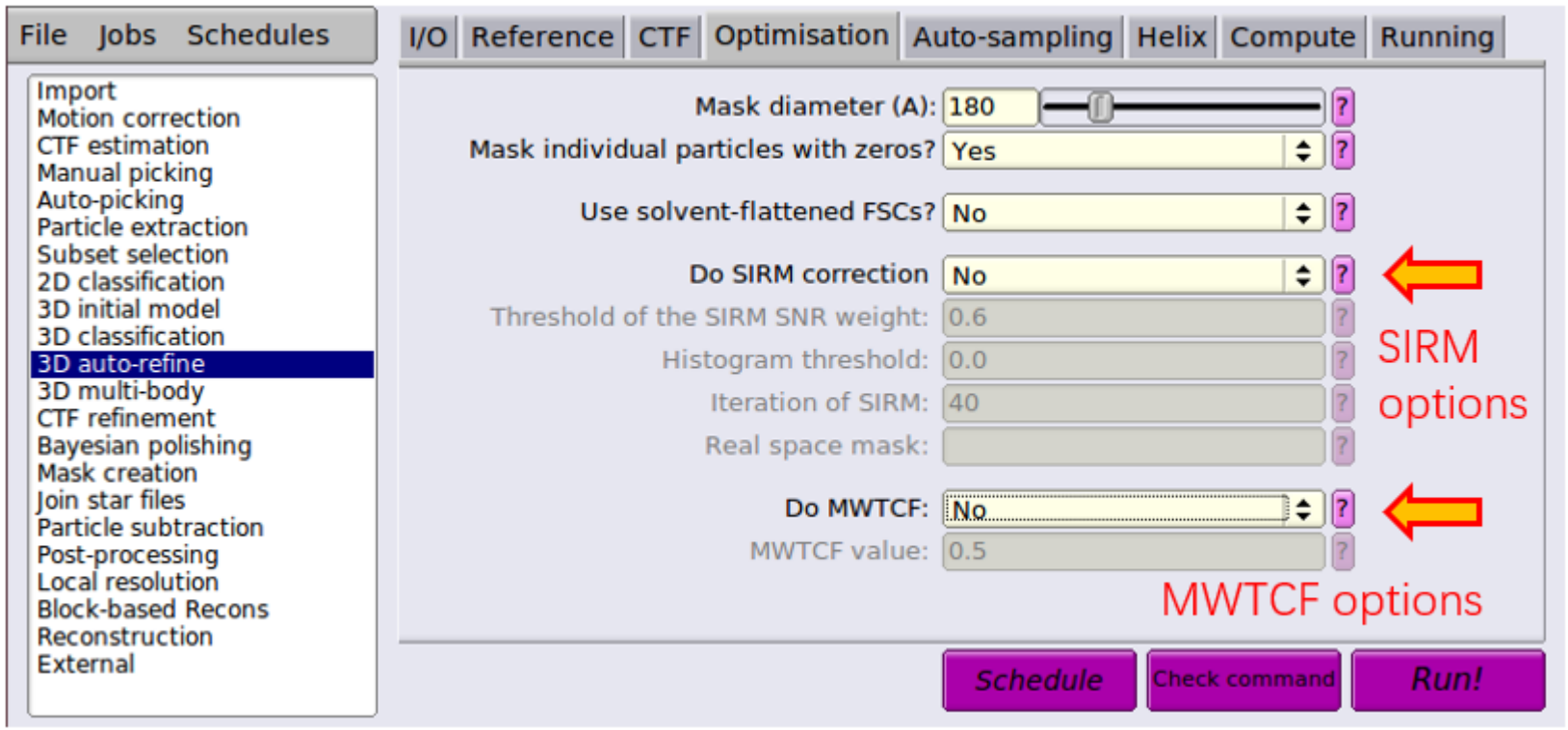
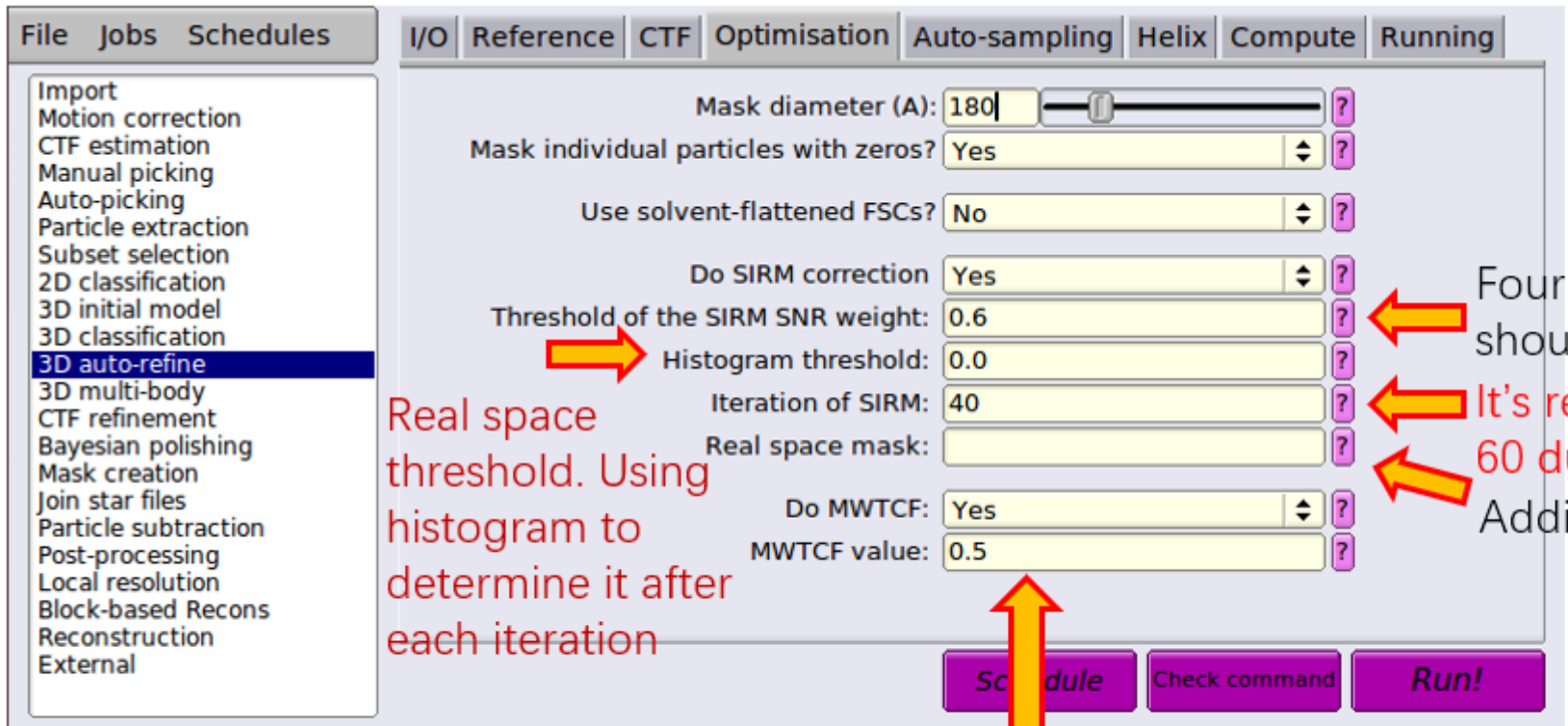
https://github.com/homurachan/SIRM_RELION

