_Files\docking.dpf -l C:\Users\User Name\Desktop\Tutorial_Files\docking.dlg

Input parameters are specified by "-p docking.dpf", and the log file containing the output and results from the docking is defined by "-l docking.dlg". This is the normal usage of AutoDock, and performs a standard docking calculation.

STEP 4: Evaluating the Results of a Docking

At the end of a docking simulation, AutoDock writes the coordinates for each docked conformation to the docking log file, along with information on clustering and interaction energies. AutoDockTools provides options for analyzing the information stored in the docking log file.

Information in the Docking Log

The **analysis** command in the docking parameter file causes AutoDock to perform a cluster analysis of the different docked conformations – the minimum energy found in each run. The results of this analysis are reported as a histogram, which may be found by searching for the word "HISTOGRAM" (all in capital letters) in the docking log file. This is followed by a table of RMSD values within each cluster.

AutoDock then writes coordinates for the conformation of best predicted energy in each cluster (to write coordinates for the best conformation from each run, include the keyword **write_all** in the docking parameter file). A header for each conformation includes information on the predicted energy of binding, broken down into several components, along with information on the state variables of the conformation. The coordinates are written in a modified PDB format, with four real values appended after the x,y,z coordinates: the vdW+hbond+desolvation energy of interaction of the atom, the electrostatic interaction of the atom, the partial charge, and the RMSD from the reference conformation.

The amount of information included in the docking log may be controlled using the "outlev" parameter, ranging from very minimal output for large virtual screens, to the default that is read by ADT (with information on docked conformations and clustering), to more verbose files that are useful for methods development.

Analyzing Docking Results with AutoDockTools

Options in the "Analyze" menu of AutoDockTools may be used to process and analyze the results from a docking simulation.

Analyze>Dockings>Open: opens a docking log file (docking.dlg).

`Analyze>Conformations>Play`

`Analyze>Conformations>PlayRankedByEnergy:` Opens a window with controls for stepping through conformations as a movie. "Play" will use the order of conformations as they were found in the docking calculations, and "PlayRankedByEnergy" will order the conformations from lowest energy to highest energy. The "&" button opens a window with additional options:

`ShowInfo` opens a panel that displays information on the predicted energy of interaction, RMSD, etc.

`BuildHbonds` and `ColorbyATOM/vdW/elect/total` allow visualization of hbonds and interaction energies.

`PlayMode` and `PlayParameters` modify the parameters of the player.

`BuildCurrent` will build a new set of coordinates in the viewer for the conformation currently specified in the player. This is useful for displaying multiple conformations in the

same view. `BuildAll` will build coordinates for all conformations in the player.

`WriteCurrent` will write a PDBQT file for the current conformation in the player. `WriteAll` will write separate PDBQT files for all conformations in the player. `WriteComplex` will write a PDBQT file for the current conformation of the ligand and the receptor.

`Analyze>Dockings>ShowInteractions:` It creates a specialized visualization to highlight interactions between the docked conformation of the ligand and the receptor. By default, the ligand is shown as ball-and-stick, surrounded by a molecular surface. The surface is colored with atomic colors in regions that contact the receptor, and gray in regions that are not in contact. Portions of the receptor that are in contact with the ligand are shown with ball-and-stick and spacefilling spheres. Hydrogen bonds are shown as a string of small spheres. A dialogue box is also launched that provides many other options for visualization.