

Molecular Replacement Tutorial

We are going to be using EPMR and CNS in this tutorial. This program can take a long time, or a very short time, depending on how good your data and model is, and how powerful your computer's processor is.

Under your tutorial directory, initiate the tutorial by typing "init_mr" and list the directory.

```
> mkdir molrep
> cp /home/tutorial/molecular_replacement/* molrep/
> cd molrep
> ls
cross_rotation.inp      model_map.inp          tex.pdb
epmr                    model_stats.inp        tex.sca
epmr.pl*               tex.cell               tex029.cv
map2omap*              tex.mtf                translation.inp
```

EPMR

To run EPMR, we need three files.

1. A 'cell' file that contains the unit cell dimensions and the space group number of your crystal (tex.cell).
2. A coordinate file of your model (tex.pdb).
3. A reflection file containing the hkl, structure factors and sigma values of the reflections (tex.epmr. Instructions to make this are written below.)

Fist, let's check out our cell file, tex.cell.

```
~/tutorial/mr> cat tex.cell
57.165      131.849      144.017      90.000      90.000      90.000  19
```

Notice the contents of this file. This file basically contains the x y z, α β γ and unit cell number (tex is in the unit $P2_12_12_1$ and is assigned the number 19 in the International Tables). You can get this information from your hkl file output from Scalepak.

You can also try to check out our coordinate file, tex.pdb. Use your favorite structure viewer or check out the text file. Are there any waters or solvent molecules in the file? If there are any, should you keep them or remove them? Do you think the water or solvent molecules help in the MR?

Now let's convert our hkl file from Scalepak into an EPMR-friendly format. Usually, the file we get from Scalepak is a file that has the hkl, intensity of our reflections (tex.sca) and sigma values. EPMR needs the file to be in a different format, with just the values for hkl, structure factors and sigma values are present. The first step is to convert the intensity values into structure factors. CNS has a command to do this conversion.

```
~/tutorial/mr> to_cns tex.sca tex.sf
```

The file `tex.sf` will now have the `hkl`, `SF` and `sigma` values that you will need. We then need to convert the `tex.sf` file into an EPMR-friendly format. Sean wrote a script (`epmr.pl`) that will do this. To run this program, type:

```
~/tutorial/mr> ./epmr.pl tex.sf > tex.epmr
```

The output file, `tex.epmr` can now be used in running EPMR.

Again, you can check if you have the necessary files by typing:

```
~/tutorial/mr> ls
cross_rotation.inp  model_stats.inp  tex.pdb  translation.inp
epmr.pl*           tex.cell        tex.sca
map2omap*          tex.epmr        tex.sf
model_map.inp      tex.mtf         tex029.cv
```

Let's start with EPMR:

```
~/tutorial/mr> epmr
```

Notice the different options EPMR gives you. The program gives you a list of options you can use to refine your search. For example, you can use "`epmr -T`" and tell the program to just do the translation search. Or you could tell it the number of molecules in your unit cell, say 2 molecules, by using "`epmr -m2`". It will then require you to give 3 files in a specific order, the cell, coordinates and reflections. Visit the EPMR website for more details.

We will now run EPMR, type in the command:

```
~/tutorial/mr> epmr tex.cell tex.pdb tex.epmr > epmr.log &
```

We can track the progress of the program by looking at its log file.

```
~/tutorial/mr> tail -f epmr.log &
```

The program will now spew out the details of the log file. Notice that it gives a bunch of text in this format:

```
. . .
Generation 42 Best = 0.5930 Average = 0.3242 Stdev = 0.0540
Generation 43 Best = 0.5930 Average = 0.3246 Stdev = 0.0532
Generation 44 Best = 0.6525 Average = 0.3275 Stdev = 0.0576
Generation 45 Best = 0.6525 Average = 0.3268 Stdev = 0.0567
Starting CG Optimization...done.
. . .
Soln 1 Rot= 301.81 68.82 222.46 Trn= 12.53 83.84 106.42 CC= 0.769 R= 0.357

Writing coordinates for best solution from 1 run(s) to 'epmr.1.pdb'.
```

Try to look at the progress of the “Best” values. Are the values generally getting larger or smaller? This value is the correlation coefficient is used to score the result. The larger the correlation coefficient, the better the probability of getting a correct answer. If you want to stop looking at the log file, you could press *Ctrl-C*.