1、Triggering SIRM correction during 3D auto-refine



1、Triggering SIRM correction during 3D auto-refine



Real space threshold. Using histogram to determine it after each iteration

Fourier SNR weight threshold. It should be set by trial-and-error.

It's recommended to be less than 60 during refinement

Additional mask

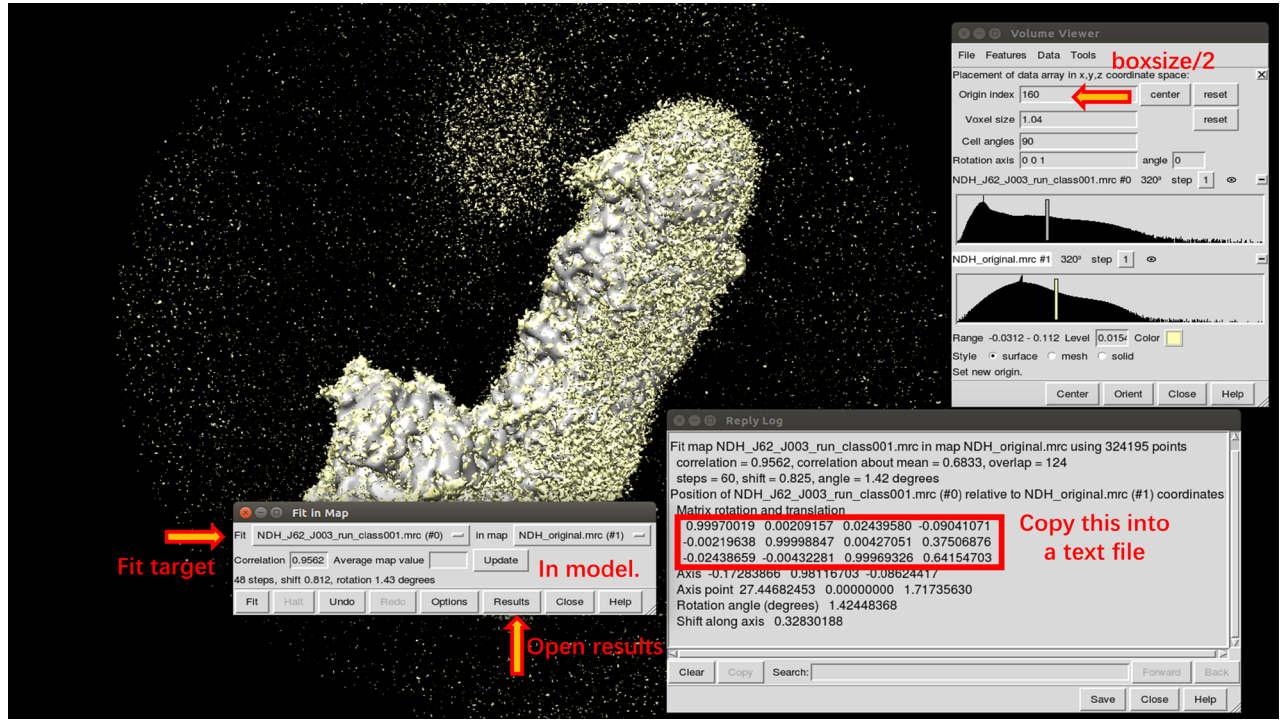
Interpolation number less than this (0.5 here)* average would not account for resolution.

