

Introduction to AutoDock and AutoDock Tools

Alexander B. Pacheco

User Services Consultant
LSU HPC & LONI
sys-help@loni.org

HPC Training Series
Louisiana State University
Baton Rouge
Mar. 28, 2012



- 1 Introduction
- 2 Using AutoDock Tools to Create Input Files
 - Editing a PDB File
 - Preparing the Ligand File for AutoDock
 - Preparing the Macromolecular File
 - Preparing the Grid Parameter File
 - Preparing the Docking Parameter File
- 3 Running AutoGrid/AutoDock Calculations on LSU HPC and LONI
 - AutoDock and AutoGrid
 - Running AutoGrid and AutoDock
- 4 Analysis Using AutoDock Tools



- 1 Introduction
- 2 Using Autodock Tools to Create Input Files
 - Editing a PDB File
 - Preparing the Ligand File for AutoDock
 - Preparing the Macromolecular File
 - Preparing the Grid Parameter File
 - Preparing the Docking Parameter File
- 3 Running AutoGrid/AutoDock Calculations on LSU HPC and LONI
 - AutoDock and AutoGrid
 - Running AutoGrid and AutoDock
- 4 Analysis Using AutoDock Tools

- Interaction between biomolecules lie at the core of all metabolic processes and life activities.
- The number of solved protein structures available in the databases is expanding exponentially.
- To understand their functions it is essential to elucidate the interaction mechanisms between the different molecules.
- e.g. drug design, antigen-antibody interaction

- Macromolecular docking is the computational modelling of the quaternary structure of complexes formed by two or more interacting biological macromolecules.
- In any docking scheme, the desire for a robust and accurate procedure and the desire to keep the computational demands at a reasonable level must be balanced.
- 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 of the system.
- The simulation time should be reasonable compared to other computation that a structural researcher may undertake, such as crystallographic refinement.
- Simply put, Molecular Docking is a computational procedure for predicting the best way(s) two molecules will interact
- Procedure consists of:
 - 1 Obtain structures of the two molecules.
 - 2 Locate the best binding site(s).
 - 3 Determine the best binding mode(s).



- A number of docking techniques have been developed to simplify the docking procedure.
- AutoDock is an automated procedure for predicting the interactions of ligands with biomolecular targets which are key to understanding fundamental aspects of biology.
- The goal of Autodock is to provide a computational tools to assist researchers in the determination of biomolecular complexes.
- AutoDock combines two methods to achieve these goals:
 - 1 rapid grid-based energy evaluation
 - 2 efficient search of torsional freedom

Download Files from

<http://www.hpc.lsu.edu/training/tutorials/#spring2012-autodock>



- 1 Introduction
- 2 Using Autodock Tools to Create Input Files
 - Editing a PDB File
 - Preparing the Ligand File for AutoDock
 - Preparing the Macromolecular File
 - Preparing the Grid Parameter File
 - Preparing the Docking Parameter File
- 3 Running AutoGrid/AutoDock Calculations on LSU HPC and LONI
 - AutoDock and AutoGrid
 - Running AutoGrid and AutoDock
- 4 Analysis Using AutoDock Tools

- Autodock Tools or ADT is a graphical user interface created by the developers of Autodock, which amongst other things helps to set up which bonds will be treated as rotatable in the ligand and to analyze dockings.
- With ADT, you can:
 - 1 View molecules in 3D, rotate & scale in real time.
 - 2 Add all hydrogens or just non-polar hydrogens.
 - 3 Assign partial atomic charges to the ligand and the macromolecule (Gasteiger or Kollman United Atom charges).
 - 4 Merge non-polar hydrogens and their charges with their parent carbon atom.
 - 5 Set up rotatable bonds in the ligand using a graphical version of AutoTors.
 - 6 Set up the AutoGrid Parameter File (GPF) using a visual representation of the grid box, and slider-based widgets.
 - 7 Set up the AutoDock Parameter File (DPF) using forms.
 - 8 Launch AutoGrid and AutoDock.
 - 9 Read in the results of an AutoDock job and graphically display them.
 - 10 View isocontoured AutoGrid affinity maps.
 - 11 And much, much more...
- Can be downloaded from <http://mgltools.scripps.edu/downloads>
 - ◆ If you haven't already, please download and install ADT now. The Hands-On Exercise that follows makes extensive use of ADT.