The output of the EPMR will be a pdb file, namely **epmr.1.pdb**. The run should last about 10 minutes, depending on how fast your computer is. Check if you have this file by typing **ls** in the command line.

So how do we know if EPMR worked? What values should we calculate for or look at? We could try calculating for R and free-R value or we could draw maps to see if the electron density fits the solution.

But before we go there, lets take a look at our EPMR result. Going back to your terminal window, type:

```
~/tutorial/mr> vi epmr.1.pdb
```

Using vi, you will be able to edit the text in the pdb file, so be careful about typing anything into this file. Press *Ctrl-F* to go forward and *Ctrl-B* to go back. Peruse the text file, it looks like a normal pdb file, right? Go to the end of the file by pressing *Shift-G*. At the bottom of the file, what do you notice? You’re right, it doesn’t have an “END” at the end of the file. CNS needs pdb files to have an “END” at the end of the file. To edit this, press *i* and go the very end of the file, type **END** and then press the *Esc* button. Then type **:w** to write this file. Then quit by typing **:q**.

We can now use this file in CNS to calculate R values. Pull up a browser window and type in the address line:

file:///home/local/LINUX/cns/doc/html/cns_solve.html

This will take you to a local directory (not in the web) where you can modify input files of CNS. On the left hand side, click *Input files* and you will go to the Input files page. Scroll up and down and look at the page. There are a lot of input files available for your use and they are grouped according to the selection on the left hand side of the screen. In the interest of time and ease, I have already made the input files we will need for this tutorial. Scroll back to the top of the page. Click on *Browse* and it will open a small window from which you can choose the input file to open. You should be by default in your mr directory. Choose *model_stats.inp*. Then click *Edit* (the blue-green box in the right hand side). Note: you need to get the input files from your directory and not from the CNS page.

This will pull up a browser window that will allow you to see the different parameters you can change. Look at the space group and parameters, do they look right? If you notice, CNS allows you to choose the resolution limits of the computations. It will also do the B-factor correction and bulk solvent correction for you. You can choose which atoms to include in the statistics as well as statistical parameters. This file is ready to go, so make sure you did not change anything. Close this window. The output file we will be looking at is called “epmr.stats”. We will now run *model_stats.inp*. On the command line, type:

```
~/tutorial/mr> cns_solve < model_stats.inp > model_stats.log &
```

The model_stats will take about 5 (or less) minutes to run, depending on your computer's processing power. After it has finished running, we can check out the R values that it has computed. Find out what the computed R values are by typing the command:

```
~/tutorial/mr> head -30 epmr.stats
```

```
===== summary =====  
  
resolution range: 50 - 2.25 A  
R-values:  
initial r= 0.5392 free_r= 0.5719  
after B-factor and/or bulk solvent correction r= 0.2548 free_r= 0.2905  
  
Monitor for target "mli" is R-value :  
working set= 0.2548 test set= 0.2905  
  
luzzati coordinate error (5.0 - 2.25 A ): 0.43 A  
cross-validated luzzati coordinate error (5.0 - 2.25 A ): 0.45 A  
sigmaa coordinate error (5.0 - 2.25 A ): 0.67 A
```

This command will allow you to see the first 30 lines of the epmr.stats file. Within that first 30 lines, you will be able to find the R values. What are the R values that you see? (look for the R values after B-factor and/or bulk solvent correction) Do you think that that R values are good enough? Is this essentially a correct answer or not? (Recall the lectures... What are acceptable R values and what are not? Is the difference between the R and free R values tolerable?)

Let's check the electron density maps next. In order to make maps, we have to use another input file from CNS, model_map.inp. Go to your CNS browser again. Click *Browse* and choose model_map.inp. Take a look at the file. Again, check if the parameters are correct. Scroll down to "map generation parameters" and look at the parameters there. We will be drawing a 2Fo-Fc map, so $u=2$, $v=1$. This map will show positive electron density, so the density should match where the atoms of the molecule are. (This file is ready to go, so don't change anything on the file)

```
~/tutorial/mr> cns_solve < model_map.inp > model_map.log &
```

This will take a few minutes. The output map (epmr.map) is a huge file (~25MB) and we will convert the file into a more compact binary format. Type in the command:

```
~/tutorial/mr> ./map2omap *map
```