2、Validation，Compare the results between RELION and CryoSPARC

1、Convert the cryoSPARC .cs file into the RELION .star file. Recommend to use the csparc2star.py from pyem package.
I suggest to use `relion_star_handler --i input.star --o output.star` that convert the output star into RELION 3.1ver

2、Refine utilizing SIRM-RELION

3、Normally, the reconstructions between cryosparc and relion would have a small rotation difference. Should be corrected first.



Use “Validation_preprocess_read_relion_star_rotate_accordingto_chimera_fitting.py” to correct.

More details: https://github.com/homurachan/Block-based-recontruction/wiki/How-to-fit-block-into-original-map

```
E:\>type J003.txt
0.99970015   0.00208527   0.02439808  -0.09048470
-0.00218999   0.99998850   0.00426617   0.37514684
-0.02438890  -0.00431832   0.99969322   0.64161460

E:\>python ./read_relion_star_rotate_accordingto_chimera_fitting.py
<input star> <chimera txt> <pixel size> <output star> <output INT translation txt>

E:\>python ./read_relion_star_rotate_accordingto_chimera_fitting.py J003.star J003.txt 1.04 J003_fit.star INT.txt
Is relion3.0? = 0
[[ 0.99970015  0.00208527  0.02439808]
 [-0.00218999  0.9999885   0.00426617]
 [-0.0243889  -0.00431832  0.99969322]]
```

4、Validation

We only need to correct the starfile that you want to use for SIRM correction. Let's call it relion_rotted.star. The results from cryoSPARC named cs_refine.star. Run the 'Validation_compare_cryoSPARC_and_relion.py' script.

python Validation_compare_cryoSPARC_and_relion.py relion_rotted.star cs_refine.star "Angluar threshold" "n for C-n symmetry" "validation_result.star"

The "angular threshold" is the maximum allowed orientation difference in deg. If larger than this threshold, the particle would be recognized as a bad one and move to the other file.

Normally, the recommended value of "angular threshold" is 2~3 for small proteins. It can also be inferred from the reported angle accuracy from RELION. Multiply that value by ~3 is generally good, but you should always try before actually use.

The "n for C-n symmetry" is designed to expand the asymmetric unit into the whole sphere. I found that even applied symmetry, sometimes the starfile from cryoSPARC is the whole sphere, which caused problems for calculation. Please check the orientational distribution graph from cryoSPARC.

4、Validation

```
71
72      for i in range(mline,len(instar_line)):
73          if(instar_line[i].split()):
74              imagename=str(instar_line[i].split()[IMG_index])
75              image_serial=int(imagename.split('@')[0])
76              image_filename_tmp=cryosparc_filename(imagename)
77              image_filename=""
78              cryosparc2relion_serial.append([])
79              cryosparc2relion_imagename.append([])
80              cryosparc2relion_line.append([])
81              for mm in range(0,len(image_filename_tmp)):
82                  image_filename+=image_filename_tmp[mm]
83                  image_filename+="_"
84              cryosparc2relion_serial[len(cryosparc2relion_serial)-1]=image_serial
85              cryosparc2relion_imagename[len(cryosparc2relion_imagename)-1]=image_filename
86              cryosparc2relion_line[len(cryosparc2relion_line)-1]=i
87
88      print("Finish reading part1")
89      orirelion_serial=[]
90      orirelion_imagename=[]
91      orirelion_line=[]
92      for i in range(mline2,len(oristar_line)):
93          if(oristar_line[i].split()):
94              imagename=str(oristar_line[i].split()[ori_IMG_index])
95              image_filename_tmp=cryosparc_filename(imagename)
96              image_serial=int(imagename.split('@')[0])
```

It should be noted that these two “cryoSPARC_filename” at Validation_compare_cryoSPARC_and_relion.py need to be assessed before use.