- An AutoDock calculation consists of multiple steps most of which need to be done using ADT
 - 1 Prepare ligand and receptor (macromolecule) for docking,
 - 2 Define grid for docking,
 - 3 Run a AutoGrid calculation,
 - 4 Define docking parameters,
 - 5 Run AutoDock calculation,
 - 6 Analyze Results.
- Each docking requires at least four input files:
 - 1 a PDBQT file for the ligand that encodes a torsion tree,
 - 2 a PDBQT file for the receptor,
 - 3 a grid parameter file (GPF) for AutoGrid calculation, and
 - 4 a docking parameter file (DPG) for AutoDock calculation.
 - ◆ If some residues in the receptor are to be modeled as flexible, a fifth file will be necessary
 - 5 a PDBQT file containing the flexible receptor residues.



Why?

Protein Data Bank (PDB) files may have a variety of problems, such as

- missing atoms, added waters, more than one molecule, chain breaks, alternate locations, *etc.*

that need to be corrected before they can be used in AutoDock.

1 Open ADT

2 Read in hsg1.pdb

3 Add waters

Select→Select From String

type HOH* in the Residue entry

Click on Add and Yes for warning message

On ADT window, you should see Selected: 127 Atoms

4 Click on Dismiss to close the Select From String widget.

5 Delete Oxygen and Add Hydrogens

Edit→Delete→Delete AtomSet→Continue

Edit→Hydrogens→Add

6 Choose to all All Hydrogens using the method noBondOrder with yes to renumbering and click OK

1612 hydrogen atoms should be added to hsg1

7 Save PDB File as hsg1.pdb

File→Save→Write PDB



- AutoDock requires that ligands have partial atomic charges and AutoDock atom types for each atom; it also requires a description of the rotatable bonds in the ligand.
- AutoDock uses the idea of a tree in which the rigid core of the molecule is a 'root', and the flexible parts are 'branches' that emanate from the root.
- Ligands are written in PDBQT-formatted files with special keywords recognized by AutoDock.
- The keywords ROOT, ENDROOT, BRANCH, and ENDBRANCH establish this torsion tree.
- The keyword TORSDOF specifies the number of torsional degrees of freedom in the ligand.
- In the AutoDock 4 force field, the TORSDOF value for a ligand is the total number of rotatable bonds in the ligand. This number excludes bonds in rings, bonds to leaf atoms, amide bonds, guanidinium bonds, etc.
- TORSDOF is used in calculating the change in free energy caused by the loss of torsional degrees of freedom upon binding.



1 Read ind.pdb

Ligand→Input→Open

2 Detect Torsion Tree Root

Ligand→Torsion Tree→Detect Root

3 Choose Torsions

Ligand→Torsion Tree→Choose Torsions

- ◆ This opens the Torsion Count widget which displays current active bonds.
- ◆ Bonds that cannot be rotated are colored red
- ◆ Bonds that could be rotated but are currently inactive are colored purple
- ◆ Bonds that are currently active are colored green
- ◆ Bonds in cycles cannot be rotated
- ◆ Only single bonds can be rotated
- ◆ The widget has buttons that let you toggle the activity bonds
- ◆ Amide bonds should not be rotatable.

4 Set number of Torsions

Ligand→Torsion Tree→Set Number of Torsions

Set fewest atoms to 6

5 Save pdbqt file as ind.pdbqt

Ligand→Output→Save as PDBQT



- AutoDock3 docks a flexible ligand to a rigid receptor
- AutoDock4 adds support for including a conformational search of flexible sidechains of designated residues to the receptor.
- The flexibility pattern in the moving sidechains of the residues is encoded with AutoDock4 specific keywords BEGIN_RES and END_RES as well as ROOT, ENDROOT, BRANCH and ENDBRANCH.
- The flexible residues are written in separate PDBQT file.
- The only rigid residues are written in the file for AutoGrid calculations.



1 Choose hsg1 as the macromolecule to have flexible residues
Flexible Residues→Input→Choose Macromolecule

2 Select the residues to be flexible
Select→Select From String
Type ARG8 in the Residue entry and click on Add

Check that 2 Residue(s) appear in the Selected: entry below

3 Define the rotatable bonds in the selected residues.

Flexible Residue→Choose Torsions in Currently Selected Residues

- ◆ All non selected residues in the macromolecule are hidden.
- ◆ Sidechains of selected residues are shown as colored green (rotatable bonds), red (unrotatable bonds) and magenta (non-rotatable bonds)
- ◆ Clicking a rotatable bond makes it non-rotatable while clicking a non-rotatable bond makes it rotatable.

Click on the rotatable bond CA and CB in each residue to inactivate it.

