**Reconstruction of Symmetrical Structures with SIMPLE**

To illustrate how to use the SIMPLE suite, we provide the following workflow including all commands executed when reconstructing GroEL with D7 symmetry. It consists of three steps: 2D alignment and clustering of the images, asymmetric 3D reconstruction from the class averages and finally symmetrisation of the volume.

\*\*\* where does the dataset comes from \*\*\* 10K phase-flipped images 140x140. Throughout the different steps of the workflow a circular mask radius (msk argument) and sampling distance (smpd argument):

smpd=1.62

msk=60

1. **In plane alignment and clustering**

Prior to 2D alignment and clustering, we begin with minimizing the effect that off-centre particles could have on the subsequent steps of in-plane alignment and 2D clustering. The method is not aimed at determining the in-plane shifts exactly but roughly centering the particles in the box. We thus first bring all the particles in broad register with respect to their 2D shifts only, regardless of their in-plane orientation. This is done with the application stackops with the argument *shalgn* set to *yes* and providing our stack (stk=groel\_stk.spi), sampling distance and mask radius:

$ simple\_stackops stk=groel\_stk.spi smpd=1.62 msk=60 shalgn=yes trs=3.5 lp=20 nthr=8 outstk=stk\_sh.spi

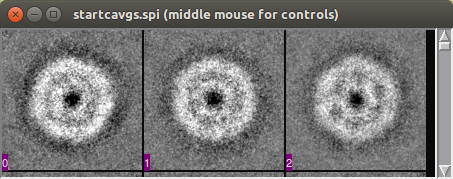
The alignment is performed with a low-pass limit of 20 Angstroms (lp=20), like in most of the subsequent steps. The iterative process will typically take a dozen iterations (a few minutes). The *trs* argument limits the shift search to the [-trs;+trs] range. We typically set the trs argument to a value to be 2.5% of the image dimension. Here the images are 140x140 pixels, so trs=3.5 pixels. There are 8 CPUs on our machine so we set nthr to 8. If you have less (or more) adjust this number accordingly.

A new centred stack (named according to the outstk argument) will be output, which we will use for the reminder of the workflow. A document named shiftdoc.txt by default that contains the calculated shifts is also created.

Next we generate random class averages to initiate the 2D clustering procedure. Given the modest size of our dataset (10,000 images) we choose 200 clusters (ncls) to obtain sufficiently populated classes. We recommended increasing this number to 500 at least for larger datasets of 30,000 to 100,000 images.

$ simple\_prime2D\_init stk=stk\_sh.spi smpd=1.62 msk=60 ncls=200 nthr=8

The simple\_prime2D\_init application will rapidly generate evenly populated class averages with random in-plane rotations, that can be visualized with your favourite image viewer. The stack of 200 class averages is named startcavgsmsk.spi and startcavgs.spi (with and without mask). In EMAN 1.9 all class averages should look like this:



We execute the 2D alignment and clustering in distributed mode:

$ simple\_stackops stk=stk\_sh.spi split=1

$ nohup distr\_simple.pl prg=prime2D stk=stk\_sh.spi oritab=prime2D\_startdoc.txt refs=startcavgsmsk.spi ncls=200 srch\_inpl=yes smpd=1.62 msk=60 lp=20 npart=1 > PRIME2DOUT &

The first instruction prepares the split stack for distributed execution. In our case we ran the clustering on a Linux workstation with 1 CPU chipset so we simply set split=1. If your machine has two chipsets you can set split to 2, keeping in mind that the npart argument in the following instruction also needs to be set to 2.

The second instruction starts the actual 2D clustering using the randomised classes as a starting point (refs argument). It will take approximately 15 iterations and under 2 hours on a modern workstation with 8 CPUs. Within the last lines of the log file PRIME2DOUT you should see the following lines:

>>> DISTRIBUTION OVERLAP: 0.9589

>>> PERCENTAGE OF SEARCH SPACE SCANNED: 99.6

>>> CORRELATION: 0.7521

>>> CONVERGED: .YES.

Our criterion for convergence is based the stability of the classes obtained. In other words, when the classes assignment are nearly identical from one iteration to the next (distribution overlap taken as >95% on average) and the particles cannot find a better matching average (search space scanned >99%, assessed by polar FT correlation) the alignment and clustering stop. In addition, each run is structured as follows. Until near convergence (search space scanned <90%) only cluster assignment and in-plane rotations are searched. After this, shifts are also searched and their limit is automatically set to 2.5% of the image dimension (see above).