If your rlnImageName is cryoSPARC like, which includes random serial at the front of the filename, use the cryoSPARC_filename function. Otherwise use the relion_filename function instead.

4、Validation

The outputs are three files. “validation_result.star” stores the angular changes of each particles. It IS NOT the actual star.

validation_result.star_lessthan_”threshold”_change_line.star is the validation picked results. The orientation information is from the FIRST input star of the script. So if you want to keep the results from cryoSPARC, put that cs_refine.star at the front. However, in our tests, the results from SIRM-RELION are generally better.

The other file: validation_result.star_largerthan_”threshold”_change_line.star
stores the dropped particles.

Summary:

1、Refine the same dataset with cryoSPARC and RELION

2、Fix the rotation difference between two software

3、Set an angular threshold. It removes the particles whose angular changes are larger.

5、Mask-Picking

The Mask-Picking process is normally performed AFTER the Validation process.
Assuming we choose “validation_result.star_lessthan_”threshold”_change_line.star”.

1、utilize `relion_star_handler` from SIRM-RELION to generate BILD from this star.
`relion_star_handler --i validation_result.star_lessthan_”threshold”_change_line.star --o test.bild --write_bild` . It can be followed by --bild_angle_step and --bild_radius which are shown on the “Write bild options”.

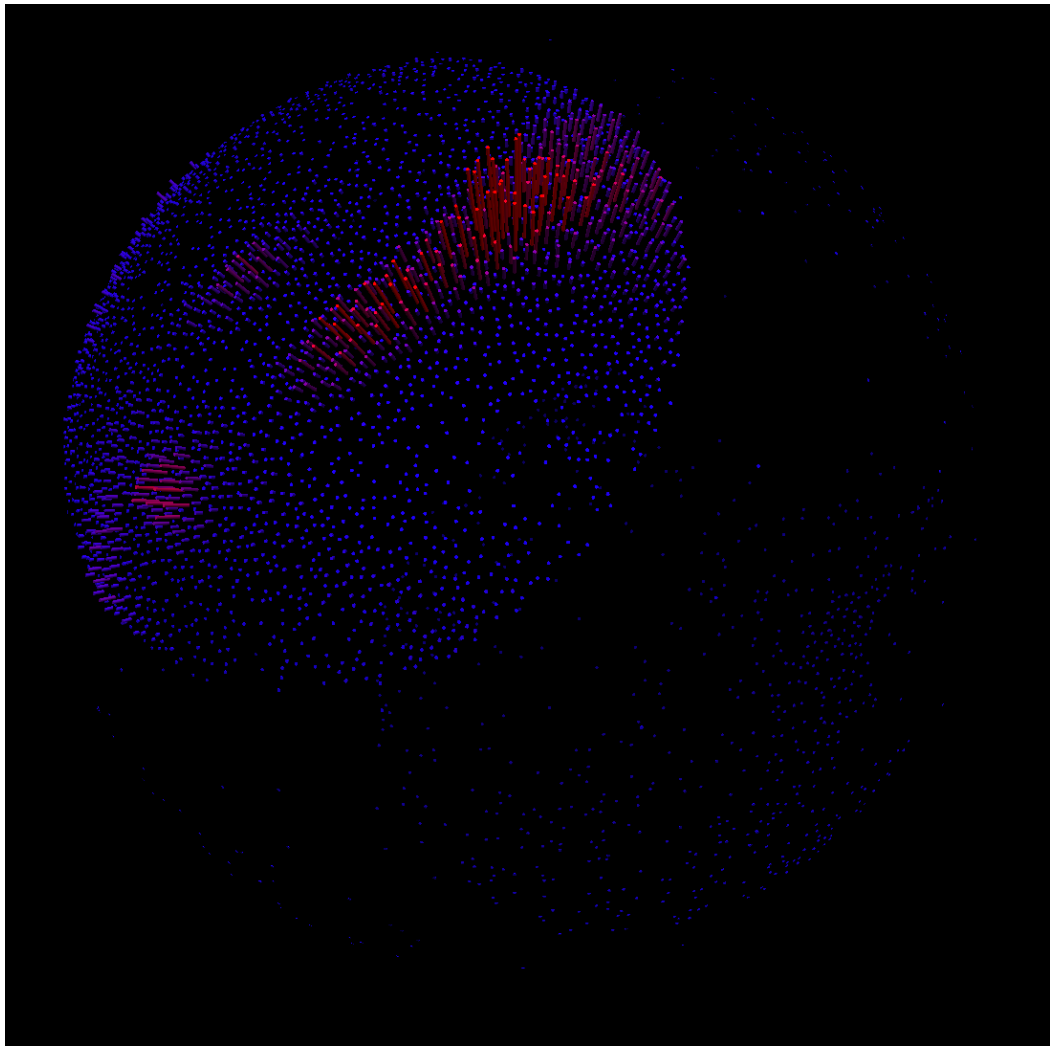
2、Use the MaskPicking_step1_read_relion_bild_produce_mrc.py to generate 3D mrc from the BILD file. Require to install numpy and mrcfile.

