Section II

The Structure of Macromolecules: *Representing and Visualizing Macromolecules – PyMOL tutorial*

Oleg Ursu, Dan C. Fara and Tudor I. Oprea

Objective

To learn how to represent and visualize proteins using the program PyMOL.

Brief info

PyMOL is an open-source, user-sponsored, molecular visualization system created by Warren Lyford DeLano and commercialized by DeLano Scientific LLC, which is a private software company dedicated to creating useful tools that become universally accessible to scientific and educational communities. It is well suited to producing high quality 3D images of small molecules and biological macromolecules such as proteins. According to the author, almost a quarter of all published images of 3D protein structures in the scientific literature were made using PyMOL.

PyMOL is one of few open source visualization tools available for use in structural biology. The **Py** portion of the software's name refers to the fact that it extends, and is extensible by, the Python programming language.

How to

1. Install PyMOL

- go to: http://pymol.sourceforge.net/
- click on Download (top of page on the left) > Non-subscriber access
- select the package for your platform
- follow the directions to install the program

*2. Download a *.pdb file from the Protein Data Bank (PDB)*

- go to http://www.rcsb.org/pdb (the PDB homepage)

The PDBs database can be searched by keyword, author, or the PDB ID of the protein you are looking for. e.g. for RecA crystal structures by the Bell group, one can search by RecA, or the Authors last names, or the PDB ID for the RecA protein of interest (1XMV for the RecA with ADP bound).

- download the *.pdb file for 1XMV
- save the file to your disk and give it a name with a .pdb suffix.

3. Open the .pdb file in PyMOL

- open PyMOL, go to *File* > *Open* and then search for your pdb file

OR

- open PyMOL and type the following command `load “path to your *.pdb file”`

If your file is correctly loaded, it will show up the model as lines and the non-bonded atoms as crosses. Note that there are two windows opened: (i) the PyMOL Tcl/Tk GUI window, and (ii) the PyMOL Viewer one – see Figure 1.