4 Save flexible residue as hsg1_flex.pdbqt and rigid residue as hsg1_rigid.pdbqt

Flexible Residues→Output→Save Flexible PDBQT

Flexible Residues→Output→Save Rigid PDBQT

5 Remove this version of hsg1



- The grid parameter file tells AutoGrid4 which receptor to compute the potentials around, the types of maps to compute and the location and extent of those maps.
- It may specify a custom library of pairwise potential energy parameters.
- In general, one map is calculated for each atom type in the ligand plus an electrostatics map and a separate desolvation map.



- 1 Open `hsg1_rigid.pdbqt` to prepare grid parameter file
Grid→Macromolecule→Open
- 2 Choose ligand file, `ind`
Grid→Set Map Types→Choose Ligand
- 3 Create Grid
Grid→Grid Box
Adjust number of points to 60 60 66, each map will have 249,307 points
Type in 2.5 6.5 and -7.5 in the x, y and z- center entries. This will center the grid on the active site of the HIV-1 protease, `hsg1`
- 4 Close this widget File→Close saving current
- 5 Save Grid Parameter File as `hsg1.gpf`
Grid→Output→Save GPF

- The docking parameter file tells AutoDock
 - which map files to use,
 - the ligand molecule to move,
 - what is the center and number of torsions are,
 - where to start the ligand,
 - the flexible residues to move if sidechain motion in the receptor is to be modeled,
 - which docking algorithm to use and how many runs to do
- Four different docking algorithm are currently available in AutoDock
 - 1 SA, the original Monte Carlo simulated annealing,
 - 2 GA, a traditional Darwinian genetic algorithm,
 - 3 LS, local search,
 - 4 GALS, hybrid genetic algorithm with local search also known as Lamarckian genetic algorithm or LGA
- Each search method has its won set of parameter which must be set before running the docking experiment.
- Parameters include what kind of random number generator to use, step sizes, etc. with the most important parameters affect how long each docking will run.
- In SA, the number of temperature cycles, the number of accepted moves and number of rejected moves determine how long a docking will take.
- In the GA and GALS, the number of evaluations and the number of generations affect how long a docking will run.



- 1 Choose Rigid Macromolecule, `hsg1_rigid.pdbqt`
Docking→Macromolecule→Set Rigid Filename
- 2 Choose Ligand, `ind`
Docking→Ligand→Choose
- 3 Choose Flexible Residue, `hsg1_flex.pdbqt`
Docking→Macromolecule→Set Flexible Residues
- 4 Set Search Parameters
Docking→Search Parameters→Genetic Algorithm, choose a short setting ie 250,000 energy evaluations
- 5 Choose docking parameters, for exercise accept default values
Docking→Docking Parameters
- 6 Save output as `ind.dpf`
Docking→Output→Lamarckian GA



- 1 Introduction
- 2 Using Autodock Tools to Create Input Files
 - Editing a PDB File
 - Preparing the Ligand File for AutoDock
 - Preparing the Macromolecular File
 - Preparing the Grid Parameter File
 - Preparing the Docking Parameter File
- 3 Running AutoGrid/AutoDock Calculations on LSU HPC and LONI
 - AutoDock and AutoGrid
 - Running AutoGrid and AutoDock
- 4 Analysis Using AutoDock Tools



- AutoDock 4.2 is installed on all LONI x86 clusters and the LSU HPC Tezpur cluster.
- Log into one of them on which you have an account.
- From Mac and Linux terminal, ssh <clustername>.loni.org or ssh tezpur.hpc.lsu.edu
- On Windows, use Putty to login to the clusters
- Add soft key for autodock, +autodock-4.3-gcc-4.3.2 to your .soft file
- resoft, to add autodock to your path
- Copy the pdbqt, gpf and dpf files that you just created
- AutoGrid and AutoDock must be run in the directory where the rigid macromolecule, ligand and parameter files are present.
- On QueenBee, all my input files are saved in /home/apacheco/CompChem/AutoDock
- Request an interactive queue
qsub -I -l walltime=1:00:00,nodes=1:ppn=4 -q checkpt



Running jobs interactively

```
[apacheco@qb4 ~]$ qsub -I -l walltime=1:00:00,nodes=1:ppn=8 -q checkpt
```

```
...
```

```
-----  
Running PBS prologue script
```

```
-----  
User and Job Data:
```

```
-----  
...
```

```
[apacheco@qb667 ~]$ cd CompChem/AutoDock/
```

```
[apacheco@qb667 AutoDock]$ autogrid4 -p hsg1.gpf -l hsg1.glg
```

```
autogrid4: Successful Completion.
```

```
[apacheco@qb667 AutoDock]$ autodock4 -p ind.dpf -l ind.dlg
```

```
autodock4: Successful Completion on "qb667"
```



- In general, you should write a submit script to run autodock and autogrid jobs and submit it to the job manager.
- The AutoGrid job should be completed before the AutoDock job starts.

