Vina Docking Tutorial with Webina

Dr. S. Eagon
California Polytechnic State University

Note: This is a tutorial on how to use the Vina program to perform structure-based docking, also called *in silico* docking and/or virtual screening. I will use the term "docking" moving forward to save time. Structure-based docking means that you have a protein target that you hold rigid, then use a program to let a single molecule "wiggle" into the "best" conformation, as determined by an energy scoring function. This is different from ligand-based screening, which means a program tries to add/delete/change groups on a given molecule to find a derivative that best fits a rigid binding pocket. This is also different from molecular dynamics (MD), which allows a molecule AND a protein to move about in a simulated solvent environment.

Software you will need

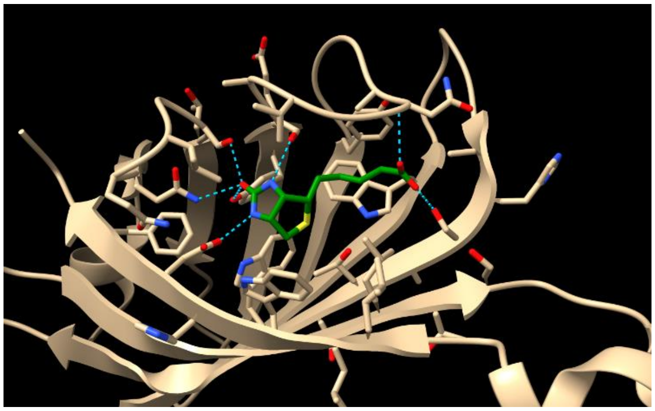
Before you begin, you need to install the following programs on your computer:

1. ChimeraX
2. Internet Browser
3. Text editor (recommend Notepad++ for Windows and BBEdit for Mac)
   - Mac Users -> Do NOT use the TextEdit app! This will save it as a .rtf file, and not a .txt file. This will cause your docking simulations to fail.

Caution: Anytime you download something, you take full responsibility for what happens to your computer. *Caveat emptor*

Goals of this Tutorial

This is a step-by-step guide on how to dock a small molecule to a protein target, using biotin binding to streptavidin as an example with the crystal structure 1stp. The goal is that you can build the biotin molecule as a 3D model, dock it to streptavidin, then show that the result matches the known binding orientation. Once you are able to do this, then you can select any protein target and build any molecule(s) you want and attempt to dock them. It's always recommended you follow this entire tutorial before trying a different molecule/protein target on your own.



Step 1: Create a folder on the desktop and add the initial files

Make a folder on the desktop called **DockingJob**. You can call this folder something else, but avoid using spaces, as this could cause problems later. If you want to add a space in the name, use the underscore and name it something like **Docking_Job**. For now, I'd recommend you just call it **DockingJob.**

All the initial files you will need can be found on my research webpage on the "Docking in the Classroom" page, looking for the "Web Docking with Webina" section. A direct link is here:

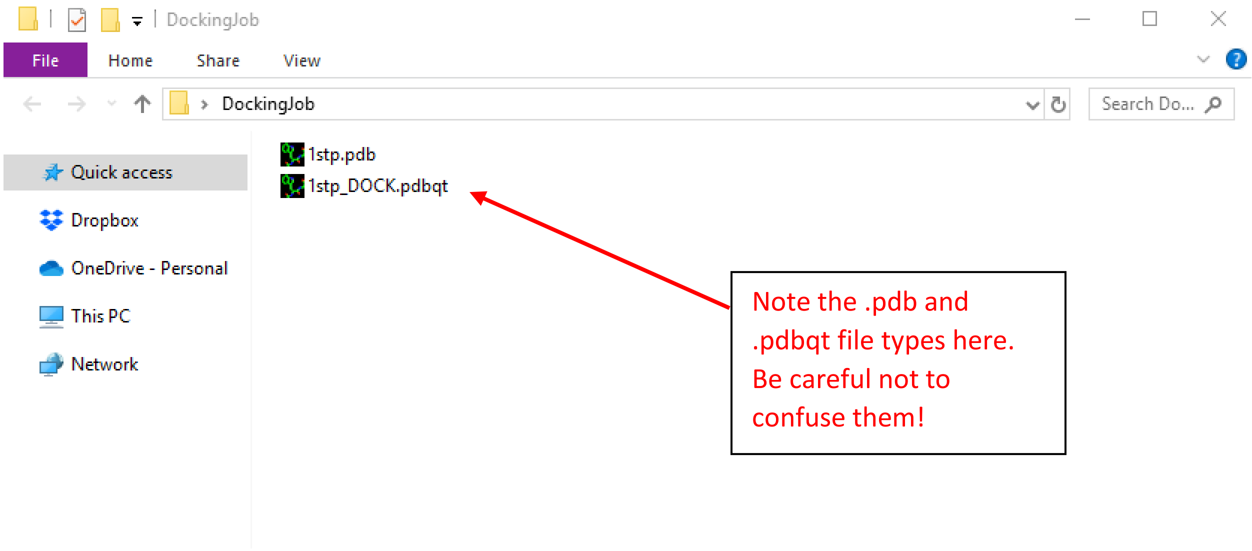
https://eagonlab.github.io/docking_in_the_classroom.html

From the page linked above, put the following files in your **DockingJob** folder:

1. The **1stp.pdb** file (can also be downloaded from PDB)
2. The **1stp_DOCK.pdbqt** file

Note: There is a more advanced tutorial on how to run Vina by line command on the same web page. If you have experience in running line commands and/or working in a Unix environment, you are welcome to try this more advanced method. It isn't worth any more points and is entirely optional. However, it does give you the ability to run large numbers of ligands quickly by using a custom script that I wrote. For most people, I recommend you stick with this Webina version, since it's much easier to use and requires less software.

- You should end up with a folder something like this:



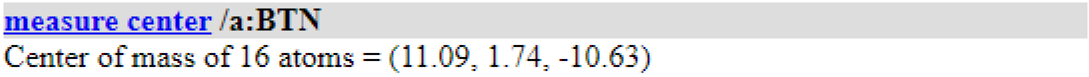
Note the .pdb and .pdbqt file types here. Be careful not to confuse them!

Note: It can be helpful to show the file extensions (the .pdb and .pdbqt). If you do not see these, then it's *highly recommended* that modify your settings so that you can. We will be interconverting several files with similar names, and the file types will be very helpful to distinguish them. Both Mac and PC have the ability to display these file types in the names—just search online for instructions.

Step 2: Find the coordinates of the ligand using ChimeraX

Before we dock any molecules, we need to determine the 3D coordinates of the ligand in the crystal. We will use these coordinates when we tell the program where to try to get our molecules to bind.

- Open the **1stp.pdb** file in ChimeraX
- Find the center of the biotin ligand using the following command:
  - $ measure center /a:BTN
    - You should get an output in the log pane that looks like this:



```
measure center /a:BTN
Center of mass of 16 atoms = (11.09, 1.74, -10.63)
```

This result tells us the coordinates of the center of the BTN (biotin) ligand in x,y,z form. In other words, the coordinates of the center of BTN is:

x = 11.09

y = 1.74

z = -10.63

You need to save these numbers in a file somewhere or write them down. We'll need them when we're setting up the parameters for the docking program.

Note: If you're wondering where these numbers come from, they arise from the PDB file. If you open a PDB file with a text editor and scroll down past the REMARKS lines, you'll see long lists of coordinates like this:

```
ATOM     1  N   ALA A  13      22.637   5.768  11.762  1.00 44.60          N
ATOM     2  CA  ALA A  13      23.655   4.852  11.146  1.00 44.15          C
ATOM     3  C   ALA A  13      24.276   5.552   9.942  1.00 43.61          C
ATOM     4  O   ALA A  13      23.942   5.192   8.794  1.00 43.25          O
ATOM     5  CB  ALA A  13      22.855   3.616  10.672  1.00 44.78          C
ATOM     6  N   GLU A  14      25.105   6.547  10.225  1.00 43.22          N
…(etc)
```

A PDB file is just a collection of points (ATOM 1, ATOM 2, etc.) with names (ALA A 13), coordinates (the first 3 numbers after the name) and the type of atom (at the end). For the example above, ATOM 1 is a Nitrogen atom in residue 13, an alanine, on chain A, with the coordinates x = 22,637, y = 5.768, and z = 11.762. The coordinate grid is centered on the crystal unit center, which is determined by the individual that solved the crystal structure.