Every iteration produces a folder named prime2D\_round\_XX that contains all the information to continue a run: a document with the current in-plane parameters (prime2Ddoc\_XX.txt) and two stacks of the current 200 class averages (masked and unmasked).

A number of temporary files are also created but they are only used internally and will be automatically deleted at the end of the run. As computer and network failures are part of using workstations and supercomputers you will be able to continue an interrupted run using the files present in these self-contained folders. You can also automatically remove the temporary files by simply typing: prime\_cleanup.pl. This should of course never be done while the application is running.

It is also necessary to keep the current folder organised to avoid data loss and confusion. We do not need the split stack anymore so type:

$ rm stack\_part1.spi

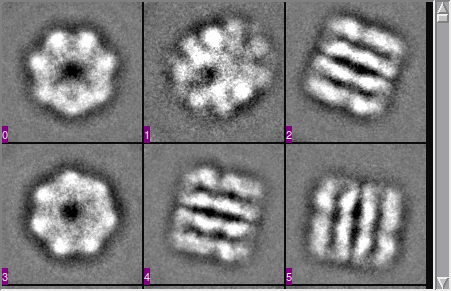
I you had used more than one CPU chipset (split>1 and npart>1) then you will have numerous files named stack\_partXX.spi, in which case you would type:

$ rm stack\_part\*.spi

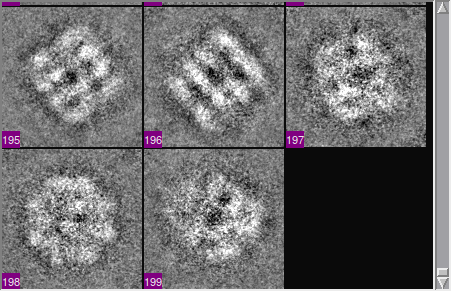
Visual examination of the 200 class averages (prime2D\_round\_15/cavgs\_iter15.spi) shows numerous images with distinctive features of GroEL such as the double ring structure and the heptameric C-symmetric rings on a uniform grey background. One can also note blurrier and images with less contrast. Typically, these correspond to lowly populated classes where the weaker SNR is likely to contribute little to the subsequent 3D reconstruction. Consequently, we rank the class averages by decreasing order of their population:

$ simple\_rank\_cavgs stk=prime2D\_round\_15/cavgs\_iter15.spi oritab=prime2D\_round\_15/prime2Ddoc\_15.txt outstk=ranked\_cavgs.spi

The stack of ranked class averages is ranked\_cavgs.spi. The first images should look like this:



In contrast, the last class averages look like this:



Now we discard the blurrier images by keeping the first 160 images. This corresponds in this case to discarding classes containing less than ~30 images per class. We simply extract the top 160 averages with:

$ simple\_stackops stk=ranked\_cavgs.spi fromp=1 top=160 outstk=selected\_cavgs.spi