`python MaskPicking_step1_read_relion_bild_produce_mrc.py test.bild 0.9375 “mrc_radius” test.bild.mrc

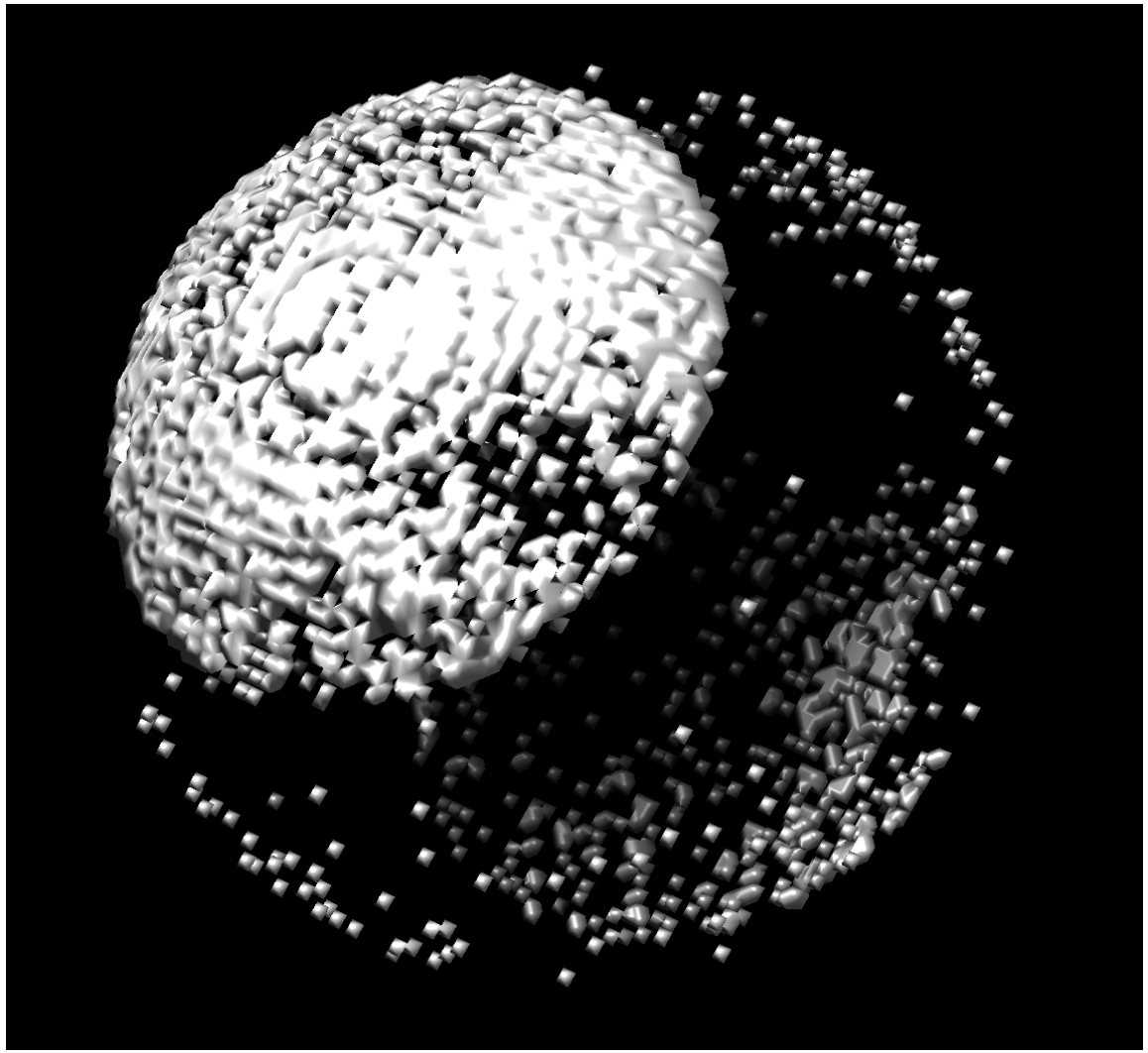
The “mrc_radius” in in pixel and should not be larger than 80, or the next few steps would be painfully slow.