At the bottom of the PDB file, you'll see a series of CONECT lines that tell ChimeraX how to stitch these atoms and residues together to form the peptide chain (yes, the word CONECT is spelled incorrectly. Blame the original programmer, who was probably just being lazy)

Step 3: Preparing the Ligands folder

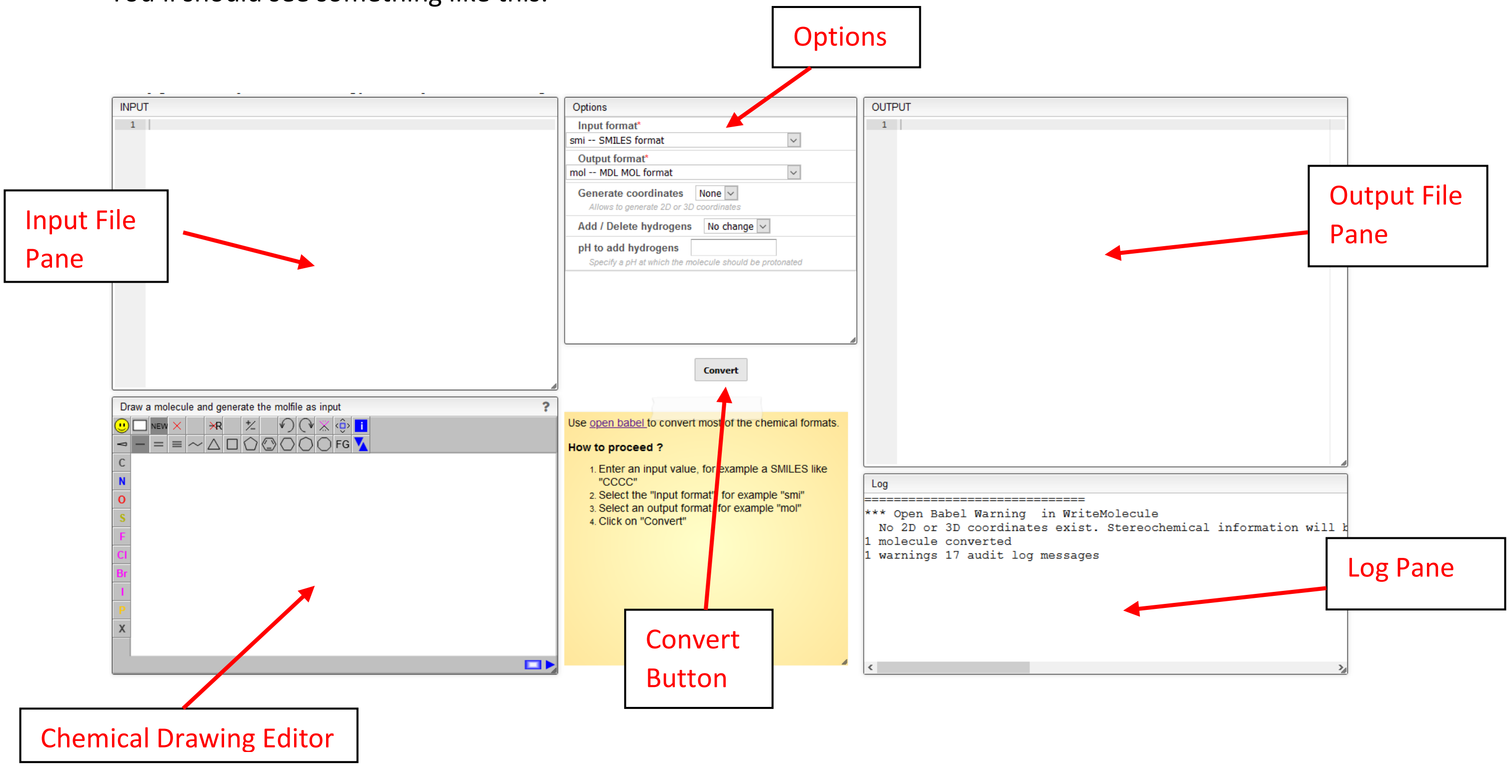
Next, we need to make a folder of ligands that we want docked to our protein. For this tutorial, we are going to try to dock biotin into streptavidin (the 1stp crystal structure). Since we already know that biotin binds streptavidin, this is a good way to make sure our docking simulations are working.

> **Important Note:**   Whenever you do a docking simulation, you should always use a positive control like this. In other words, be sure to dock a molecule you know binds your protein to make sure that everything is working. There's an old saying in computational chemistry that goes "garbage in, garbage out." This means the computer will always spit *something* out, but you need to have a known starting point, otherwise you might just end up with a bunch of garbage and not be able to tell the difference. This has happened to a lot of publications over the years.

- Start by creating a new folder called **Ligands** in the **DockingJobs** folder
- Now we need to create a file for biotin that the docking program can use and place it in the **Ligands** folder (biotin is the molecule that binds the streptavidin protein). Our docking program, Vina, looks at molecules that are in PDBQT format. So, we need to make a PDBQT file for biotin. To do this, we are going to use an online server (the OpenBabel server) that you can find at this website:

https://www.cheminfo.org/Chemistry/Cheminformatics/FormatConverter/index.html

You'll should see something like this:



Input format*: smi -- SMILES format

Output format*: mol -- MDL MOL format

Generate coordinates: None

Allows to generate 2D or 3D coordinates

Add / Delete hydrogens: No change

pH to add hydrogens:

Specify a pH at which the molecule should be protonated

Use open babel to convert most of the chemical formats.

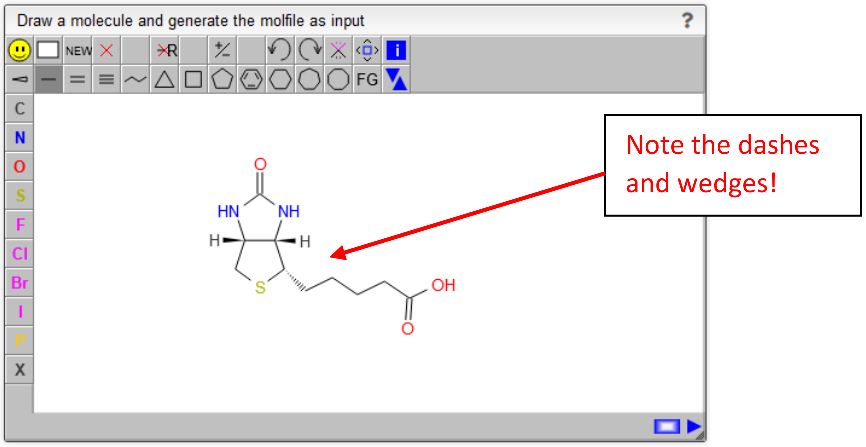
How to proceed ?

1. Enter an input value, for example a SMILES like "CCCC"
2. Select the "Input format", for example "smi"
3. Select an output format, for example "mol"
4. Click on "Convert"

```
==============================
*** Open Babel Warning  in WriteMolecule
  No 2D or 3D coordinates exist. Stereochemical information will b
1 molecule converted
1 warnings 17 audit log messages
```

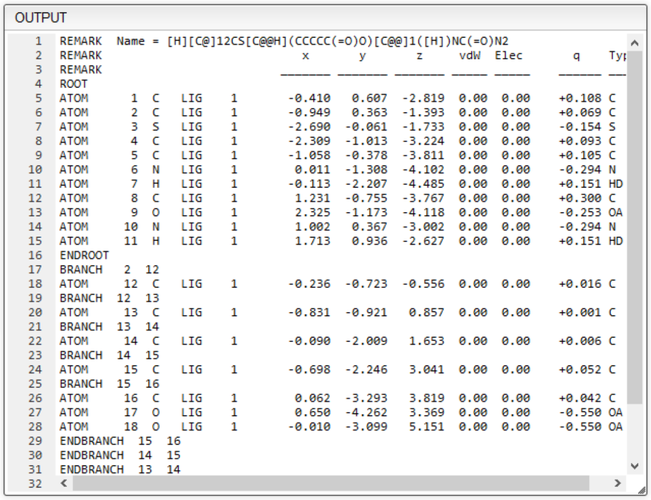
This webpage will allow us to turn molecular structures into PDBQT files for docking.

- In the chemical drawing editor window, draw the structure of biotin. Be sure to have the stereochemistry correct!
    - Notice as you draw the structure, you'll see the coordinates pop up in the input file pane
    - Here's how it will look in the editor pane:



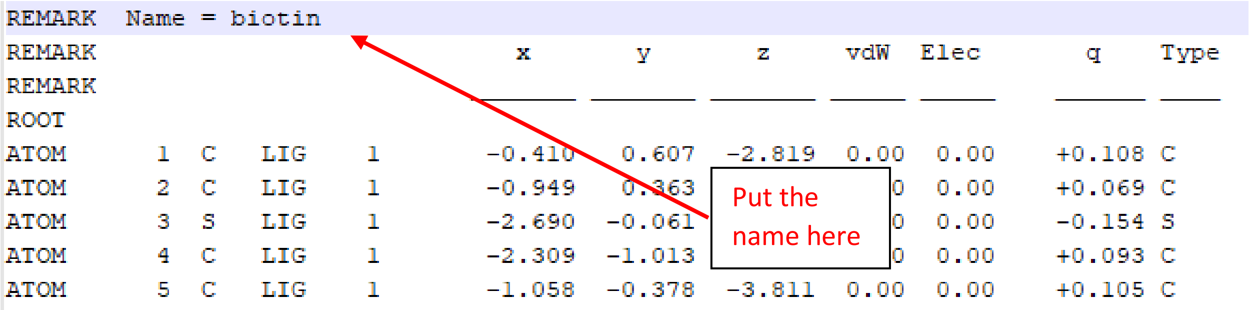
Note the dashes and wedges!

