Lesson 6: Introduction

MODELING WITH AN ACCESSORY PROGRAM (STALKINIT); EXPLORING PEET FILES AND STRUCTURE.

In this exercise, we'll use unrealistically simple data (no noise, no missing wedge) to explore the Etomo GUI, some of the files used by PEET, and use of an aid to simplify the model task and speed alignment.

1) At a shell prompt execute
`cd $WORKSHOP_HOME/PEET_Labs/Intro-1`

2) Open *pi-a.rec* in 3dmod by running `3dmod pi-a.rec`. Adjust the zoom factor to **4** by pressing the **=** or – keys or by clicking on the up or down arrow icons, and then press **Shift+R** to re-size the ZaP window accordingly. You'll see 3 cartoons of the Greek letter pi, with differing orientations but all conveniently (and unrealistically!) centered in the XY plane.

3) Rather than simply placing a model point near the center of each particle, we'll create an initial, accessory model, which will define the rotation axis for each particle. In 3dmod, select **Model** mode, press **Edit / Object/ Type** and select **Open** as the Object Type. Then close the Object Type dialog.

4) Middle-click in the middle of the crossbar at the top (the "head") of the left-most pi to place a model point there, then middle-click in the middle of the base of this letter pi to place the "tail" there.

This simple 2-point contour (actually the vector from tail to head) defines the rotation axis (Y) for this particle. Press **N** or **Edit / Contour / New** to create a new contour, and similarly place 2 points in the middle pi, again placing the head first. Repeat for the right-most pi. In the 3dmod info window go to **File / Save Model As** *pi-stalks.mod*. You should end up with 3 open contours with 2 points in each contour as shown above. Your first point of each contour should be the head and the second point should be the tail. Exit 3dmod.

5) This model is not suitable for use as input to a PEET alignment, since there are 2 points per particle. Instead, we'll use it as input to a program stalkInit to generate the final model in addition to a RotAxes file and an initial motive list. To learn more about stalkInit run `PEETHelp` and select the entry for stalkInit or run `man stalkInit.` Use "**q**" to exit a man page invoked at the command prompt. PEETHelp does not require Internet access, although the results will be displayed in a web browser.

6) Suppose you were familiar with stalkInit, but had forgotten the arguments it needs. Run `stalkInit` with no arguments, and it will print a brief usage message summarizing the available arguments. Most PEET programs follow this convention. PEETHelp and plotFSC, which work with no arguments, are exceptions.

7) Run `stalkInit pi-stalks.mod 1 0`

The first argument, 1, means that we will be using particle 1 (the pi on the left) as our reference particle. The 3d argument, 0, says to suppress random rotations about the particle Y-axes. Often you can just run stalkInit with only the name of the input model. In this case, since the particles are already in the XY plane, random axial rotation would be counter-productive.

8) stalkInit will list the names of the three output models it generates. They differ only in whether the subvolume will be centered at the head (*head.mod*), the tail (*tail.mod*), or their midpoint (*centroid.mod*). We want the midpoint, so we'll rename that model to *pi.mod* and delete the other two. Execute:
```
mv centroid.mod pi.mod
rm head.mod tail.mod
```

9) stalkInit also generated a rotation axes file, *pi-stalks_RotAxes.csv*, and an initial motive list, *pi-stalks_InitMOTL.csv*. Notice that both types of files end in .csv, which stands for "comma-separated-value". CSV files are text files containing simple values separated by commas. They're used for several purposes in PEET, since they're human readable and editable with text editors or spreadsheet programs. Please keep in mind that while all motive lists in PEET are CSV files, not all CSV files are motive lists!

10) Examine the contents of the RotAxes file by running `cat pi-stalks_RotAxes.csv`. You will see that it contains 3 lines (one for each particle), and each line contains 3 numbers; these are the X, Y, and Z components of the rotation axis vector for the corresponding particle in tomogram coordinates.

11) PEET runs generate many output files, so we require each run to be in its own "project" directory. We've already created a subdirectory, *PEET/run1* for you to use. Move the rotation axis file and initial motive list into that directory, renaming them as you do so:

```
mv pi-stalks_InitMOTL.csv PEET/run1/initMOTL.csv
mv pi-stalks_RotAxes.csv PEET/run1/pi_Tom1_RotAxes.csv
```

PEET can only use rotation axes files in the project directory with the above naming convention or with a different naming convention in the directory where the tomogram is located. See yaxisType on the PEET man page for more details. Putting them in the project directory, as we've done here, is recommended.

12) Next, we'll set up and do a PEET run using the Etomo GUI. Execute `cd PEET/run1`, and then run `etomo`. (Note the command must be all lower case). Select **Subvolume Averaging (PEET).** Normally, at this point, you would create a new project and enter project settings manually. We'll save some time by instead copying the settings from a project I've already set up. In the resulting **Starting PEET** dialog, select **Copy project from**, use the File Chooser to go up one directory and then into **run1Done**. Select **pi.epe**, and press **OK**. The newly copied project will initially be set up to use the same input files as the project we copied from, which is typically not what you want, so we'll correct that next.