This will convert the map file into an omap file, which is an order of magnitude smaller and can be handled by the computer much faster. By now, you should have the following files:

```
~/tutorial/mr> ls  
cross_rotation.inp      epmr.1.pdb          epmr.coeff          epmr.log
```

epmr.map	epmr.stats	model_stats.inp	tex.epmr
epmr_negative.peaks	luz_epmr_cv.plot	model_stats.log	tex.mtf
epmr.omap	luz_epmr.plot	sig_epmr_cv.plot	tex.pdb
epmr.pl*	map2omap*	sig_epmr.plot	tex.sca
epmr_positive.peaks	model_map.inp	tex029.cv	tex.sf
epmr_r.list	model_map.log	tex.cell	translation.inp

Now let's use O to look at the structure and the maps, type:

```
~/tutorial/mr> o
```

Then, press *Enter* six (6) times. A black O screen will appear in which you can view your structure and electron density. Make the O window as large or small as you like.

I've made a set of commands strung into one major command for O so that you will not have to go through an O tutorial as well. Type in **@epmr**. This will call up a bunch of commands that will allow you to see the structure. To show the electron density, type "**fm_draw ;**". Play around with the dials (or move the mouse around) and check to see if the structure corresponds to the electron density calculated. Is there a good agreement between the electron density and structure? Can you see some of the secondary structure like alpha-helices or beta-sheets from the electron density? Do you think that the EPMR solution is correct?

As you are playing around with the structure, notice that I've centered the model and the map at an arbitrary position (a50 ca). If you wish to change the center, go ahead and center by typing the command: **Centre_atom <atom name>** in the command line.

Notice that center is spelled with a "tre" and not a "ter". Atom name comes in this format: a(number) (atom id), for example: *a155 ca* will center on residue 155 of chain a on the alpha carbon and *a192 o* will center on residue 192 of chain a on the oxygen atom. Notice that all the residues in tex have the chain identifier "a".

Look around in O and play with it as much as you like. Type **quit** in the command line to exit.

CNS

Rotation:

We will not try using CNS to do the same thing that we did with EPMR. You should still have your CNS window open, click *Browse* and select *cross_rotation.inp*. Click *Edit*. Survey this file.

As with the *model_stats.inp*, you will be able to dictate certain parameters like the space group and cell dimensions. It will ask for a *pdb* file as well as an *hkl* file. You will also set the resolution limits. Are the resolution limits in this file reasonable for a rotation search? Under 'rotation function' you can choose the 'method for cross rotation function'. Which would you choose? For the purposes of this tutorial, let us use the 'real space' method. By default, the selection is 'fast direct', so make sure to check if 'real space' is selected. Check out the rest of the file, there are several parameters you

have the power to change as you see fit. This file is ready to go so make sure you did not change anything.

To start the rotation search, type:

```
~/tutorial/mr> cns_solve < cross_rotation.inp > cross_rotation.log &
```

Cross rotation searches will take anywhere from less than 10 minutes to hours, depending on your data, model and processing power. For this particular run, it will be about 10 minutes. The output file is called `tex_xrot.list`. Take a look at this list:

```
~/tutorial/mr> cat tex_xrot.list
```

```
! index, theta1, theta2, theta3, RF-function (EPSILON= 0.25)
      1   302.568   69.070  221.752   11.7937
      65   298.975   64.884  232.453    9.8128
     148   285.332   75.349  186.842    8.9807
     177   301.961   71.163  211.961    8.8661
     235   294.425   56.512  228.572    8.5960
     241   313.148   48.140  225.720    8.5690
     282   294.301   79.535  277.938    8.4661
     426   283.701   79.535  175.701    8.2453
     463   315.888   43.953  214.638    8.2022
     735   295.629   66.977  243.129    7.9202
     744   322.461   52.326  213.514    7.9153
     803   282.453   75.349  197.547    7.8695
    1000   285.436   60.698  228.164    7.7101
```

Recall, what do these numbers mean? Did we get a clear hit in the rotation search, if so which one?

Now we can do the translation search. From your CNS window click *Browse* and select `translation.inp`. Click *Edit* and peruse the file.

