BCB410 Protein-Ligand Docking Exercise Set

Shirin Shahsavand

December 11, 2011

1. Describe the search algorithm(s) AutoDock uses for solving protein-ligand docking problems.

AutoDock uses 3 different approaches for searching for the best conformation: local (Solis & Wets), global (Simulated Annealing and Genetic Algorithm), and global-local (glocal—Lamarckian Genetic Algorithm). The most recent and the most sophisticated approach is the glocal approach using LGA, which is very similar to GA. (The global approach of SA is also discussed in next 2 questions). I will, therefore, explain GA and LGA approach. The majority of GA's mimic the major characteristics of Darwinian evolution and apply Mendelian genetics. They have a one-way transfer of information from the genotype to the phenotype. However, in those cases where an inverse mapping function exists i.e., one that yields a genotype from a given phenotype, it is possible to finish a local search by replacing the individual with the result of the local search. This is called the Lamarckian genetic algorithm LGA.

Genetic Algorithm (GA):

GA applies a set of genetic operations to a population of solution candidates (individuals) of an optimization problem, iteratively producing better results.

The algorithm starts by creating an initial random population of fixed size. Afterwards, the objective function is evaluated for each individual to calculate its fitness score. Individuals with the worst scores are discarded, while the remaining solution candidates may create progeny. Mutation modifies existing individuals; new individuals are produced by mating. However, mutations are not applied to a number of top–ranked individuals, which is called elitism. When the population has grown again to the fixed size, the algorithm iterates until a convergence criterion is met.

The following is the GA that AutoDock uses for searching.

- *Start with a random population (50-300)*
- ***Genes** correspond to **state variables***
- *Perform genetic operations*
  - ***Crossover***
    - *1-point crossover, ABCD + abcd → Abcd + aBCD*
    - *2-point crossover, ABCD + abcd → AbCD + aBcd*
    - *uniform crossover, ABCD + abcd → AbCd + aBcD*
    - *arithmetic crossover, ABCD + abcd →* $[\alpha\ ABCD + (1-\alpha)\ abcd] + [(1-\alpha)\ ABCD + \alpha\ abcd]$ *where:* $0 < \alpha < 1$
  - ***Mutation***
    - *Add/subtract a random amount from randomly selected genes, A→ A'*
- *Compute the **fitness** of individuals (energy evaluation)*
- *Proportional Selection & Elitism*
- *If total energy evaluations or maximum generations reached, stop*

Lamarckian Genetic Algorithm (LGA):

In LGA, a dedicated local search procedure is applied to improve the fitness of existing individuals. It allows local optimization of the Phenotype, which is then applied to the Genotype. The advantage is that it does not require gradient information in order to proceed.

2. How does Simulated Annealing solve the problem of protein-ligand docking?
   - Ligand starts at initial state (random or user-defined).
   - The temperature of the system is reduced with time and the moves of the atoms are accepted depending on its energy compared to previous energy (with a probability proportional to the temperature!)
   - Repeat until reaching final solution.

The following is the algorithm in detail.

- *Ligand starts at a random (or user-specified) position/orientation/conformation ('state')*
- *Constant-temperature annealing cycle:*
  - *Ligand's state undergoes a random change.*
  - *Compare the energy of the new position with that of the last position; if it is:*
    - *lower, the move is 'accepted';*
    - *higher, the move is accepted if* $e(-\Delta E/kT) > 0$ *;*
    - *otherwise the current move is 'rejected'.*
  - *Cycle ends when we exceed either the number of accepted or rejected moves.*
- *Annealing temperature is reduced,* $0.85 < g < 1$
  - $T_i = g\, T_{i-1}$
- *Repeat.*
- *Stops at the maximum number of cycles.*

3. What are the advantages of simulated annealing?

One obvious advantage of the method is that it is more amenable to incorporate ligand flexibility into its modeling whereas shape complementarity techniques have to use some ingenious methods to incorporate flexibility in ligands. Another advantage is that the process is physically closer to what happens in reality, when the protein and ligand approach each other after molecular recognition.

4. Describe the force field that AutoDock 4 uses.

AutoDock 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.

5. Briefly describe the terms used in AutoDock 4 force field equation.

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

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

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:

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

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 exponential term with distance-weighting factor σ=3.5Å.

Don't worry about the details of the equation, just know what contributes to the force field used in AutoDock, and that the force field is semi-empirical.