13) On the Etomo **Setup** tab, press the **>** symbols above the **Volume**, **Model**, and **Initial MOTL** columns to display the full path to the corresponding files. You'll see that the volume and model files are from 2 directories up (**../../**), which happens to be correct in this case. The initial motive list is wrong, however. Either remove the leading **../run1Done/** prefix manually, or use the File Chooser to browse to and select the *initMOTL.csv* file in the *PEET/run1* project directory.

14) Review the various settings on the **Setup** and **Run** tabs to see if you understand them. Hover your mouse over a field or a control in the Etomo GUI to view a pop-up tool tip. You can also right-click on a blank area to access the PEET Users Guide.

15) Switch to the **Run** Tab and press **Run** at the bottom left.

16) This small project will not take long to run, even with only a few cpus. While it's running, we'll explore what's going on and look at some of the intermediate and output files. In the terminal, type `ls` to display the contents of the directory. You'll notice files *pi.epe* and *pi.prm*. The epe file, called the project file, is what you'll use to re-open this same project at a later time: `etomo *.epe`. You'll rarely need to be concerned about its contents, although it is readable text. The prm or parameter file is where settings you make in the Etomo GUI are stored, in readable MATLAB syntax. Some advanced options are available only by manually editing this file; we'll see examples of this later in the workshop.

17) When the run was started, Etomo ran a PEET program prmParser to split this job up into multiple tasks or "chunks". In the project directory, there are a series of files *pi-start.com*, *pi-001.com*, … *pi-finish.com*. Each of these contains the commands for a single chunk. Run `cat pi-start.com` to examine the contents of this file. You will see that it executes 3 programs: PEETCleanup removes output from any previous runs, while prepareRef and prepareEM, respectively, generate the initial reference and motive list for the first iteration. Once the alignment has finished, most **.com* files will be automatically deleted, but you can always re-create them if desired by running `prmParser *.prm.` Examining the com and log files in various scenarios is a good way to learn more about individual programs used in a PEET run. *pi-001* through *pi-003.com* each run alignSubset to align a single particle at iteration 1. A corresponding log file (*e.g. pi-start.log*, *pi-002.log*, *etc.*) will be created as each chunk is executed. Run `gedit pi-002.log` to examine one of these log files. As each chunk completes successfully, CHUNK DONE is appended to the end of its log file. Exit gedit when finished.

18) After the run finishes, press **Open averages in 3dmod** in Etomo. This will automatically open the Isosurface and ZaP views. Alternatively, you can run `3dmod *AvgVol*.mrc` at the command line, and open the Isosurface view using the hotkey Shift+U or from the Image menu. Adjust the Isosurface Threshold so that the pi is visible. Examine the resulting averages with 1, 2, and 3 particles in the ZaP window, using the **4th D** arrows, and in the Model View. Not surprisingly, they're nearly identical in this toy example. Exit 3dmod when finished.

Run `3dmod pi_Ref2.mrc unMaskedpi_Ref2.mrc` to examine the references used as input to search iteration 2. Notice that the references are considerably larger than the requested Volume Size (found on the Setup tab). PEET automatically chooses an appropriate reference size for you except when you supply an external reference. Notice how masking eliminates the ghostly copies of neighboring particles that are present in the unmasked references at iterations after the 1st. Appropriate masking can reduce interference from noise and extraneous structures leading to better alignments. Exit 3dmod when finished.

19) Run `cat pi_MOTL_Tom1_Iter*.csv` and examine the initial, intermediate, and final motive lists. The file named **Iter1** is the input to the first iteration (the initial motive list), while **Iter3** is the output motive list from the 2nd, final search iteration. (*i.e.* it would be input to iteration 3 if there were one). Motive lists and references are special in terms of this off-by-one numbering. Most other files are numbered by the iteration at which they are produced.

Compare the shifts (in columns **11-13**) and the Euler angles (in columns **17-19**) in the initial motive list with those in the final output. In this case, the initial motive list created by stalkInit was quite good and only minor adjustments to position and orientation are found at later iterations. Also of interest are the cross-

correlation coefficients in column 1. In this case, they are close to 1 indicating nearly perfect correlation. If only real data behaved like this!

20) After alignment, we have an initial model giving starting positions, and the final motive list containing changes to those locations as well as rotations to be applied. In many cases, it will be desirable to create new model(s) containing the final, corrected positions and new motive list(s) containing only the rotations. This is easy to do. Run `createAlignedModel *.prm` and note the names of the resulting output files: *pi_Tom1_Iter2.csv, pi_Tom1_Iter2_Summary.csv, pi_Tom1_Iter2.mod, pi_Tom1_Iter2_RotAxes.csv*. If you wish, examine the newly generated motive list by running `cat pi_Tom1_Iter2.csv` and notice that the shifts are now all 0's since the new model contains the aligned locations. CreateAlignedModel has several useful properties, which we will explore in later exercises.

21) When you have finished examining your results, exit Etomo, close any remaining 3dmod windows, and clean up the directory by executing `PEETCleanup *.prm`. If you wish, `rm *~` will also remove backups; please be careful not to put a space between the * and the ~ however, or you will remove everything!

    When run with only a prm file, PEETCleanup deletes only intermediate files. When invoked with an optional 1 following the prm file, it also deletes output files, saving a backup copy of them. The later form is invoked automatically each time you start a new run.