    - If you haven't used the JSME editor before, here are a few tips:
        - It's best to draw the skeleton first then hover your cursor over the atoms to change them with the keyboard. For example, if you hover over an atom and hit the "N" on your keyboard, it will change it to a nitrogen atom.
        - You can get the dashed line by first drawing the wedge with the wedge tool (above the C button) then clicking on the bond again.
- In the "Options" pane, do the following:
    - Change the "Input format" to "mol -- MDL MOL format" (the top option)
    - Change the "Output format" drop-down menu to select "pdbqt -- AutoDock PDBQT format"
    - Select "3D" from the "Generate coordinates" box
    - Leave the "Add / delete hydrogens" selection at "No change"
    - add the number 7.4 to the "pH to add hydrogens" box (this is physiological pH)
        - This ensures that all functional groups are at the appropriate protonation state
- Lastly, hit the "Convert" button to get the following in the "Output Pane"



```
REMARK  Name = [H][C@]12CS[C@@H](CCCCC(=O)O)[C@@]1([H])NC(=O)N2
REMARK                              x       y       z     vdW  Elec       q    Typ
REMARK                           _______ _______ _______ _____ _____    ______ ___
ROOT
ATOM      1  C   LIG     1      -0.410   0.607  -2.819  0.00  0.00    +0.108 C
ATOM      2  C   LIG     1      -0.949   0.363  -1.393  0.00  0.00    +0.069 C
ATOM      3  S   LIG     1      -2.690  -0.061  -1.733  0.00  0.00    -0.154 S
ATOM      4  C   LIG     1      -2.309  -1.013  -3.224  0.00  0.00    +0.093 C
ATOM      5  C   LIG     1      -1.058  -0.378  -3.811  0.00  0.00    +0.105 C
ATOM      6  N   LIG     1       0.011  -1.308  -4.102  0.00  0.00    -0.294 N
ATOM      7  H   LIG     1      -0.113  -2.207  -4.485  0.00  0.00    +0.151 HD
ATOM      8  C   LIG     1       1.231  -0.755  -3.767  0.00  0.00    +0.300 C
ATOM      9  O   LIG     1       2.325  -1.173  -4.118  0.00  0.00    -0.253 OA
ATOM     10  N   LIG     1       1.002   0.367  -3.002  0.00  0.00    -0.294 N
ATOM     11  H   LIG     1       1.713   0.936  -2.627  0.00  0.00    +0.151 HD
ENDROOT
BRANCH   2  12
ATOM     12  C   LIG     1      -0.236  -0.723  -0.556  0.00  0.00    +0.016 C
BRANCH  12  13
ATOM     13  C   LIG     1      -0.831  -0.921   0.857  0.00  0.00    +0.001 C
BRANCH  13  14
ATOM     14  C   LIG     1      -0.090  -2.009   1.653  0.00  0.00    +0.006 C
BRANCH  14  15
ATOM     15  C   LIG     1      -0.698  -2.246   3.041  0.00  0.00    +0.052 C
BRANCH  15  16
ATOM     16  C   LIG     1       0.062  -3.293   3.819  0.00  0.00    +0.042 C
ATOM     17  O   LIG     1       0.650  -4.262   3.369  0.00  0.00    -0.550 OA
ATOM     18  O   LIG     1      -0.010  -3.099   5.151  0.00  0.00    -0.550 OA
ENDBRANCH  15  16
ENDBRANCH  14  15
ENDBRANCH  13  14
```

- In your **Ligands** folder (which is in the **DockingJob** folder), create a text file and call it **ligand_biotin.txt**
  - PC Users: You can create a text file with Notepad or similar program
  - Mac Users: You need to use the BBEdit app to make a text file
    - Mac Users -> Do NOT use the TextEdit app! This will save it as a .rtf file, and not a .txt file. This can cause problems later as we convert files.
- Copy and paste the contents of the "Output pane" into the **ligand_biotin.txt** file.
- On the first line of your text file, replace the long SMILES string with just the name biotin so it looks like this:



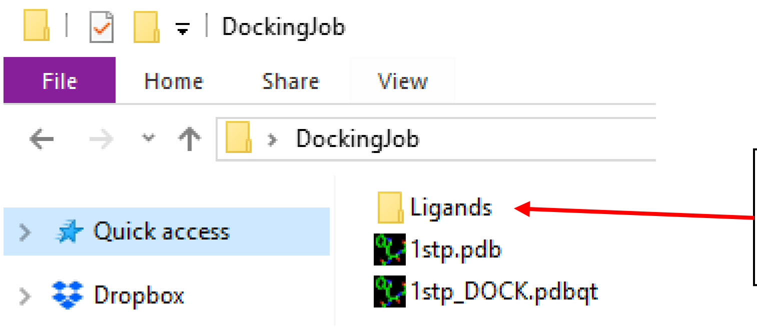
```
REMARK  Name = biotin
REMARK                          x       y       z     vdW   Elec         q    Type
REMARK                       _______ _______ _______ _____ _____      ______ ____
ROOT
ATOM      1  C   LIG    1    -0.410   0.607  -2.819  0.00  0.00      +0.108 C
ATOM      2  C   LIG    1    -0.949   0.363         0  0.00      +0.069 C
ATOM      3  S   LIG    1    -2.690  -0.061         0  0.00      -0.154 S
ATOM      4  C   LIG    1    -2.309  -1.013         0  0.00      +0.093 C
ATOM      5  C   LIG    1    -1.058  -0.378  -3.811  0.00  0.00      +0.105 C
```

Put the name here

Note: Notice that the PDBQT file looks a lot like a PDB file in some ways! PDB files came first, with the "PDB" standing for "Protein DataBase" file. To run a docking simulation, the program must also know the relative charge at each atom (Q) and the atom type (T). So, PDB + charge (Q) + type (T) gives you a PDBQT file!

- Lastly, change the file extension of the text file to pdbqt. You can do this several ways:
  - Just change the file name by modifying "**ligand_biotin.txt**" to "**ligand_biotin.pdbqt**"
    - Ignore any warnings the computer gives about doing this.
    - Another way to change the file extension is through using the "Save As" command and changing the file extension
    - If you can't see the file extensions (the .txt, .doc, .pdb stuff), you'll need to change the settings on how you view your folder items to see them.

- You should now have a folder that looks something like this:



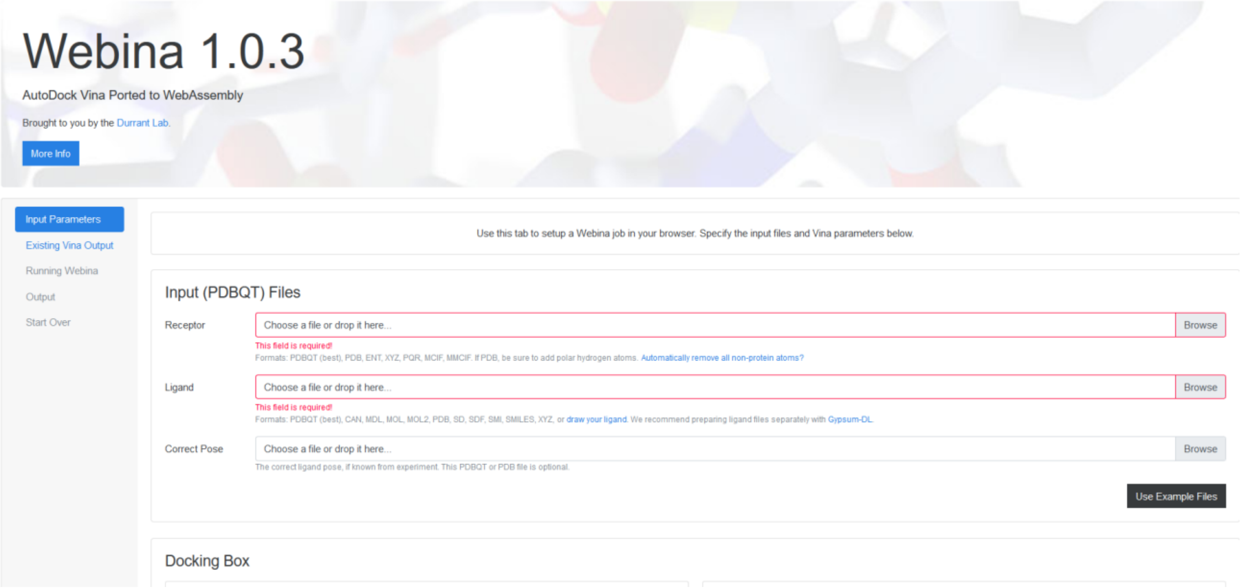
ligand_biotin.pdbqt file is in the Ligands folder

Step 4: Using Webina to Dock your Molecule(s)

Webina a is browser interface that runs Vina and is provided by the Durrant Lab at the University of Pittsburgh. You can find it at the following web address:

https://durrantlab.pitt.edu/webina/

You'll see something that looks like this:



Webina 1.0.3

AutoDock Vina Ported to WebAssembly

Brought to you by the Durrant Lab.

Use this tab to setup a Webina job in your browser. Specify the input files and Vina parameters below.

Input (PDBQT) Files

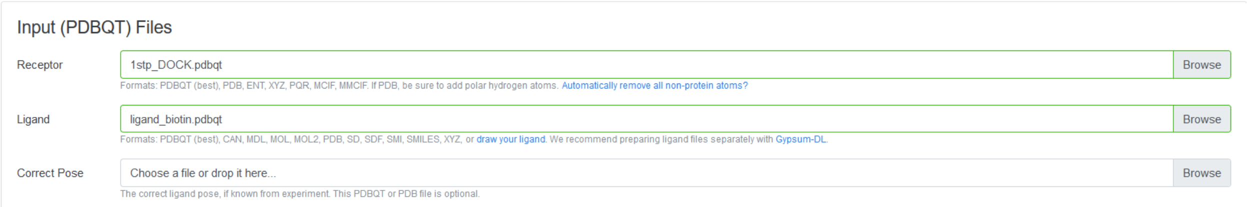
Receptor: Choose a file or drop it here... [Browse]
This field is required!
Formats: PDBQT (best), PDB, ENT, XYZ, PQR, MCIF, MMCIF. If PDB, be sure to add polar hydrogen atoms. Automatically remove all non-protein atoms?