5、Mask-Picking

The BILD



The generated mrc

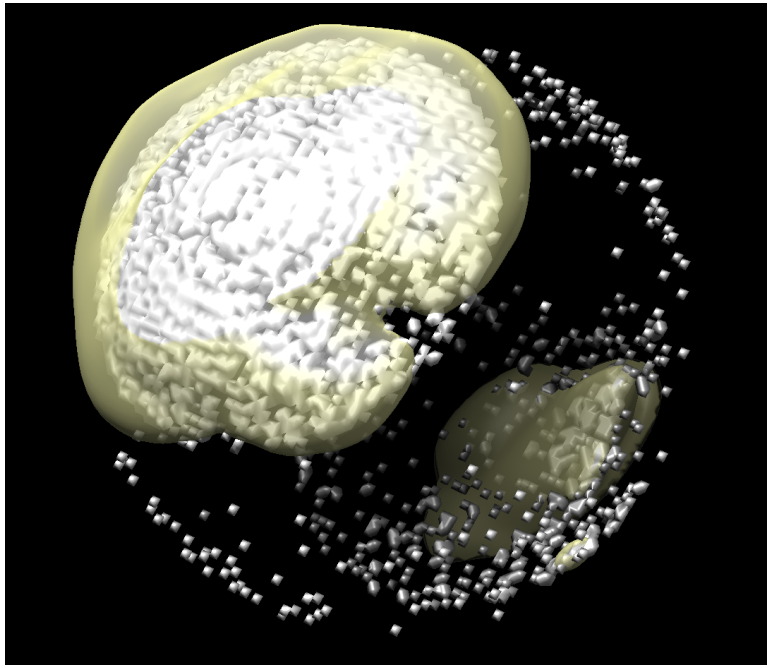


5、Mask-Picking

It's very clear that our 3D mrc needs smoothness. A very easy way is to lowpass. I recommend to use proc3d from EMAN package.

`proc3d test.bild.mrc test.bild_lp.mrc apix=0.5 lp=5`

The apix is not to be used later. It only marks that the map is low-pass to 1/5 of the Nyquist frequency.



After low-pass, pick a good threshold and erase the area that you don't want. Generate mask by using relion_mask_create.

Next step is to generate an Euler angle table to select particles. In all, all particles whose orientation stay outside of the mask will be removed.

5、Mask-Picking

Generate an Euler angle table.

Please use SIRM-RELION v1.1. The command is:

`relion_image_handler --i test_lp_erased_mask.mrc --o nouse.mrc --gen_MP_table --MP_table out_anglelist.star --MP_stepsize “angle_step”`

The angle_step is the stepsize between sampling point. Normally the default value is enough.

Check the info on the relion_image_handler

```
====== additional subtract options =====
  --optimise_scale_subtract (false) : Optimise scale between maps before subtraction?
   --optimise_bfactor_subtract (0.) : Search range for relative B-factor for subtraction (in A^2)
        --mask_optimise_subtract () : Use only voxels in this mask to optimise scale for subtraction
====== SIRM parameter. Not recommended for initial refine, except for local refinement. =====
                     --SIRM (false) : Perform SIRM reconstruction
                 --thres_SIRM (0.3) : SIRM threshold in Fourier space
                --Hist_cutoff (0.0) : cut-off of real space restrain
                 --thres_hist (0.0) : SIRM, histgram threshold in real space
                   --iter_SIRM (30) : Number of iteraion of SIRM
              --SIRM_snr_weight () : SNR weight file in RFLOAT
          --max_radius_SIRM (100.0) : SIRM, histgram threshold in real space
            --gen_MP_table (false) : Read MRC file and generate Mask-Picking table. Use together with --MP_table and --MP_stepsize
                    --MP_table () : Output name of Mask-Picking table.
             --MP_stepsize (0.9375) : Stepsize of Mask-picking table
```

5、Mask-Picking

Dropping particles using the Euler angle table.

Please use SIRM-RELION v1.1. The command is:

relion_star_handler --i validation_result.star_lessthan_"threshold"_change_line.star --o maskpick_result.star --MP_write_particles --MP_table out_anglelist.star --MP_stepsize angle_step"

Check the info on the relion_star_handler

```
====== !!!!Special option of Block-based reconstruction =====
  --do_invert_BBR_handerness (false) : Suppose your handerness was correct.
====== Write bild options =====
             --write_bild (false) : Generate bild from input star file.
        --bild_angle_step (0.9375) : Angle step of bild
            --bild_radius (128.0) : Radius of inner sphere of bild
====== Mask-Picking, read tables write particles. =====
      --MP_write_particles (false) : Enable Mask-Picking. Use together with --MP_table and --MP_stepsize.
                  --MP_table () : Mask-Picking table name
           --MP_stepsize (0.9375) : Stepsize of Mask-picking table
```

Summery:

1、After validation, convert starfile to BILD and 3D mrc.