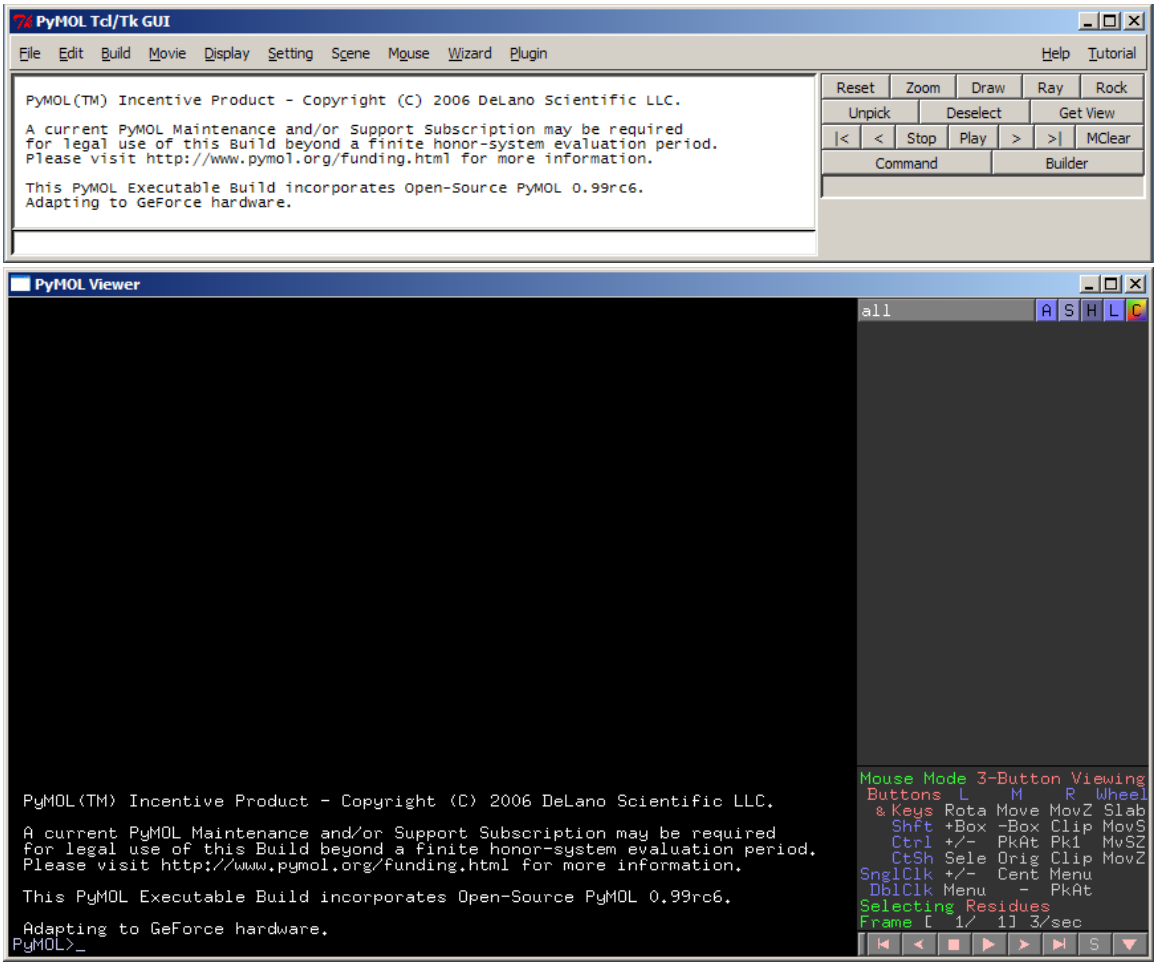

Figure 1

As you become more familiar with PyMOL you should be able to use both windows to optimize your PyMOL experience. `Typed commands` can always be retrieved with the up arrow.

4. Moving your model around in PyMOL (3-button mouse)

- Left button: rotate
- Middle button: translate X and Y
- Right button: translate Z (zoom in and out)

5. Identifying atoms

"Picking" any atom with the left mouse button will cause its identity to be displayed in the text command window. There are a number of ways to manipulate the representation of the model. On the right side of the PyMOL viewer are the names of the PDB files that are open as well as different portions of the proteins that you have selected (more on that later). Next to the names are 5 letters A(ction), S(how), H(ide), L(abel), and C(olor). If you want to change the way your molecule is displayed you can click on S and select a different depiction of your protein. For instance on the line that contains the pdb name select S and then click cartoons

(alternative command is to type: `show cartoon` in Viewer window; to get nice looking cartoons type the command: `set cartoon_fancy_helices,1`. Now your protein is modeled as a cartoon representation with helices and sheets. Also notice that the lines are still visible. To remove the line representation, select H and then *lines* (or type: `hide lines`). To hide the non-bonded residues select H and then *non-bonded* (or type: `hide nonbonded`). **Note:** if you use A, S, H, L, or C in the (all) column(s) the actions will be performed to all objects and selections.

6. Making selections

There are a number of ways that can be used for selecting and highlighting the residues of interest.

6.1 Using the select command

- `select resi 100` # will select residue 100 (note that the selection you just created appears in the right hand column and is named by default (sele), all the following commands will overwrite this selection. From here you can manipulate all the atoms or residues in that selection.)
- `select resi 100-120` # will select all the residues between 100 and 120, inclusive
- `select resi 100+120` # will select the residues 100 and 120
- `select resn asp` # will select all aspartic acid residues
- `select name CA` # will select all C-alpha carbons
- `select hetatm` # will select all non-protein atoms in the pdb file

To avoid overwriting selection it can be helpful to name the selections. This can be done in two ways.

- during the selection process

`select res100, resi 100` # will select residue 100 and # name the selection res100

- rename the selection using the *Action* button. Click on the A and select *rename selection*.

It is usually a good idea to name your selections so that you can keep track of them. You can be quite elaborate with the selection commands and combine attributes, for example:

`select resi 100-120 and (resn asp+glu)` # will yield any asp or glu residues within the residue range 100-120.

6.2 Using the sequence display

- go to *Display* > *Sequence*. This will display the single letter sequence of the protein in the pdb file as well as ligands, waters, and ions. You can click on individual amino acids or groups of residues to form a selection.

6.3 Using the secondary structure

You can select residues by secondary structure:

- `select ss h` # selects helix
- `select ss s` # selects sheets
- `select ss ""` # selects the unstructured

There is a very good explanation of using the select command in the PyMOL manual
http://pymol.sourceforge.net/html/index.html

7. Using colors

Colors can be used to highlight various aspects of a structure, i.e. the C-alpha rainbow (accessed under the C column, *spectrum* option), which makes it easier to see the overall chain trace. Also try coloring the whole molecule by *b-factor* (in the same menu).

8. Advanced

- download and open using PyMOL the PDB structure with the code 1XMS
- `select adp, resn ANP` # Create an object containing only ADP and name it adp
- select C column, *by element* and *first option* # Color this selection by atom type.