Ligand: Choose a file or drop it here... [Browse]
This field is required!
Formats: PDBQT (best), CAN, MDL, MOL, MOL2, PDB, SD, SDF, SMI, SMILES, XYZ, or draw your ligand. We recommend preparing ligand files separately with Gypsum-DL.

Correct Pose: Choose a file or drop it here... [Browse]
The correct ligand pose, if known from experiment. This PDBQT or PDB file is optional.

Docking Box

- Start by uploading the receptor PDBQT file (**1stp_DOCK.pdbqt**) and the ligand file for the molecule we want to dock (**ligand_biotin.pdbqt**).
  - Leave the "Correct Pose" blank
- If you uploaded the correct files, the red fields should turn green like this:



Input (PDBQT) Files

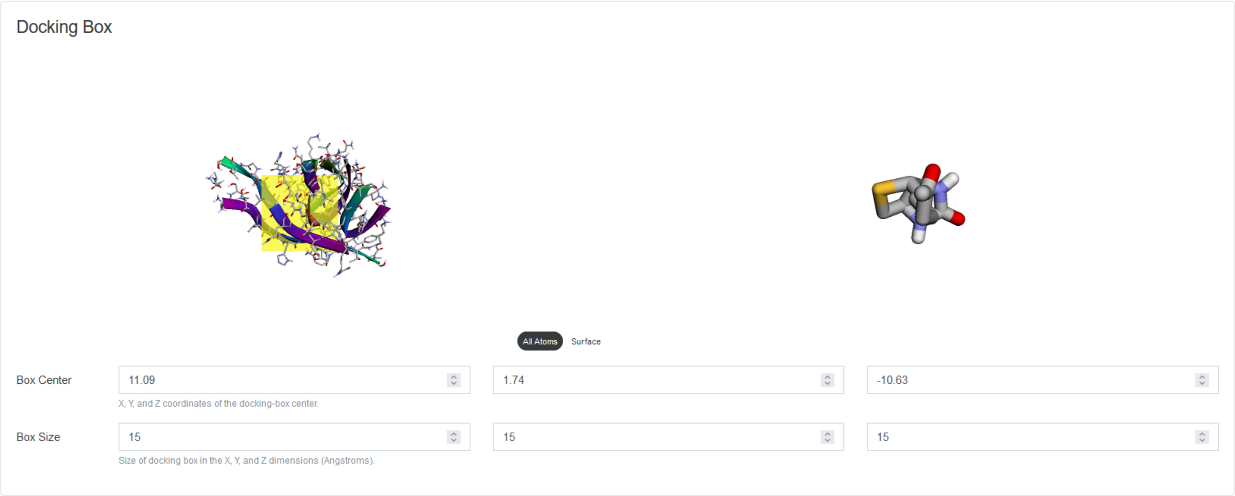
Receptor: 1stp_DOCK.pdbqt [Browse]
Formats: PDBQT (best), PDB, ENT, XYZ, PQR, MCIF, MMCIF. If PDB, be sure to add polar hydrogen atoms. Automatically remove all non-protein atoms?

Ligand: ligand_biotin.pdbqt [Browse]
Formats: PDBQT (best), CAN, MDL, MOL, MOL2, PDB, SD, SDF, SMI, SMILES, XYZ, or draw your ligand. We recommend preparing ligand files separately with Gypsum-DL.

Correct Pose: Choose a file or drop it here... [Browse]
The correct ligand pose, if known from experiment. This PDBQT or PDB file is optional.

- In the "Docking Box" section, you need to enter the x,y,z coordinates we found in **Step 2** earlier.
  - Note -> if you run a different crystal (other than 1stp), then you need different numbers here (like for an individual assignment).
- For the "Box Size," just enter 15 in all three boxes
  - The "size" tells Vina how big our cubic search area is in Angstroms. We're telling it to search a 15 x 15 x 15 Å cube, which is large enough to accommodate drug-like molecules (anything less than 500 g/mol or so)
  - You can use 15 for most docking jobs as a default value

- Your “Docking Box” section should now look something like this:



Docking Box

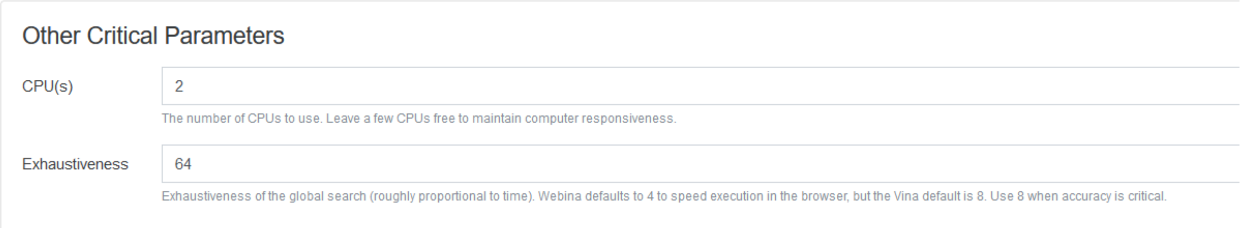
Box Center: 11.09 | 1.74 | -10.63

X, Y, and Z coordinates of the docking-box center.

Box Size: 15 | 15 | 15

Size of docking box in the X, Y, and Z dimensions (Angstroms).

- In the “Other Critical Parameters” section, set the CPU(s) to 2 and the Exhaustiveness to 64 like this:



Other Critical Parameters

CPU(s): 2

The number of CPUs to use. Leave a few CPUs free to maintain computer responsiveness.

Exhaustiveness: 64

Exhaustiveness of the global search (roughly proportional to time). Webina defaults to 4 to speed execution in the browser, but the Vina default is 8. Use 8 when accuracy is critical.

  - The “cpu” line tells the program how many logical cores from the CPU can be utilized to run the simulation.
  - The “num_modes” means the max number of times Vina will try to dock something
  - The “exhaustiveness” tells Vina how intensive to make the computations.

Note: The amount of time Vina takes depends greatly upon the “cpu” setting. The more CPU power you devote to the program, the faster it will go. However, if you utilize 100% of your CPUs to the docking job, this can cause all other programs you may be running (and even your OS) to crash.

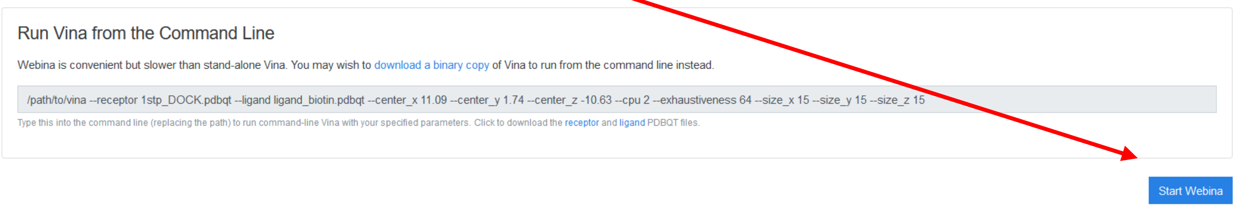
A setting of “cpu = 2” is pretty low, but I’ve picked it so that any computer should be able to run it without too many problems. A good rule of thumb is to not devote more than 50% CPU power to any single job. There is a point, also, where too many cpu threads actually slow down the program. The sweet spot is usually around 4-8 cpu threads.

Remember that these numbers correspond to logical core threads, not physical cores. So a quad core with hyperthreading (2 logical cores per physical CPU) will give you a total of 8 logical cores that can thread. If you want to know more about hyperthreading and logical cores vs. physical cores, consider a minor in Computer Science or Computer Engineering.

If you’re not sure about any of this, just leave the CPU setting to 2.

- Leave the settings in the “Advanced Parameters” alone

Once you have everything above set and ready to go, just hit the blue “Start Webina” button at the lower right of the web page to start the job!



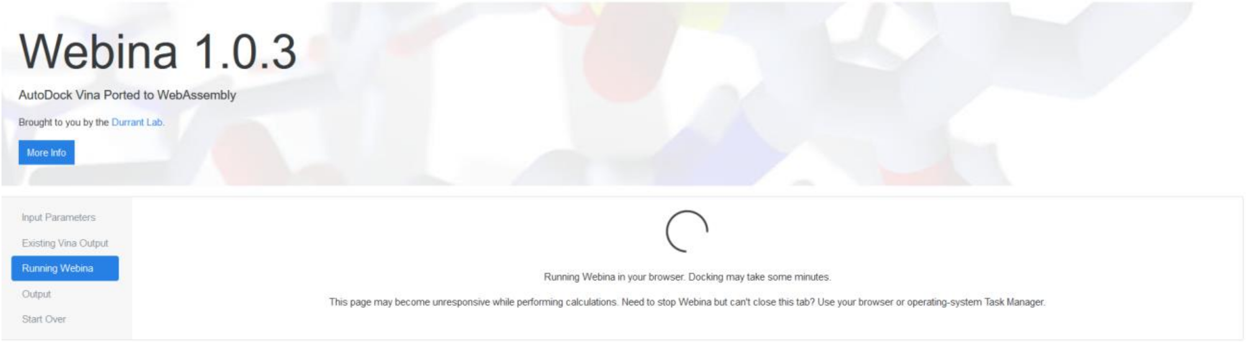
Run Vina from the Command Line

Webina is convenient but slower than stand-alone Vina. You may wish to download a binary copy of Vina to run from the command line instead.

```
/path/to/vina --receptor 1stp_DOCK.pdbqt --ligand ligand_biotin.pdbqt --center_x 11.09 --center_y 1.74 --center_z -10.63 --cpu 2 --exhaustiveness 64 --size_x 15 --size_y 15 --size_z 15
```

Type this into the command line (replacing the path) to run command-line Vina with your specified parameters. Click to download the receptor and ligand PDBQT files.