2、Low-pass the mrc and erase the area that you don’t want to use.

3、Generate angle table and select particles.
Note: Use relion_star_handler to handle starfile. Use relion_image_handler to handler mrcfile.

6、Final reconstruction

It should be noted that the SNR weight was automatically generated during auto-refine. However, there is no such procedure during reconstruction. So we have to generate a FSC weighting file.

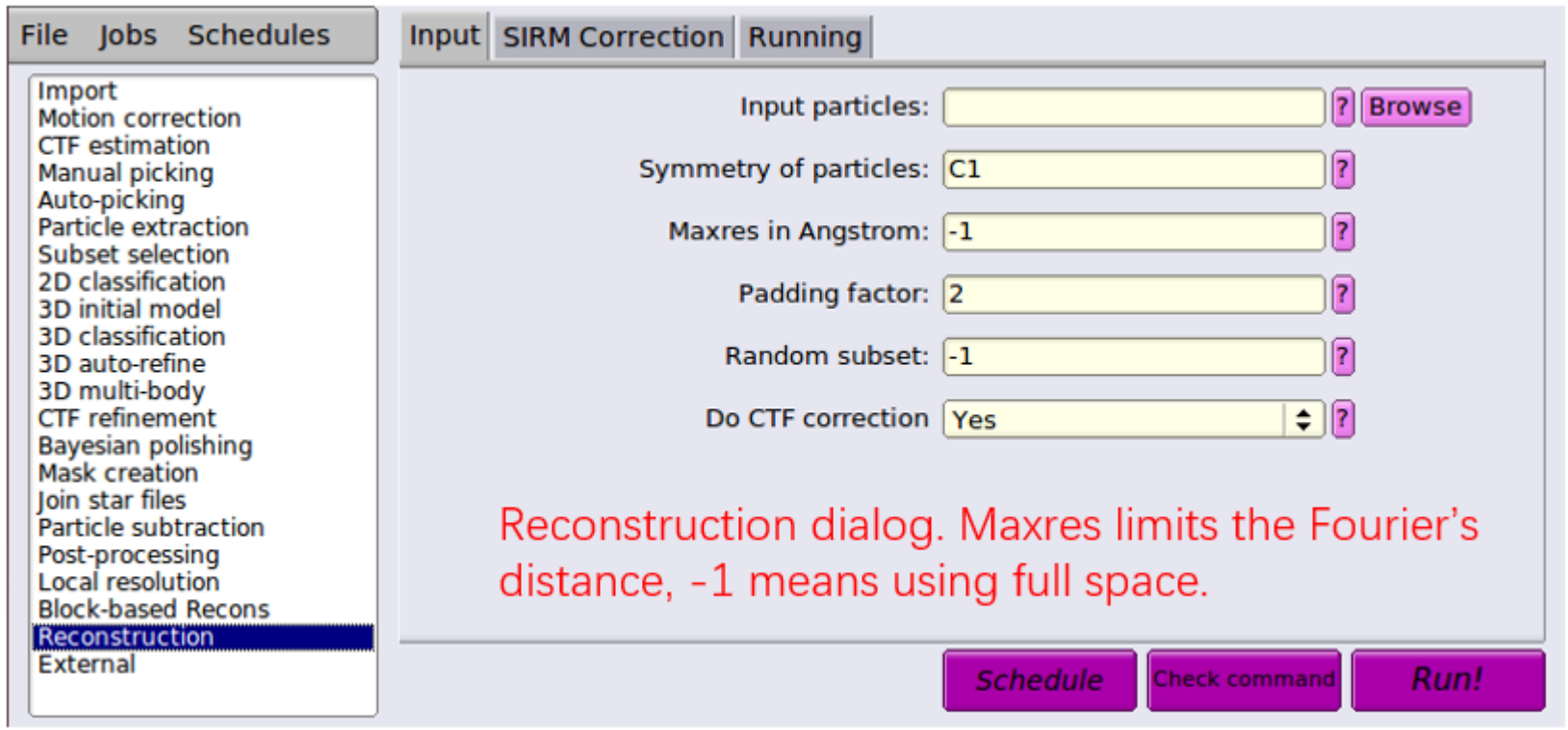
Perform relion_postprocess, then use the script convert_star_2_SIRM_weight.py.

`python convert_star_2_SIRM_weight.py postprocess.star weight.star`

to generate the FSC weight.