As with rotation, do you think the resolution limits are reasonable? Scroll down to ‘pre-translation PC-refinement parameters’. This is the part where you can choose to do PC refinement or not. You can set the number of cycles and which parts to do PC refinement on. Under ‘translation search parameters’ is where we enter the results of our rotation search. Do you see where it is supposed to be entered? Scroll down further, what do you notice? Correct... this input file allows you to choose to do rigid-body refinement as well.

Translation:

To start the translation search, type:

```
~/tutorial/mr> cns_solve < translation.inp > translation.log &
```

This run will take about 10 minutes as well. If you type `ls` in the command line every few minutes, you will see that new `pdb` files are being written out. After the run, you will also find a `tex_rs.list` (Input `rs` for real space). Look at this list file.

```
~/tutorial/mr> cat tex_rs.list
```

		theta1	theta2	theta3	transX	transY	transZ	monitor	packing
R#	1	301.85	68.83	222.46	18.27	37.21	11.77	0.739	0.4478
R#	2	301.86	68.84	222.45	-10.31	37.22	11.79	0.739	0.4478
R#	3	283.69	76.01	185.02	7.35	10.85	39.30	0.077	0.4109
R#	4	301.86	68.84	222.46	18.28	37.22	11.78	0.739	0.4478
R#	5	301.86	68.85	222.42	18.23	37.22	83.81	0.739	0.4477
R#	6	309.74	43.51	226.22	17.41	25.07	-0.08	0.079	0.4418
R#	7	293.09	76.78	283.32	12.94	53.67	19.87	0.093	0.4212
R#	8	283.85	75.02	184.12	12.14	9.45	58.79	0.082	0.4013
R#	9	318.32	43.16	213.71	26.75	43.51	39.59	0.080	0.3558
R#	10	298.28	69.27	240.01	-0.71	36.04	11.98	0.093	0.4410

```
=== best solution ===
```

		theta1	theta2	theta3	transX	transY	transZ	monitor	packing
R#	4	301.86	68.84	222.46	18.28	37.22	11.78	0.739	0.4478

```
>>>> mean of monitor values:          0.34600  
>>>> standard deviation around mean:  0.32069
```

Look at the input file, what seems to be the best answer? Are the monitor and packing value reasonable? Recall, what does the packing value indicate? Take a look at the angles in the ‘winning’ pdb file, does this number look familiar? They should be similar or close to the results of the winning rotation in the rotation list file, tex_xrot.list.

Now we can run model_stats.inp on the ‘winning’ pdb

In the CNS window, click *Browse*, select model_stats.inp and click *Edit*. Change the file tex.pdb under ‘molecular structure’ in the ‘coordinate file’ field into **tex_rs_4.pdb**. Note, make sure that the winning pdb is indeed ‘4’ from your tex_rs.list. The number could vary depending on the run. If not, change “4” to the winning pdb in your run. Scroll to the bottom of this file and change the output names in this order:

```
epmr.stats → tex_rs_4.stats
```

```
luz_epmr → luztex_rs
```

```
sig_epmr → sigtex_rs
```

Save the file by clicking *Save updated file*. Make sure that you save it under your mr directory. You will be asked if you want to replace the file. Just click *OK*. Run the model_stats.inp as before:

```
~/tutorial/mr> cns_solve < model_stats.inp > model_stats.log &
```

Look at the output file tex_rs_4.stats

```
~/tutorial/mr> head -30 tex_rs_4.stats
```

```
===== summary =====
```

```
resolution range: 50 - 2.25 A
R-values:
initial                    r= 0.5389 free_r= 0.5714
after B-factor and/or bulk solvent correction r= 0.2537 free_r= 0.2901

Monitor for target "mli" is R-value :
working set= 0.2537 test set= 0.2901

                luzzati coordinate error (5.0 - 2.25 A ):    0.43 A
cross-validated luzzati coordinate error (5.0 - 2.25 A ):    0.47 A
                sigmaa coordinate error (5.0 - 2.25 A ):    0.66 A
```

What are the R values? Does it look like the CNS rotation and translation worked?

Congratulations! You have just successfully done molecular replacement using EPMR and CNS. I hope that this has been educational for you as it was for me. Many thanks to Sean for his data and his help with the tutorial.

Please do comment on the tutorial. If you have any questions or problems with it, don't hesitate to email me catherineydy@yahoo.com or call in at the Horvath lab 5851868. Thanks.