Quiz 1. What residues are in the ATP binding Pocket of RecA?

- `select adp-pocket, byres adp around 5` `#` This will select all residues around 5 Å of APD

Quiz 2. What atoms are within H-bonding distance to ADP?

- `select near_adp, adp around 3.5`

Only the atoms within 3.5 Å of adp are selected. If we want to select the residues within 3.5 angstroms we need the byres command:

- `select resi_nr_adp, byres adp around 3.5`

Quiz 3. How far away is the carboxyl O2D of Asp100 from nitrogen N6 of ADP?
First we need to find out where these atoms are. As is usually the case in PyMOL, there are multiple ways to accomplish this. We can do this by:

- `select o2d-n6, resi 100 and name OD2 or resn ADP and name N6`

Now we can see where these two atoms are. First, un-highlight your selection name. Next, by double clicking on either atom you can confirm that you have the correct two atoms selected.
OR

- having the sequence displayed, find residues 100 and the ADP and click on them both. A selection is set up (although you don't really need it) and they are highlighted.

Now that we know the atoms, we can measure the distances.

- click on *Wizard* in the Pymol Tcl/Tk GUI and select *Measurement*
- follow the directions given in the right hand panel A (the *Measurement* dialog)

The distance between the Asp100 OD2 and ADP N6 is 2.47 – see Figure 2.

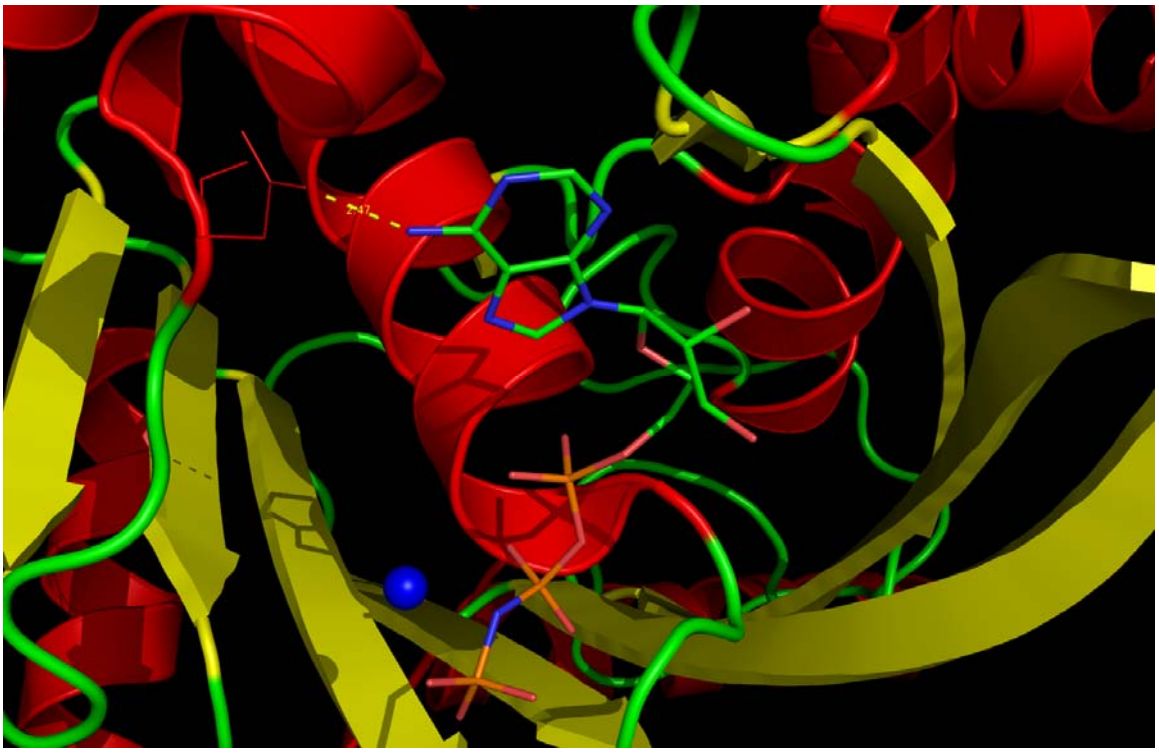

Figure 2

A measure object will also appear in the column on the right. If you want to keep measuring distance you can click on *Create New Object* to choose whether you want to create a new object with your next measurement or to just merge it with your previous measurement.

Another powerful measurement tool is the *Polar Neighbors* tool. With this tool you can click on an atom and it's nearest polar contacts are calculated. It is located on the tab directly under the *Measurement* heading in the right column.