- As Webina docks your molecule, you should see a web page indicating that the simulation is running:



Webina 1.0.3

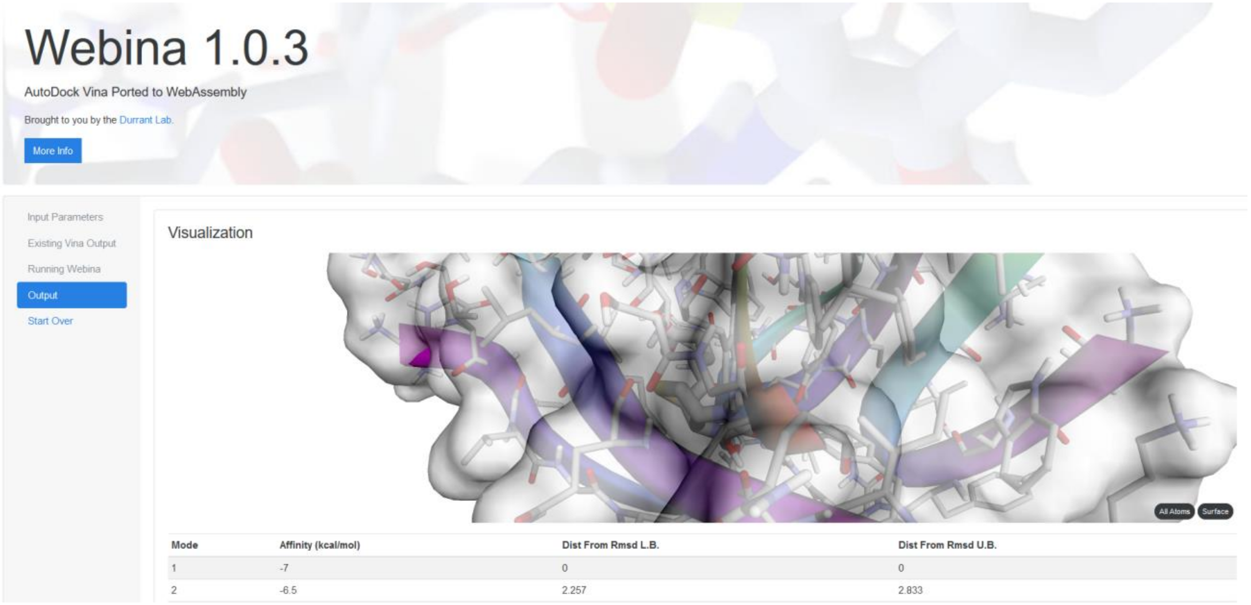
AutoDock Vina Ported to WebAssembly

Brought to you by the Durrant Lab.

Running Webina in your browser. Docking may take some minutes.

This page may become unresponsive while performing calculations. Need to stop Webina but can't close this tab? Use your browser or operating-system Task Manager.

- Once Webina is done, you should get a results page that looks something like this:



Webina 1.0.3

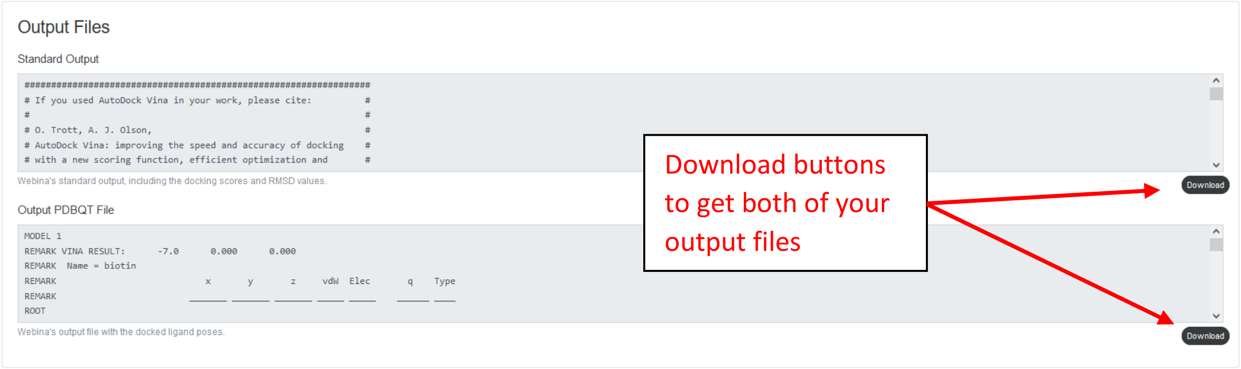
AutoDock Vina Ported to WebAssembly

Brought to you by the Durrant Lab.

Visualization

Mode	Affinity (kcal/mol)	Dist From Rmsd L.B.	Dist From Rmsd U.B.
1	-7	0	0
2	-6.5	2.257	2.833

- Scroll down to find the files we want, the Standard Output file (a text file called stdout.txt) and the Output PDBQT File (called webina_out.pdbqt), which you can get by hitting the correct “Download” buttons:



```
##################################################################
# If you used AutoDock Vina in your work, please cite:           #
#                                                                #
# O. Trott, A. J. Olson,                                         #
# AutoDock Vina: improving the speed and accuracy of docking     #
# with a new scoring function, efficient optimization and        #
```

Webina's standard output, including the docking scores and RMSD values.

```
MODEL 1
REMARK VINA RESULT:      -7.0      0.000      0.000
REMARK  Name = biotin
REMARK                          x       y       z     vdW  Elec       q    Type
REMARK                       _______ _______ _______ _____ _____    ______ ____
ROOT
```

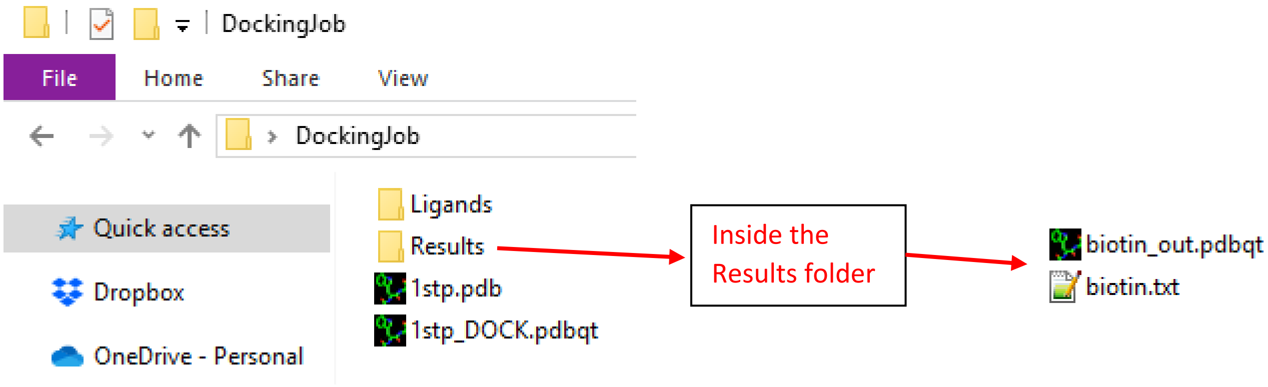
Webina's output file with the docked ligand poses.

Download buttons to get both of your output files

- Create a **Results** folder in your **DockingJob** folder, then place these downloaded files in the **Results** folder.
    - Rename the **webina_out.pdbqt** file to **biotin_out.pdbqt**
    - Rename the **stdout.txt** to **biotin.txt**

Important: You need to remember to rename these files as soon as you make them, and to keep everything organized. Although we run just one molecule in this tutorial, you will be running docking jobs with multiple ligands. Webina always gives the same default name to the output files (stdout.txt and webina_out.pdbqt), so files can easily get confused/overwritten if you aren't careful about naming them. So, get in the habit of renaming them as soon as you put them in the **Results** folder!

- You **DockingJob** folder should now look something like this:



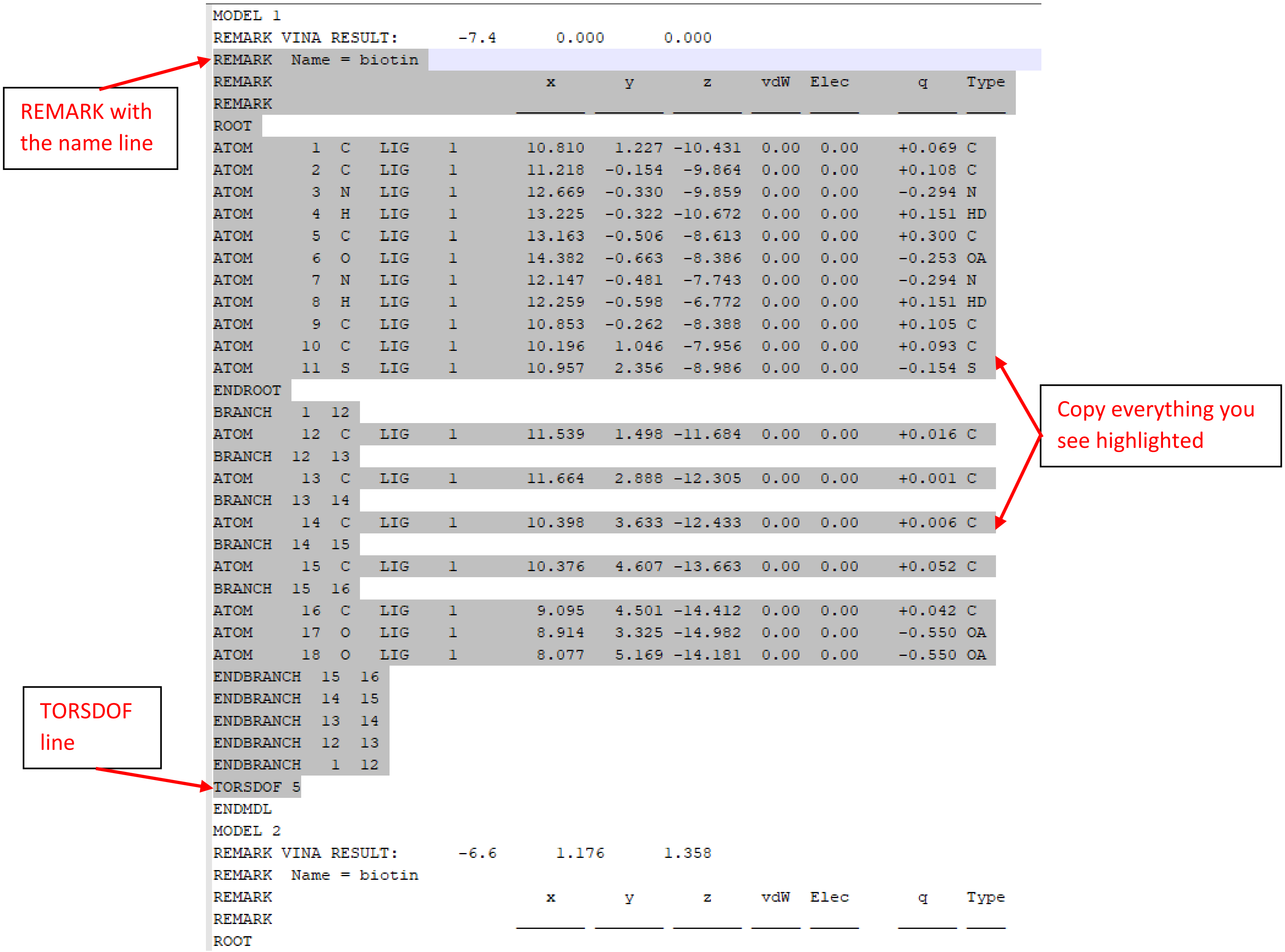
Inside the Results folder

Note: If you are running more than one ligand, then you need to repeat **Step 4** for each and every ligand, one at a time. It's recommended that you store all of your individual ligands in the **Ligands** folder, and put the output files in the **Results** folder to help keep track of everything. Be sure you download all of your output files before you hit the “Start Over” button on Webina!

Step 5: Processing your docked ligand into a PDB file

To properly examine our output ligand in ChimeraX, we need to convert the output of the Vina program into a PDB file that we can see.

- Open your **Results** folder
- Open the **biotin_out.pdbqt** file with a text editor
    - You should notice that this file contains ~10 different models of biotin, which each molecule the result of a different simulation run (named model 1, model 2, etc.)
    - The results are always organized so model #1 = best
- We want to take the best result (model #1 from the **biotin_out.pdbqt** file) and turn it into a PDB file. We are going to use the same online server we used in Step 3 to do this.
- Copy everything from REMARK with the name of the ligand to the TORSDOF line:



```
MODEL 1
REMARK VINA RESULT:      -7.4      0.000      0.000
REMARK  Name = biotin
REMARK                            x       y       z     vdW  Elec       q    Type
REMARK                         _______ _______ _______ _____ _____    ______ ____
ROOT
ATOM      1  C   LIG     1      10.810   1.227 -10.431  0.00  0.00    +0.069 C
ATOM      2  C   LIG     1      11.218  -0.154  -9.864  0.00  0.00    +0.108 C
ATOM      3  N   LIG     1      12.669  -0.330  -9.859  0.00  0.00    -0.294 N
ATOM      4  H   LIG     1      13.225  -0.322 -10.672  0.00  0.00    +0.151 HD
ATOM      5  C   LIG     1      13.163  -0.506  -8.613  0.00  0.00    +0.300 C
ATOM      6  O   LIG     1      14.382  -0.663  -8.386  0.00  0.00    -0.253 OA
ATOM      7  N   LIG     1      12.147  -0.481  -7.743  0.00  0.00    -0.294 N
ATOM      8  H   LIG     1      12.259  -0.598  -6.772  0.00  0.00    +0.151 HD
ATOM      9  C   LIG     1      10.853  -0.262  -8.388  0.00  0.00    +0.105 C
ATOM     10  C   LIG     1      10.196   1.046  -7.956  0.00  0.00    +0.093 C
ATOM     11  S   LIG     1      10.957   2.356  -8.986  0.00  0.00    -0.154 S
ENDROOT
BRANCH   1  12
ATOM     12  C   LIG     1      11.539   1.498 -11.684  0.00  0.00    +0.016 C
BRANCH  12  13
ATOM     13  C   LIG     1      11.664   2.888 -12.305  0.00  0.00    +0.001 C
BRANCH  13  14
ATOM     14  C   LIG     1      10.398   3.633 -12.433  0.00  0.00    +0.006 C
BRANCH  14  15
ATOM     15  C   LIG     1      10.376   4.607 -13.663  0.00  0.00    +0.052 C
BRANCH  15  16
ATOM     16  C   LIG     1       9.095   4.501 -14.412  0.00  0.00    +0.042 C
ATOM     17  O   LIG     1       8.914   3.325 -14.982  0.00  0.00    -0.550 OA
ATOM     18  O   LIG     1       8.077   5.169 -14.181  0.00  0.00    -0.550 OA
ENDBRANCH  15  16
ENDBRANCH  14  15
ENDBRANCH  13  14
ENDBRANCH  12  13
ENDBRANCH   1  12
TORSDOF 5
ENDMDL
MODEL 2
REMARK VINA RESULT:      -6.6      1.176      1.358
REMARK  Name = biotin
REMARK                            x       y       z     vdW  Elec       q    Type
REMARK                         _______ _______ _______ _____ _____    ______ ____
ROOT
```

REMARK with the name line

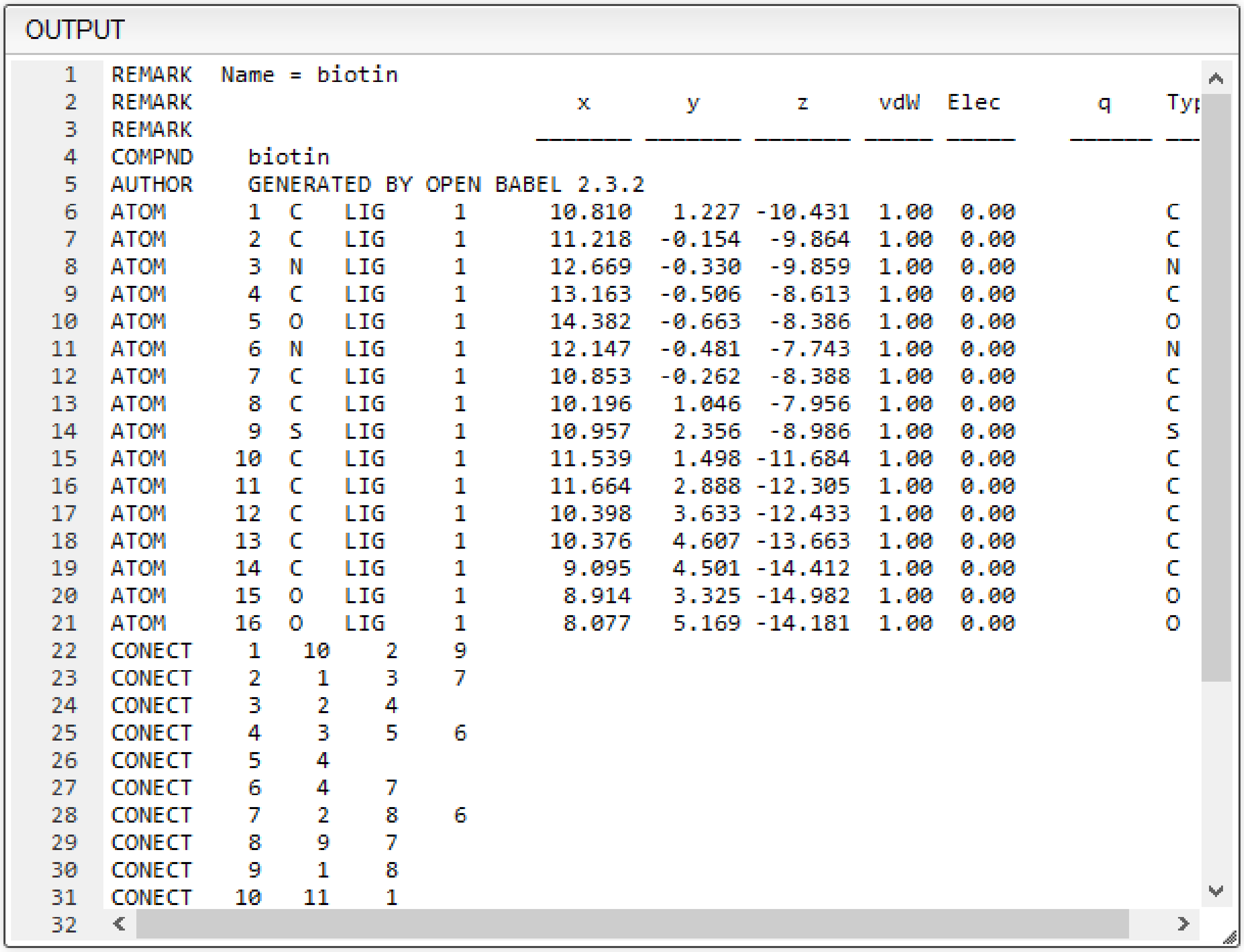
Copy everything you see highlighted

TORSDOF line

- In a web browser, return to the OpenBabel server you used to make your ligands:

https://www.cheminfo.org/Chemistry/Cheminformatics/FormatConverter/index.html

- In the “INPUT” window, paste the lines you copied from your **biotin_out.pdbqt** file (the highlighted text shown in the previous picture)
- In the “OPTIONS” pane, select the following:
    - “Input format” should be “pdbqt -- AutoDock PDBQT format”
    - “Output format” should be “ pdb -- Protein Data Bank format”
    - “Generate coordinates” should be blank (empty)
    - “Add / Delete hydrogens” should be set to “Delete”
    - “pH to add hydrogens” should be blank (empty)
- Hit the “Convert” button to get your new PDB file in the “Output” pane.