6. Explain how geometric hashing is used for pattern recognition in bioinformatics.

In rigid protein-ligand docking, a program discretizes the different conformations of molecules providing pairs or trios of values for different points on the objects. In an off-line step, the objects are encoded by treating each pairs of points as a geometric basis. The remaining points can be represented in an invariant fashion with respect to this basis using two parameters. For each point, its quantized transformed coordinates are stored in the hash table as a key, and indices of the basis points as a value. Then a new pair of basis points is selected, and the process is repeated. In the on-line (recognition) step, randomly selected pairs of data points are considered as candidate bases. For each candidate basis, the remaining data points are encoded according to the basis and possible correspondences from the object are found in the previously constructed table. The candidate basis is accepted if a sufficiently large number of the data points index a consistent object basis.

7. What is DOCK? How are DOCK and AutoDock different?

   It is the most successful rigid protein-ligand docking technique, efficient enough to screen entire chemical database rapidly for lead compounds. The basic idea behind DOCK is to represent active site by set of spheres, and perform sphere matching. DOCK is used for rigid protein-ligand docking whereas AutoDock is used for flexible. Their search algorithms are also very different, one DOCK uses SPHGEN/MATCH, AutoDock mainly uses Genetic Algorithm.

8. Why do the results differ when multiple docking are done with the same input?

   AutoDock uses a random number generator to create new poses for the ligand during its search. The random number generator produces a sequence of random numbers based on two initial seeds. The new conformations for the search are created using this sequence of random numbers to set location, orientation and torsion values. The default values for these two seeds are 'pid' and 'time'. Process id and time vary between AutoDock calculations. Therefore, the sequence of random numbers is different between different AutoDock calculations. As a result, the 'search' is encountering a different set of random conformations, and the results differ.

9. When AutoDock 4 performs conformational clustering on the docking results, which atoms are used in the clustering?

   In AutoDock 4, conformational clustering is performed after all the dockings have finished if the keyword ``analysis” is given in the docking parameter file (DPF).

   By default, only the atoms in the moving ligand (defined by the ``move” keyword in the DPF) are used in the RMSD clustering calculations. There is a DPF keyword, ``rmsatoms” that can take the argument ``all”, that tells AutoDock 4 to include the atoms in the flexible residues in the receptor in the RMSD calculations for the clustering.

10. What can cause high Reference RMSD values in DLG (docking log file)?

    The "Reference RMSD" values that are printed in the "RMSD TABLE" in the DLG are computed from the coordinates of either the input ligand (PDBQ or PDBQT) file specified by the "move" command in the DPF or the ligand (PDBQ or PDBQT) file one specifies in the "rmsref" command in the DPF. If one does not specify the "rmsref" command, and the ligand input coordinates happen to be translated far from the receptor, it will result in high Reference RMSD values.

11. How many AutoGrid grid maps are needed for a protein-ligand docking simulation with AutoDock?

    For every atom type in the ligand on AutoGrid map is needed; plus an electrostatics map and a desolvation map. E.g., for ethanol, C_2H_5OH, you would need C, O and H maps plus an electrostatics ‘e’ map plus a desolvation ‘d’ map.

12. What factor contributes to a good result in AutoDock the most?

    In general, the more rotatable bonds in the ligand, the more difficult it will be to find good binding modes in repeated docking experiments, so the less rotatable bonds present, the better the quality of the results.

13. How big should the AutoGrid grid box be?

    The grid volume should be large enough to at least allow the ligand to rotate freely, even when the ligand is in its most fully- extended conformation.

14. Is it possible to identify potential binding sites of a ligand on a protein with AutoDock?

    Yes, if you do not know where the ligand binds, you can build a grid volume that is big enough to cover the entire surface of the protein, using a larger grid spacing than the default value of 0.375Å, and more grid points in each dimension. Then you can perform preliminary docking experiments with AutoDock to see if there are particular regions of the protein that are preferred by the ligand. This is sometimes referred to as “blind docking”. Then, in a second round of docking experiments, you can build smaller grids around these potential binding sites and dock in these smaller grids.

    If the protein is very large, then you can break it up into overlapping grids and dock into each of these grid sets, e.g. one covering the top half, one covering the lower half, and one covering the middle half.

15. What characteristic defines the ``best root” in AutoDock, and how is it chosen?

    The best root is the atom in the ligand with the smallest largest sub-tree. In the case of a tie, if either atom is in a cycle, it is picked to be root. If neither atom is in a cycle, the first found is picked. (If both are in a cycle, the first found is picked)

16. As we know, AutoDock can be used for ``blind docking”. Briefly explain how you can use AutoDock when the structure of the ligand and the protein are both known, but the location of the binding site is unknown.