Quiz 4. What polar contacts does the NZ of Lys72 make?

We can also find all the possible polar contacts that can be made by a particular object.

- Click on the A of the (adp) selection and then *Find > Polar Contacts > to other atoms in the object*. Note that there is now an adp_polar_conts selection. By clicking on S and then *Labels* the distances of the polar contacts are shown.

9. Aligning two molecules

- open both the 1XMV and 1XMS structures in PyMOL.

These two models are of the same protein but have different ligands in the binding pocket. With the alignment tool, we detect slight differences between similar structures. The basic command is:

- align 1XMV, 1XMS

When keyed in this way, PyMOL will try to align all residues in both pdbs (if you scroll through the command line, you'll see that PyMOL has trouble with the water molecules).

OR we can be more specific. Let's say we just want to align the protein backbone of these two proteins:

- `align 1XMV and name C+CA+N+O, 1XMS and name C+CA+N+O`

You can also align based on residues, secondary structure, and a number of other qualifiers. You should notice that these two models align very well, but are there any differences around nucleotide binding pocket? We can do this in different ways, but let's try first to select the residues around 5 Å of the ligands.

- select both the ADP and the ANP ligands:
  - `select adp, resn ADP`
  - `select anp, resn ANP`
- select the residues within 5 Å:
  - `select near5_adp, 1xmv and byres adp around 5` # by adding the <1xmv and> we are only selecting the residues in the 1xmv.pdb
  - `select near5-anp, 1xms and byres anp around 5`

Quiz 5. What residues have different orientations near the nucleotides? (see Figure 3)

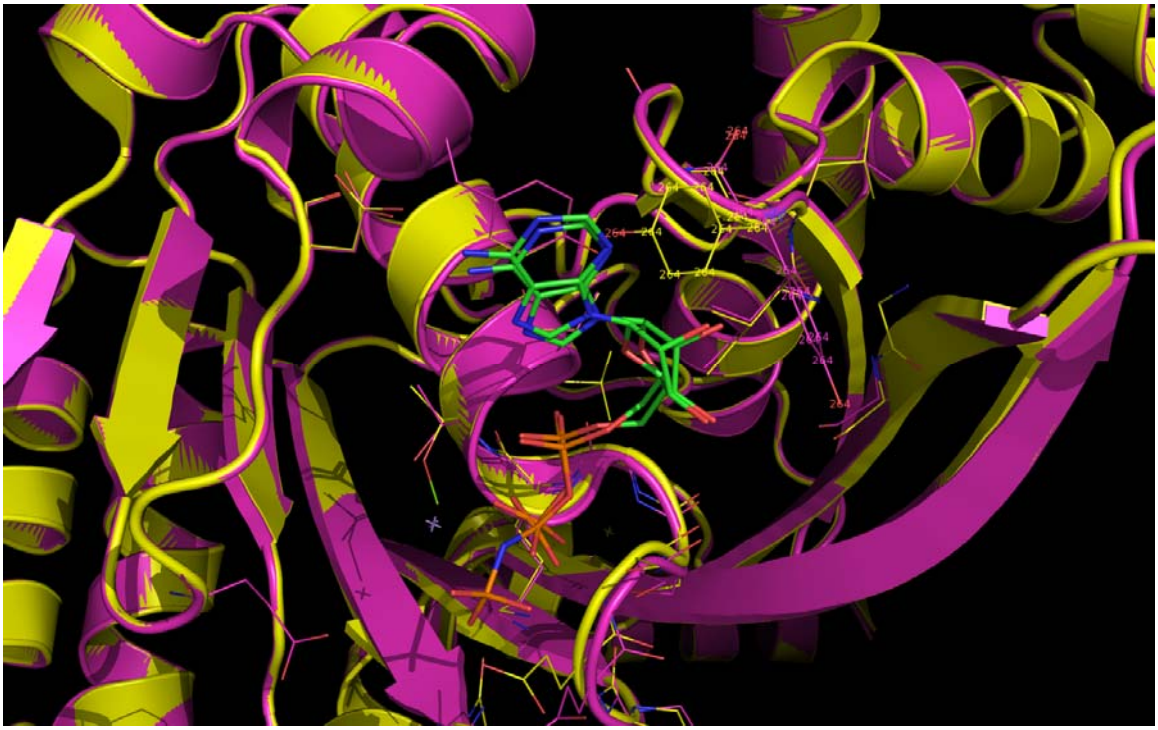

Figure 3