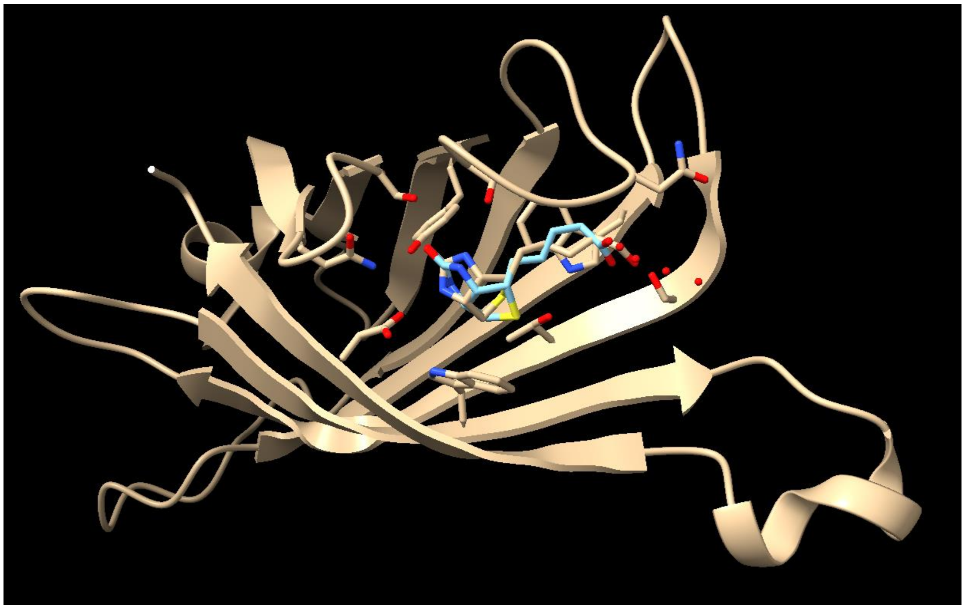


```
REMARK  Name = biotin
REMARK                                x       y       z      vdW   Elec        q     Typ
REMARK                             _______ _______ _______ _____ _____      ______ ___
COMPND    biotin
AUTHOR    GENERATED BY OPEN BABEL 2.3.2
ATOM      1  C   LIG     1      10.810   1.227 -10.431  1.00  0.00           C
ATOM      2  C   LIG     1      11.218  -0.154  -9.864  1.00  0.00           C
ATOM      3  N   LIG     1      12.669  -0.330  -9.859  1.00  0.00           N
ATOM      4  C   LIG     1      13.163  -0.506  -8.613  1.00  0.00           C
ATOM      5  O   LIG     1      14.382  -0.663  -8.386  1.00  0.00           O
ATOM      6  N   LIG     1      12.147  -0.481  -7.743  1.00  0.00           N
ATOM      7  C   LIG     1      10.853  -0.262  -8.388  1.00  0.00           C
ATOM      8  C   LIG     1      10.196   1.046  -7.956  1.00  0.00           C
ATOM      9  S   LIG     1      10.957   2.356  -8.986  1.00  0.00           S
ATOM     10  C   LIG     1      11.539   1.498 -11.684  1.00  0.00           C
ATOM     11  C   LIG     1      11.664   2.888 -12.305  1.00  0.00           C
ATOM     12  C   LIG     1      10.398   3.633 -12.433  1.00  0.00           C
ATOM     13  C   LIG     1      10.376   4.607 -13.663  1.00  0.00           C
ATOM     14  C   LIG     1       9.095   4.501 -14.412  1.00  0.00           C
ATOM     15  O   LIG     1       8.914   3.325 -14.982  1.00  0.00           O
ATOM     16  O   LIG     1       8.077   5.169 -14.181  1.00  0.00           O
CONECT    1   10    2    9
CONECT    2    1    3    7
CONECT    3    2    4
CONECT    4    3    5    6
CONECT    5    4
CONECT    6    4    7
CONECT    7    2    8    6
CONECT    8    9    7
CONECT    9    1    8
CONECT   10   11    1
```

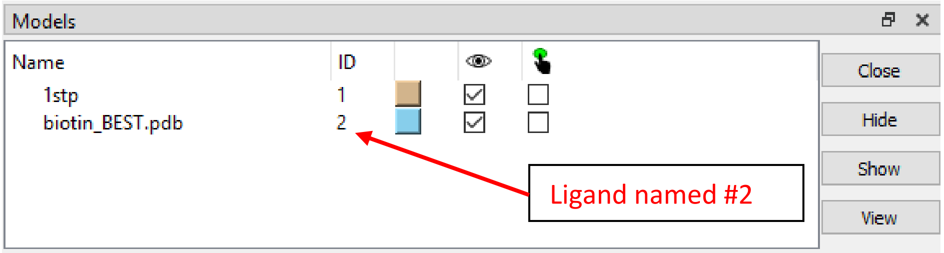
- Back in your **Results** folder, create a text file called **biotin_BEST.txt**
    - Mac users--be sure to use BBedit when you make this text file
- Paste the contents of the “Output” pane into the **biotin_BEST.txt** file.
- Change the file extension of the **biotin_BEST.txt** file to **biotin_BEST.pdb**
    - You can also do this by renaming the file using the Save As command

Step 6: Visualizing your docked ligand & finding contacts for your docked drug

- Open ChimeraX
- Using the “Open” command under the “Home” tab, open the file **1stp.pdb**
- Use the “Open” command to also open your **biotin_BEST.pdb**
  - Ignore any warnings that ChimeraX gives about bad lines
- You should see something like this:



- Your docked biotin is the light blue colored chain. We can visually see that the docked molecule is in the same pocket and more or less lines up with original biotin ligand (there is some wiggle in the chain, but the core is in the same orientation)
  - Your exact orientation may be slightly different than what you see, but the biotin molecules should “stack” pretty well.
- **Save this final ChimeraX session file!** This is everything we have worked for!
- Before we try to find contacts, pay close attention to your “Models” pane:

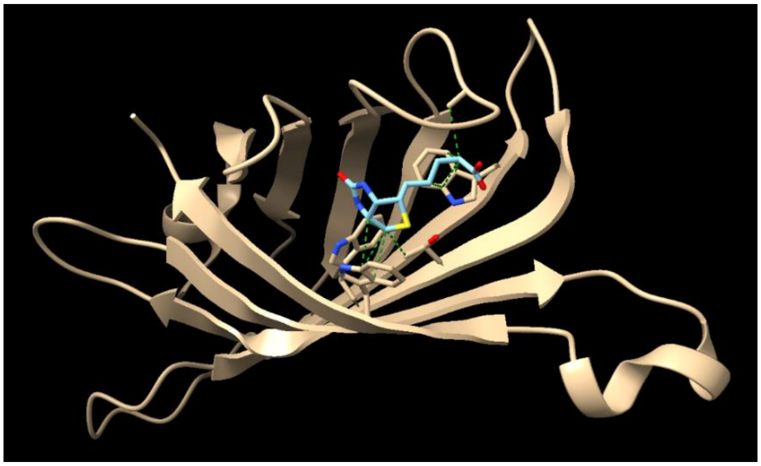


Name	ID		👁	
1stp	1	🟫	☑	□
biotin_BEST.pdb	2	🟦	☑	□

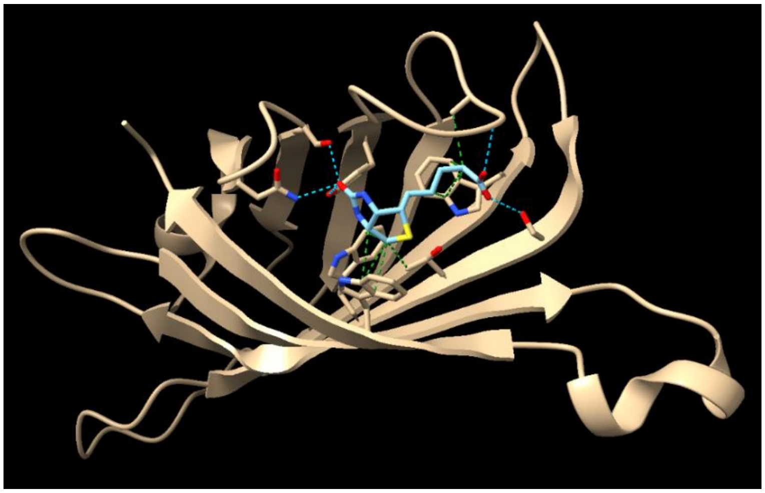
Ligand named #2

- Note that our docked ligand appears with the ID of “2,” meaning that we need to refer to it as #2 in the following commands.
  - You can easily change the color of your docked ligand by clicking on the sky-blue box and selecting a different color.
  - If you opened the files in a different order that described at the beginning of this step, you may see a different number. Just use whatever number appears under the ID.