    It will be necessary to set up the dockings to search the entire surface of the protein (or other macromolecule) of interest. This can be achieved using AutoGrid to create very large grid maps, with the maximum number of points in each dimension, and if necessary, creating sets of adjacent grid map volumes that cover the macromolecule. The third-party tool BDT can be used to set up such sets of grid maps.

17. When is AutoDock not suitable?

    When there are no 3D structures available, the modeled structure is of poor quality, there are too many variables (32 torsions, 2048 atoms, 22 atom types), or the target protein is too flexible.

18. Given two files, hsg1.pdb (a protein data base file for HIV-1 Protease) and ind.pdb (an inhibitor), using AutoDock 4 with ADT, dock the two molecules (explain how you would do it).

    I will go through the steps required to solve this question. Keep in mind that every proteins-ligand docking problem is different, and you will have to change AutoDock options for each problem accordingly.

    Reading input files:

    The first step to solving every problem is reading the macromolecule (protein) and ligand files and modifying them to the right format for docking.

    - For the macromolecule:

    File → Read Molecule → hsg1.pdb – You can change the way the molecule looks using the menu options and the options provided in the dash. For a better view, for example, you can click on the circle in front of the macromolecule's name and under "S&B" and then do Color → By Atom Type → All Geometrics → OK

    This would cause the following colors to appear:

    Carbons that are aliphatic (C) - white,
    Carbons that are aromatic (A) - green,
    Nitrogens (N) - blue,
    Oxygens (O) - red,
    Sulfurs (S) - yellow,
    Hydrogens (H) - cyan.

To manipulate hydrogens you should click on: Edit → Hydrogens → Add
Choose to add **All Hydrogens** using Method **noBondOrder** with **yes** to renumbering. Click OK to add the polar hydrogens. 1612 hydrogen atoms are added to hsg1. (These modifications may be different for another problem) Hide hsg1 before going on by clicking on the gray showMolecules rectangle for hsg1 in the Dashboard.

- For the ligand:

Hide hsg1 in and using the menu click on Ligand → Input → Open → ind.pdb
After the ligand is loaded in the viewer, ADT initializes it. This process involves a number of steps. Then, ADT reports its findings.

Ligand → Torsion Tree → Detect Root detects the best root and marks it with a green sphere. This best root is the atom in the ligand with the smallest largest subtree. In the case of a tie, if either atom is in a cycle, it is picked to be root. If neither atom is in a cycle, the first found is picked. (If both are in a cycle, the first found is picked). As you might imagine, this can be a slow process for large ligands. However, at this point in our example, the root portion includes only the best root atom, atom C11, because all its bonds to other atoms are

rotatable. (This is an optional step)
Ligand → Torsion Tree → Choose Torsions… shows Torsion Count widget. The widget displays the number of currently active bonds. You can set which of these are to be rotatable. For this exercise, there is no need to change the setting, but you may have to do so for another problem.
Ligand → Torsion Tree → Set Number of Torsions allows us to set the total number of active bonds while specifying whether you want active bonds which move the fewest atoms or those which move the most. For our exercise, enter 6 in the number of active torsions field and that move the fewest atoms active.

Save the file by clicking on Ligand → Output → Save as PDBQT.

Preparing the flexible residue file:
Redisplay hsg1 using its show Molecules rectangle in the Dashboard. Choose hsg1 as the macromolecule to have flexible residues: Flexible Residues → Input → Choose Macromolecule...

Click on hsg1 in the widget that opens and on Select Molecule . Click Yes when asked if you want to merge the non-polar hydrogens. Click on OK in the formatting summary widget. Now, select the residues to be flexible by clicking on Select → Select From String Click on Clear Form to empty the entries. Type ARG8 in the Residue entry and click on Add, and then click on Dismiss to close the Select From String widget. Check that 2 Residues appear in the Selected: entry below the 3D Viewer.

Now, we have to define the rotatable bonds in the selected residues.
Flexible Residues → Choose Torsions in Currently → Selected Residues...
This hides all the non-selected residues in the macromolecule. The side chains of the selected residues are shown with currently rotatable bonds colored green, unrotatable bonds colored red and non-rotatable bonds colored magenta. The total number of rotatable bonds is listed in the Torsion Count widget. Clicking on a rotatable bond makes it non-rotatable. Clicking on a non- rotatable bond makes it rotatable.