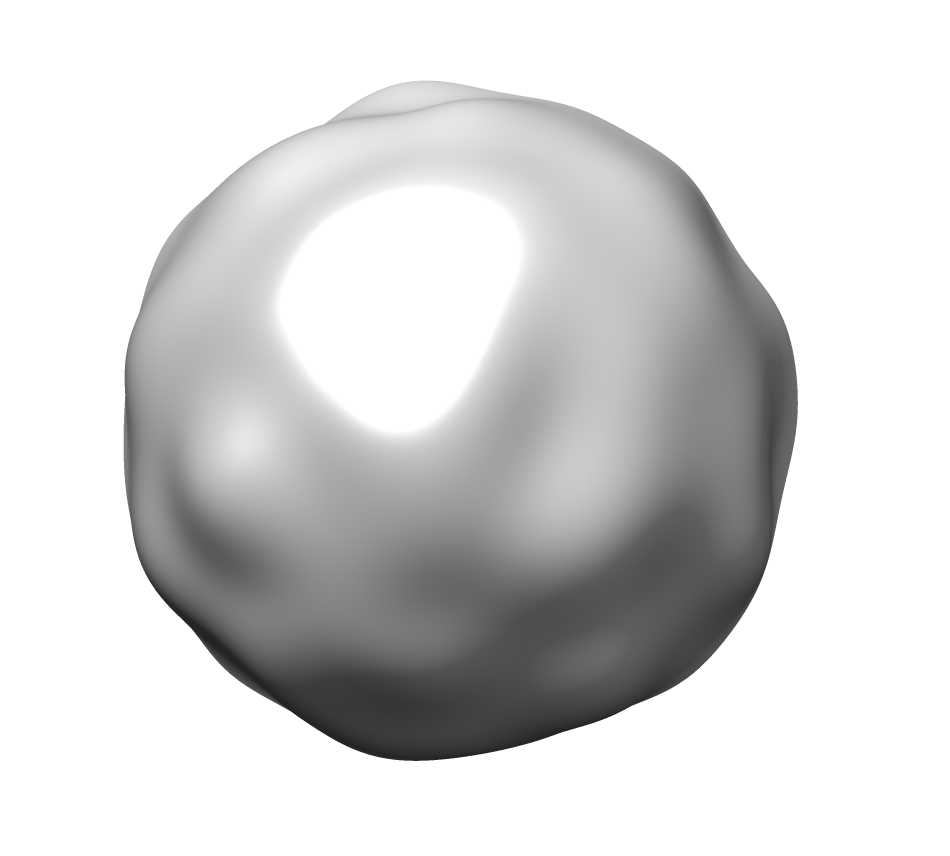
where fromp and top define the range of images to preserve. With this reduced stack (selected\_cavgs.spi) we will now generate an ab initio 3D reconstruction of the molecule.

**2. Ab initio 3D reconstruction**

We first need a random volume to initiate the search of the five in-plane and out-of-plane parameters of our selected class averages. Like the previous 2D step we execute:

$ simple\_prime3D\_init stk=selected\_cavgs.spi smpd=1.62 msk=60 nthr=8 lp=20

Consistently with section 1 we set a low-pass limit of 20 Angstroms. This command will generate two files: a volume reconstructed from random orientation parameters (startvol\_state1.spi, see below) and the document containing these parameters (prime3D\_startdoc.txt).



We can now start the search with:

$ simple\_stackops stk=selected\_cavgs.spi split=1

$ nohup distr\_simple.pl prg=prime3D stk=selected\_cavgs.spi vol1=startvol\_state1.spi smpd=1.62 msk=60 lp=20 oritab=prime3D\_startdoc.txt npart=1 > PRIME3DOUT &

Again, we first split the stack for distributed execution. Then we run PRIME3D providing the randomised orientations (oritab argument) and volume (vol1 argument) we have just prepared. After approximately 16 iterations the run converges. At the end of PRIME3DOUT you will find:

>>> ANGLE OF FEASIBLE REGION: 14.1

>>> AVERAGE ANGULAR DISTANCE BTW ORIS: 2.4

>>> PERCENTAGE OF SEARCH SPACE SCANNED: 100.0

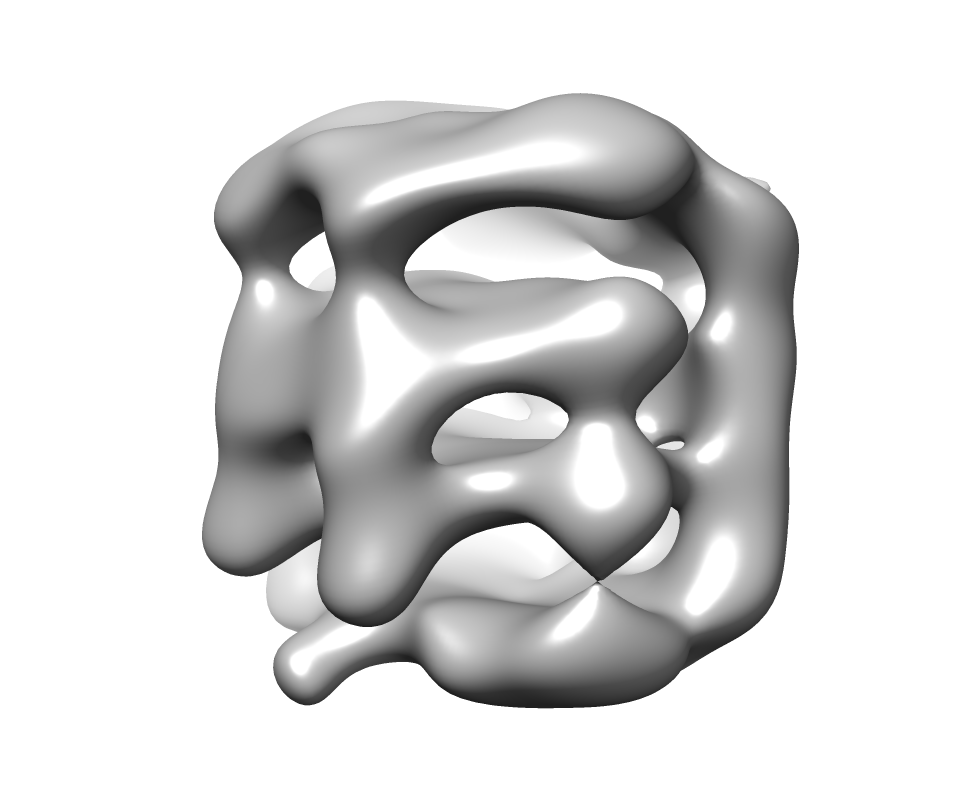
>>> CORRELATION: 0.9178

>>> ANGULAR SDEV OF MODEL: 40.81

>>> UPDATE LOW-PASS LIMIT: .NO.

>>> CONVERGED: .YES.

The recvol\_state1.spi volume and the corresponding orientation parameters (prime3Ddoc\_16.txt) are produced in the the prime3D\_round\_16 folder. Like in 2D clustering prime3D outputs all the information necessary for continuing a run in numbered folders. As we have set a low-pass limit of 20 Angstroms the volume is blobby but still captures the overall shape of GroEL. Keep also in mind that we have so far made no assumption about symmetry and the volume has been reconstructed in the C1 symmetry group.



**3. Symmetrisation and refinement of the volume.**

In order to symmetrise the volume we need to identify the best possible axis of symmetry given GroEL well-known D7 symmetry group. This is done with simple\_symsrch, given the C1 volume and orientation parameters (vol1 & oritab) and the same low-pass limit than previously (lp=20). The symmetrised orientations are output to sym\_d7.txt (outfile).

$ simple\_symsrch vol1=prime3D\_round\_16/recvol\_state1.spi smpd=1.62 msk=60 oritab=prime3D\_round\_16/prime3Ddoc\_16.txt pgrp=d7 outfile=sym\_d7.txt nthr=8 lp=20 > SYMOUT

On output the optimal axis of symmetry (pgrp=d7) identified is listed (e1,e2 and e3 fields for phi,theta and psi) along with its correlation (here corr=0.77). It is likely that you will obtain different values for the axis of symmetry. Given the stochastic nature of the 2D alignment/clustering and the 3D reconstruction the determined axis is bound to differ. Furthermore, with such D7 symmetry there are 14 (2x7) theoretically identical axis to choose from. Nonetheless, the final volume is reproducible and captures the structure of GroEL as judged by the numerous existing crystallographic and EM structures.

With the new symmetrised orientation parameters of the class averages we now obtain the particles symmetrised orientation parameters. We first create a text file called doclist.txt (with for instance emacs) that contains a single line:

sym\_d7.txt

and map the class averages orientation parameters to the particles (stk) by providing the selected class averages (stk2), all the class averages (stk3) and the in-plane parameters of the particles in the class averages (oritab):

$ simple\_map2ptcls stk=groel\_stk.spi stk2=selected\_cavgs.spi stk3=cavgs\_iter16.spi oritab=prime2D\_round\_16/prime2Ddoc\_16.txt doclist=doclist.txt nthr=8