- Let’s start finding contacts of our docked molecule by removing the original ligand and then hiding atoms with the following commands:
    - $ delete :BTN
    - $ hide atoms
- Next, find the vdW contacts between your docked molecule and the protein:
    - $ contacts /a@C* restrict #2@C* distance 3.8 reveal true name vdW
        - All molecules have vdW, so if you don’t see anything, there must be a problem with the command that you entered. Double-check all spelling!
        - Remember that our docked molecule is recognized by the program as having the name “#2” as listed in the “Models” pane.
        - If you didn’t remove the original :BTN, then it’ll be stacked on top and you’ll get a mess. You need to delete the original ligand for this part to work.



- Next, find your H-bonds with the following command:
    - $ hbonds #2 restrict protein reveal true
        - Not all ligands have H-bonds, so it’s fine if nothing appears in this step for other molecules. You should find them for this tutorial, however:



Once you have all of your contacts, you can finish processing the molecule by changing colors, naming nearby residues, showing surfaces, etc. by following the same process used in the ChimeraX assignment.

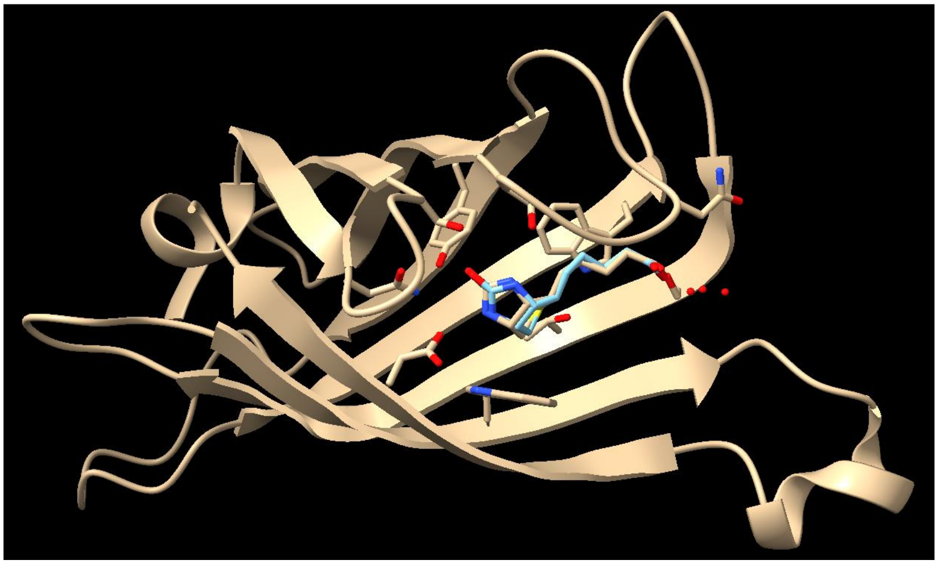
Optional: Calculating RMSD

If you're going to publish a docking study, sometimes it can help to have a known standard to compare numerically—that way, you can give a number indicate how well your parameters work when considering a positive control (we will not be doing this, so this whole section is optional and only for the curious). To provide a hard number, we often use the RMSD (root-mean-square deviation of atomic positions). You can read more about how the numbers are actually calculated here:

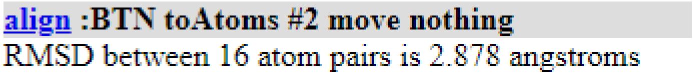
https://en.wikipedia.org/wiki/Root-mean-square_deviation_of_atomic_positions

Here's a quick tutorial on how to do this with our biotin structure:

- Open ChimeraX
- Using the "Open" command under the "Home" tab, open the file **1stp.pdb**
- Use the "Open" command to also open your **biotin_BEST.pdb**
    - Ignore any warnings that ChimeraX gives about bad lines
- You should see something like this:



- To confirm our accuracy, we are going to calculate the RMSD (Root Mean Square Deviation) with ChimeraX. This gives us a measure of the variance (how closely we matched the known answer)
- Calculate the RMSD between the two biotin molecules with the following command:
    - $ align :BTN toAtoms #2 move nothing



```
align :BTN toAtoms #2 move nothing
RMSD between 16 atom pairs is 2.878 angstroms
```

- The align command only words if you have the exact same set of atoms
- The "toAtoms" means to use the different atom types as a comparison. In other words, N to N, C to C, etc.
- The "move nothing" tells ChimeraX not to physically move the molecules.

    - Docked ligands can usually be selected using the # tag, like the "#2" above. You can see these tags if you hover your mouse over the docked ligand.
- As far as RMSD values go, < 1.5 is great, 1.5-3.0 is good, 3.0-5.0 is okay, and 5.0-7.5 is not great, and >7.5 is bad.
    - Our number (2.878) is good, and indicates success! Your number may be different, but it doesn't have to be "good" for the purposes of this exercise.

Note: If you decide to do the RMSD for your actual assignment, don't be concerned if your calculated RMSD is more than 5. Since we won't be publishing these results, it's not really a big concern. If we wanted to improve this score, then we'd have to consider changing our parameters or consider using a different docking program. In fact, when running a screen in research, we often test several programs and compare the RMSD values to see which program works "best."

Even more extra stuff for the curious

If you want to learn more about the Vina program, you can find the publication describing it here:

https://doi.org/10.1002/jcc.21334

Vina was funded through the NIH by taxpayers, so that is why it's freely available.

Frequently asked questions

1) How do you make PDBQT files? What if I want to make my own and play around with another target?

The PDBQT file is created by me and is not discussed in this tutorial. It is a specific file type only used by docking software, and requires additional programing skills to prepare, as well as the program OpenBabel (which is free). If you are interested in how these are created (or want to create your own), you can find a link to instructions on the same page of my research website (the Docking in the Classroom page).

2) Are there other docking programs out there? Is Vina special?

There are more than a dozen programs out there that do the same thing. A few are free and open source, while the rest are commercial. Vina is like the Mazda of docking programs. It's been around a long time and it's not pretty, but it works well and is easy to use (works on Linux, PC and Mac). I have picked Vina because it's free and is one of the few that runs on both PC and Mac.

3) Can I play around with other docking programs to see how they work?

Not easily, no. Most are only available for Linux systems, and only a few for PC. Mac versions are rare. Other free, somewhat modern programs out there include Smina (a fork of Vina) and idock.

4) What does the calculated energy value mean?

Energy values calculated by these programs mean very little from a thermodynamic point of view. Remember, you're holding the protein rigid (not letting it move), placing it in empty space, then trying to force a molecule into a pocket. This is not representative at all of an actual protein in a cell, which is a moving system surrounded by water molecules. The energy values do tell you the relative ***predicted*** ranking of one molecule versus another. So, the bigger the binding energy, the more likely it is ***predicted*** by the program to bind. Another thing to keep in mind is that small molecules will always have smaller numbers due to the way these scoring functions work.

5) What is a "good" energy value to have?

Remembering that these numbers are predictions, you can use the following rules of thumb:

- -11 kcal/mol or better = great!
- -9 to -11 kcal/mole = good
- -7 to -9 kcal/mole = okay
- -5 to -7 kcal/mole = oof…
- -5 kcal/mole or lower = call the ambulance, because this thing has no pulse

Keep in mind that these rules of thumb only apply to fairly large drug-like molecules that bind tightly to a well-defined pocket. Just because you get a value of -5.2 or something doesn't mean the drug won't bind. In particular, you'll see small energy values if you work with very small & polar molecules because Vina doesn't do well with these kinds of molecules. Vina also has problems calculating metal interactions, so it often scores drugs that bind metals centers poorly. Keep in mind—we're only looking for the relative ranking of one compound versus another. You do NOT have to find something that binds with a high energy value to have a molecule that binds.

6) Can I play around with the exhaustiveness settings on Webina?

Yes, you can. Keep in mind that the default value is:

exhaustiveness = 8

Be careful when changing the "exhaustiveness" value, as this isn't just a linear multiplication. In other words, doubling this value doesn't mean that Vina will take double the time--it could be much more! The actual time depends on your available RAM and your cpu and num_modes settings. If this number is higher than your system resources, it can crash your computer. I wouldn't recommend going above 80 with this number. That, and there's a paper out there showing that it isn't worth it with Vina to go higher than 10x the default anyways. (https://doi.org/10.1016/j.jmgm.2020.107532)

Also note that by changing the exhaustiveness, you're not necessarily more likely to get the "correct" answer here. Beyond a certain point, adding more conformations just ends up forcing Vina to find more "bad" conformations, and higher "exhaustiveness" just causes Vina to spend a lot of time making tiny tweaks to small groups to find just the "right" orientation (should the methyl be twisted 12.43° or 12.42° relative to the plane of the ring?...still insufficient data….I'm sorry, I can't do that Dave.) ***Remember: Any computer simulation will be limited by our understanding of these systems, and the "shortcuts" that we take to calculate an output value in a reasonable amount of time.***