Click on the rotatable bond between CA and CB in each residue to inactivate it. This leaves a total of 6 rotatable bonds in the two flexible ARG8 residues. Click on Close . Clear the selection by clicking on the pencil eraser icon.

We must save the macromolecule in two files: one containing the formatted, flexible ARG8 residues and the other all the rest of the residues in the macromolecule. To do so, click on Flexible Residues → Output → Save Flexible PDBQT... and type hsg1_flex.pdbqt in the AutoFlex File: browser and click Save.

Then click on Flexible Residues → Output → Save Rigid PDBQT... and type hsg1_rigid.pdbqt in the AutoFlex Non-Flexible Residue Output File: browser and click Save. Remove this version of hsg1 by clicking on Edit → Delete → Delete Molecule

Preparing the Macromolecule:

Using our previously edited hsg1.pdb file, clicking on Grid → Macromolecule → Choose → hsg1_rigid.pdbqt. Click OK on the WARNING dialog box.

Preparing the grid parameter file:

Grid → Set Map Types → Choose Ligand... , select 'ind' and click on the Select Ligand button. Or, use Grid → Set Map Types → Open Ligand... .

Grid → Grid Box opens a widget which displays the Current Total Grid Points per map. This tells you how big each grid map will be: (n_x + 1) x (n_y+ 1) x (n_z+ 1), where n_x is the number of grid points in the x-dimension, *etc.*

Adjust the number of points in each dimension to 60, 60, 66. Notice that each map will have 249,307 points. Type in 2.5, 6.5 and -7.5 in the x center, y center and z center entries. This will center the grid box on the active site of the HIV-1 protease, **hsg1**. Close this widget by clicking File → Close saving current.

Grid → Output → Save GPF will allow you to save your gpf file ☺

Click on Grid → Edit GPF to show you the details of the file you just wrote.

Running AutoGrid:

By clicking on Run → Run AutoGrid, AutoGrid Opens the Run AutoGrid widget. Click on Launch to continue.

Preparing the docking parameter file:

Docking → Macromolecule → Set Rigid Filename allows you to select the file you wrote previously hsg1_rigid.pdbqt.

Docking → Ligand → Choose allows you to choose your ligand. Choosing our formatted ligand opens a panel that tells you the name of the current ligand, its atom types, its center, its number of active torsions and its number of torsional degrees of freedom. You can set a specific initial position of the ligand and initial relative dihedral offsets and values for its active torsions. For our exercise we will use the defaults. Click Close to close this widget.

Docking → Macromolecule → Set Flexible Residues

Filename... lets you Choose hsg1_flex.pdbqt.

Docking → Search Parameters → Genetic Algorithm opens a widget that lets you change the genetic algorithm specific parameters. This lets you change the genetic algorithm specific parameters. It is a good idea to do a trial run with fewer energy evaluations.
For our exercise, we will use the short setting, i.e. 250,000 energy evaluations. This is listed as "Maximum Number of evals". Click Accept to continue.

Docking → Docking Parameters… allows you to choose which random number generator to use, the random number generator seeds, the energy outside the grid, the maximum allowable initial energy, the maximum number of retries, the step size parameters, output format specification and whether or not to do a cluster analysis of the results. For today, use the defaults and just click Close.

Docking → Output → Lamarckian GA (4.2) … allows you file that will contain docking parameters and instructions for a Lamarckian Genetic Algorithm (LGA) docking.

Starting AutoDock:

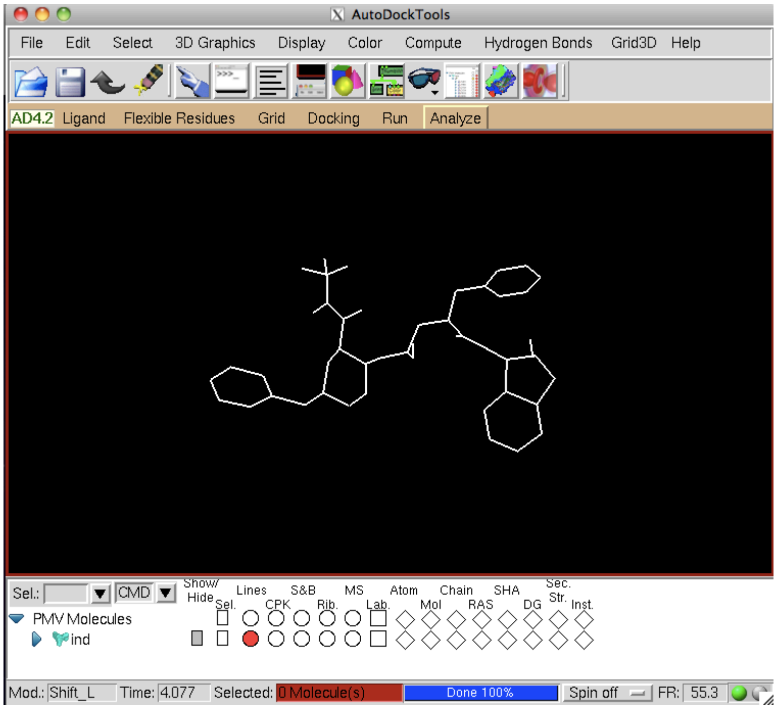
Run → Run AutoDock shows you a window very similar to the one we saw with AutoGrid. Click on Launch to perform the task.

Docking is done! Now it's time to analyze some results ☺

19. Analyze the results of the docking using ADT analysis Tools.

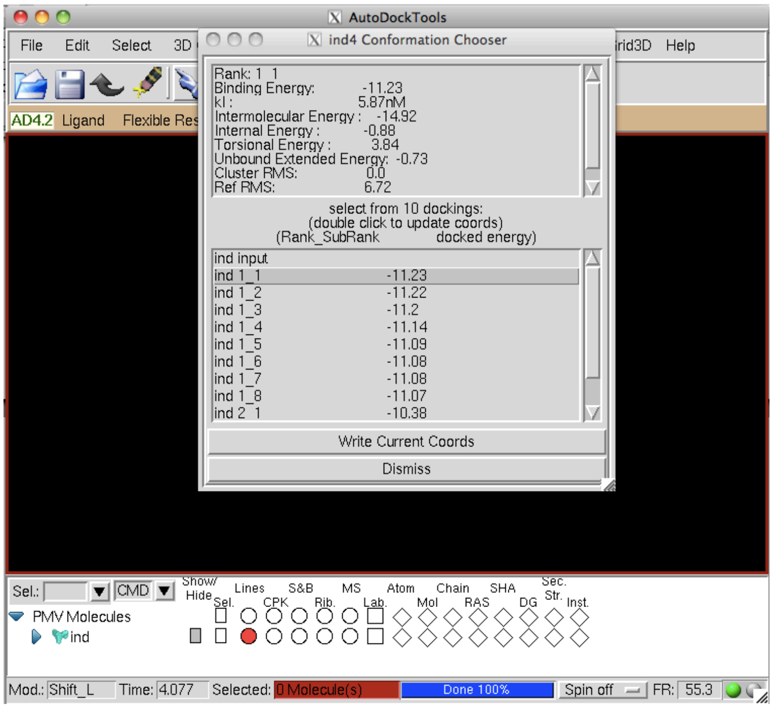
    If you have not done the previous exercise, do not worry! ☺ I have included the results files for your convenience. You can just use those and analyze them.

    First you need to be able to read the docking log files. The following steps helps you to Choose the AutoDock log file you would like to Analyze. Clicking on Analyze -> Dockings -> Open opens a file browser that lets us choose a file with the extension .dlg. Choose ind.dlg. Reading a docking log creates a Docking instance in the viewer. A Conformation instance is created for each docked result found 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 atoms' coordinates can be computed or the coordinates themselves. Conformations also have energies: docked energy, binding energy, and possibly per atom electrostatic and vdw energies. In this case the result is the following picture:


AutoDockTools
File Edit Select 3D Graphics Display Color Compute Hydrogen Bonds Grid3D Help
AD4.2 Ligand Flexible Residues Grid Docking Run Analyze
Sel.: CMD
Show/ Hide Sel. Lines CPK S&B Rib. MS Lab. Atom Mol Chain RAS SHA DG Sec. Str. Inst.
PMV Molecules
ind
Mod.: Shift_L Time: 4.077 Selected: 0 Molecule(s) Done 100% Spin off FR: 55.3

Analyze → Conformations → Load…

This opens ind Conformation Chooser which gives you a concise view of the energies and clusters of the docked results. The following picture is a snapshot of this.