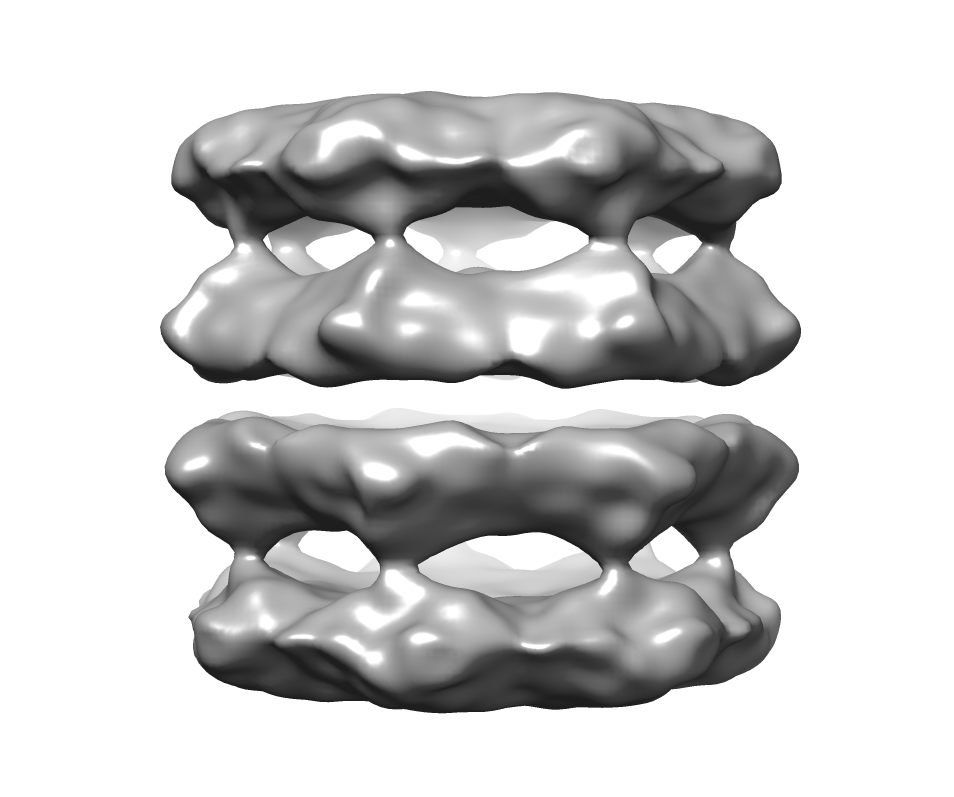
Next we reconstruct a symmetrised volume with the particles and the mapped orientation parameters (mapped\_ptcls\_params.txt output by simple\_symsrch) and specifying the symmetry group (pgrp=d7):

$ simple\_eo\_recvol stk=groel\_stk.spi oritab=mapped\_ptcls\_params.txt smpd=1.62 msk=60 nthr=8 pgrp=d7

After several minutes we obtain a new volume (recvol\_state1.spi) and its resolution:

>>> RESOLUTION AT FSC=0.143 DETERMINED TO: 12.60

>>> RESOLUTION AT FSC=0.500 DETERMINED TO: 17.45



As recvol\_state1.spi is the default name used internally by PRIME3D it is best to rename it to avoid having it overwritten:

$ mv recvol\_state1.spi sym\_recvol\_state1.spi

simple\_eo\_recvol and simple\_prime3D also produce a file that contains the spectral correlations and resolution called fsc\_state1.bin. Make sure to backup this file as PRIME3D will overwrite it if executed in the same folder. Here we just copy it because in the next step we will refine the volume and PRIME3D will require the resolution contained in fsc\_state.bin to initiate the refinment:

$ cp fsc\_state1.bin eo\_fsc\_state1.bin

Finally, we refine our initial model applying D7 symmetry with PRIME3D.

$ simple\_stackops stk=groel\_stk.mrc split=8

$ nohup distr\_simple.pl prg=prime3D stk=groel-stk.spi vol1=sym\_recvol\_state1.spi smpd=1.62 msk=60 eo=yes oritab=mapped\_ptcls\_params.txt npart=8 > PRIME3DOUT2 &

With the first instruction we split the stack for distributed execution. Here the refinement run will be split over 8 different CPU chipsets on a Linux cluster (split=8; see Section 5 of the manual for more details). We use PRIME3D differently this time. Instead of setting a 20 Angstroms low-pass limit the resolution of the volume is calculated automatically at every iteration (eo=yes), and starting from our symmetrised volume (vol1=sym\_recvol\_state1.spi). After 10 steps the final resolution obtained from the particles printed at the end of the PRIME3DOUT2 (and also stored in the fsc\_state1.bin) is better than 8 Angstroms. Examination of the volume will reveal the appearance of alpha-helical features.

>>> RESOLUTION AT FSC=0.143 DETERMINED TO: 7.32

>>> RESOLUTION AT FSC=0.500 DETERMINED TO: 8.10