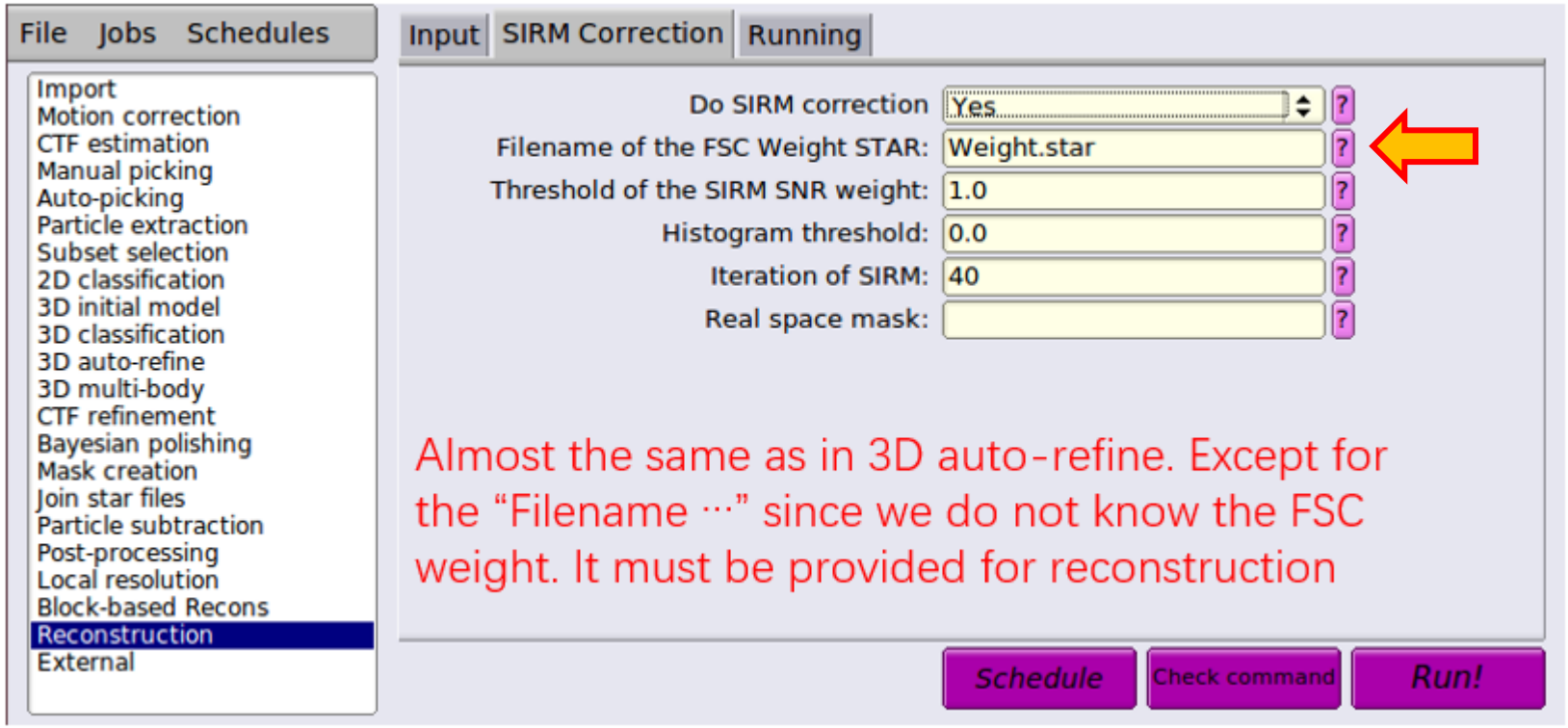
Then use SIRM-RELION GUI.

6、Final reconstruction



From our experience, Separately perform SIRM to the odd/even half could potentially leading to FSC raising problem beyond resolution. As a result, we alternatively to choose reconstructing all the particles, and applying FSC weight, b-factor sharpening and low-pass filter manually.

6、Final reconstruction



6、Final reconstruction

After the reconstruction, use the relion_image_handler from SIRM-RELION to apply the filters.

```
====== Post-process like b-factor and FSC weighting. Should be used together with --angpix =====
             --postprocess (false) : Do postprocess for single map.
                    --post_fsc () : FSC to be applied.
                        --lp (-1) : Low-passed frequency
                        --bf (0.) : B-factor to be applied
```

The command is:

`relion_image_handler --i reconstrct_SIRM_full.mrc –o reconstrct_SIRM_full_lp3p6_bfM60.mrc --angpix 1.42 --postprocess --post_fsc weight.star --lp 3.6 --bf -60`

The post_fsc is the exact FSC from the “convert_star_2_SIRM_weight.py”.