AutoDockTools
File Edit Select 3D Grid3D Help
AD4.2 Ligand Flexible Res
ind4 Conformation Chooser
Sel.: CMD
Show/ Hide Sel. Lines CPK S&B Rib. MS Lab. Atom Mol Chain RAS SHA DG Sec. Str. Inst.
PMV Molecules
ind
Mod.: Shift_L Time: 4.077 Selected: 0 Molecule(s) Done 100% Spin off FR: 55.3

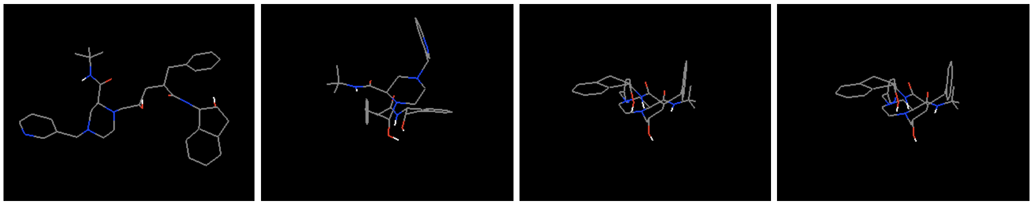
```
Rank: 1  1
Binding Energy:          -11.23
kI :                      5.87nM
Intermolecular Energy :   -14.92
Internal Energy :         -0.88
Torsional Energy :         3.84
Unbound Extended Energy:  -0.73
Cluster RMS:              0.0
Ref RMS:                  6.72

            select from 10 dockings:
          (double click to update coords)
      (Rank_SubRank             docked energy)

ind input
ind 1_1                    -11.23
ind 1_2                    -11.22
ind 1_3                    -11.2
ind 1_4                    -11.14
ind 1_5                    -11.09
ind 1_6                    -11.08
ind 1_7                    -11.08
ind 1_8                    -11.07
ind 2  1                   -10.38
```

Write Current Coords

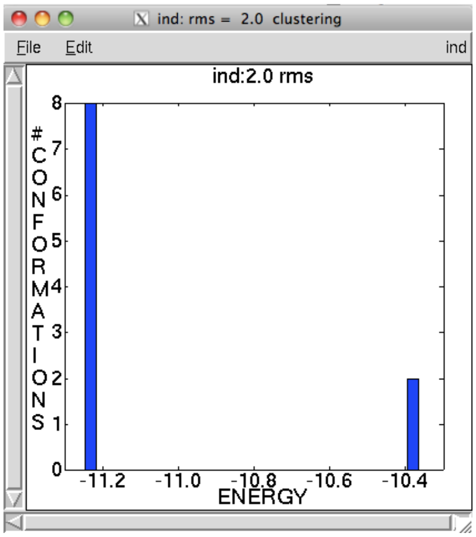
Dismiss

The lower panel lists the docked conformations for the ligand grouped according to the clustering performed at the end of the AutoDock calculation. The input conformation is the first entry in this list. And the best result is 1_1: lowest energy cluster_best individual in cluster. **Docked Energy** is the sum of the intermolecular and internal energy components and for the best conformations is -11.23. **Cluster RMS** is the root mean square difference rms between this individual and the seed for the cluster. 1_1 is the seed for the first cluster so its Cluster RMS is 0.0. **Ref RMS** is the rms between the specified reference structure. **freeEnergy** is the sum of the intermolecular energy plus the torsion entropy penalty which is a constant times the number of rotatable bonds in the ligand, **kI** calculated from the Docked Energy. 10 clusters were formed with our docking results. The range in energy between the 'best' docking and the seed of the last cluster is 0.85.

Next step in the analysis is visualizing the docked conformations.
Analyze -> Conformations -> Play… opens a Conformation player (CP) that we can use to examine the docked conformations of ind.pdbqt. The CP has a current list of conformations and a current ID list. Using the arrow, I went through the 11 conformations (10 + original). The following images are the original and the best 3 conformations suggested by AutoDock (from left to right)



Now, we will analyze the clustering conformations. Analyze -> Clustering -> Show… Opens an instance of a Python object, an interactive histogram chart labelled 'ind_1:rms = 2.0 clustering', following image shows this outcome.



The bars of this chart represent the clusters computed at the specified rmsd. The bars are sorted by energy of the lowest-energy conformation in that cluster and start off colored blue.
For example, the lowest energy conformation in the second bar is 2_1. The height of the bar represents how many conformations are in that cluster.

This concludes our analysis! ☺