AutoGrid submit script

```
[apacheco@qb4 AutoDock]$ cat submit_grid.pbs
#!/bin/bash
#
#PBS -q checkpt
#PBS -l nodes=1:ppn=8
#PBS -l walltime=20:00
#PBS -o autogrid.err
#PBS -e autogrid.err
#PBS -j oe
#PBS -N AutoGrid4

export EXEC=autogrid4
export INPUT=hsg1.gpf
export OUTPUT=hsg1.glg
export WORK_DIR=$PBS_O_WORKDIR

cd $WORK_DIR

$EXEC -p $INPUT -l $OUTPUT
[apacheco@qb4 AutoDock]$ qsub submit_grid.pbs
```

AutoDock submit script

```
[apacheco@qb4 AutoDock]$ cat submit_dock.pbs
#!/bin/bash
#
#PBS -q checkpt
#PBS -l nodes=1:ppn=8
#PBS -l walltime=20:00
#PBS -o autodock.err
#PBS -e autodock.err
#PBS -j oe
#PBS -N AutoDock4

export EXEC=autodock4
export INPUT=ind.dpf
export OUTPUT=ind.dlg
export WORK_DIR=$PBS_O_WORKDIR

cd $WORK_DIR

$EXEC -p $INPUT -l $OUTPUT
[apacheco@qb4 AutoDock]$ qsub submit_dock.pbs
```



The following script would allow you to submit both jobs at the same time.

```
[apacheco@qb4 AutoDock]$ cat submit_both
#!/bin/bash

FIRST=`qsub submit_grid.pbs`
echo $FIRST
SECOND=`qsub -W depend=afterok:$FIRST submit_dock.pbs`
echo $SECOND
exit 0
```

Note: For AutoDock job to run, the AutoGrid job should be completed without errors. It is your responsibility to check for this, do not email me if autodock4 doesn't run.

Usage: ./submit_both

- 1 Introduction
- 2 Using Autodock Tools to Create Input Files
 - Editing a PDB File
 - Preparing the Ligand File for AutoDock
 - Preparing the Macromolecular File
 - Preparing the Grid Parameter File
 - Preparing the Docking Parameter File
- 3 Running AutoGrid/AutoDock Calculations on LSU HPC and LONI
 - AutoDock and AutoGrid
 - Running AutoGrid and AutoDock
- 4 Analysis Using AutoDock Tools



- Key results in a docking logfile
 - 1 docked structures found at the end of each run,
 - 2 energies of these structures,
 - 3 similarities between the docked structures measured by computing the root-mean-square (rms) deviation.
- Docking results consist of the PDBQ of the cartesian coordinates of the atoms in the docked molecule, along with state variables that describe this docked conformation and position.



1 Choose AutoDock log file to analyze, ind.dlg

Analyze→Dockings→Open

Reading a docking log creates a Docking instance

A Conformation instance is created for each docked result in the docking log.

A Conformation represents a specific state of the ligand and has either a particular set of state variables from which all the ligand atom coordinates can be computed or the coordinates themselves.

Conformations also have energies: docked energy, binding energy and possibly per atom electrostatic and vdw energies.

2 Load Conformation Chooser

Analyze→Conformations→Load

This gives a concise view of energies and clusters of the docked results.



- The 'best' docking result can be considered to be the conformation with the lowest (docked) energy or one selected based on its rms deviation from a reference structure.

- ④ Open a conformation player to examine docked conformations

Analyze→Conformations→Play

Conformation 0 is the original, input conformation.

- An AutoDock experiment has several solutions.
- The reliability of a docking result depends on the similarity of its final docked conformations.
- One way to measure the reliability of the result is to compare the rmsd of the lowest energy conformations and their rmsd to one another to group them into families of similar conformations or 'clusters'

1 Analyze→Clusterings→Show

This opens an histogram chart, bars represent the clusters computed at the specified rmsd. The histogram bars are sorted by energy of the lowest energy conformation in that cluster and start off colored blue.

2 Analyze→Clusterings→Recluster

Opens a widget which lets you enter a series of new rms tolerances as floating point numbers separated by spaces.