Auto3dem user's guide (v4.0)

This document describes how to use the auto3dem image reconstruction software. At this point the documentation should still be considered a work in progress, but additional content should be added soon.

Table of Contents

Introduction

Auto3dem is an automated system for 3D structure determination from images of vitrified particles. The software was specifically developed for icosahedral viruses, but should be able to handle particles of any symmetry as long as the particles are roughly spherical and their images can readily be divided into circular annuli. Even

this restriction can be relaxed if initial orientations for the images are available and auto3dem is run in refine mode.

The software can be run in either serial or parallel mode, but for best performance the latter is preferred. The underlying programs called by auto3dem have all been parallelized using MPI and you will need to have an MPI implementation (e.g. MPICH or OpenMPI) installed on your compute hardware in order to run in parallel mode.

If a starting model is not available, one can be constructed with the aid of the script setup_rmc that is distributed with the software. Setup_rmc creates the input files needed to construct multiple random models, runs each for a specified number of iterations, and compares the results to identify what is likely the best starting map.

To reference auto3dem, please cite:

> Yan X., R.S. Sinkovits, and T.S. Baker (2007) AUTO3DEM: an automated and high throughput program for image reconstruction of icosahedral particles. J. Struct. Bio. 157: 73-82.

To reference the random model method, please cite:

> Yan X., K.A. Dryden, J. Tang, and T.S. Baker (2007) Ab initio random model method facilitates 3D reconstruction of icosahedral particles. J. Struct. Bio. 157: 211-225.

Running auto3dem

Launching a new run

Auto3dem is launched from the Unix/Linux command line or from within a batch script using the following syntax

```
auto3dem -ncpu ncpu -input infile [-nodefile nfile]
```

ncpu = number of CPUs (can also use -np)
infile = auto3dem input file
nfile = name of file containing list of nodes (optional)

The node file does not generally need to be specified. This option is provided for those users who wish to run on a specific set of nodes. It is also used internally for jobs run on batch systems that use the PBS scheduler.

If launching auto3dem from the command line, you will most likely wish to run in the background by appending an ampersand (&) to the end of the line. When running on the same machine where the job was submitted, the CPU and memory usage can be monitored using the Linux top command.

Other than the number of CPUs, all parameters that control the behavior of auto3dem are set in the input file. The majority of the rest of this document focuses on constructing this file.

Data files and directory structure

Auto3dem does not require the data files to be in any particular location, but we find it most convenient to place the particle parameter files and boxed image files in directories named dat and pif, respectively. An example directory structure is given below.

```
Virus/
        dat/
                file1.dat_000
                file2.dat_000
        pif/
                file1_box.pif
                file1_box.pif
```

To maximize flexibility and avoid having to edit the particle parameter files when calculations are run in different locations, we suggest using relative path names to specify the locations of the corresponding boxed images. For example, the first lines of the parameter files would contain the lines

```
../pif/file1_box.pif
../pif/file2_box.pif
```

Summary file

Auto3dem writes out a summary file that keeps track of important reconstruction parameters as a function of iteration number. Example output is shown below

```
------ AUTO3DEM version v3.15 (parallel) ------

itr:          iteration number
              0 = starting map constructed from images
mode:         refinement mode s=search r=refine
              search mode only followed by (bin factor)
estres:       estimated resolution of model
              * = FSC never < 0.5
delta angle:  spacing between projections (degrees)
map undamp:   resolution to which map is computed
map damp:     resolution at which map density damped to zero
time:         wallclock time for iteration
cpu:          number of CPUs (MPI processes)
nptles:       number ptles used to construct map
ntot:         total number ptles in parameter files
nmg:          number of micrographs
defocus       defocus range (microns)

All resolutions expressed in Angstroms, times in seconds
```

```
                 delta    map     map
itr mode estres  angle undamp   damp    time cpu nptles  ntot  nmg  defocus
--- ---- ------ ------ ------ ------ ----- --- ------ ------ --- ---------
  1 s(2) 25.00*   0.50  16.67  14.29    32   8   3329   3329  68 0.90-3.47
  2 s(2) 20.35    0.50  14.99  13.04    34   8   3329   3329  68 0.90-3.47
  3 s(2) 17.71    0.50  15.05  13.08    33   8   3329   3329  68 0.90-3.47
  4 s(2) 20.27    0.50  15.87  13.69    33   8   3329   3329  68 0.90-3.47
  5 s(1) 20.14    0.50  15.93  13.74    91   8   3329   3329  68 0.90-3.47
  6    r 17.57    0.25  12.08  10.78   145   8   3329   3329  68 0.90-3.47
  7    r 13.59    0.25   9.73   8.87   116   8   3329   3329  68 0.90-3.47
```

When a new run is launched, a new summary file will be generated with a name based on the current directory (e.g. Virus_summary). If a summary file already exists, it will be copied to a new location (e.g. Virus_summary_backup). For continuation runs, new results will be appended to the existing summary file.

Restarting auto3dem

Auto3dem writes two restart files for each iteration. The first is written after the new particle origins and orientations are calculated and the second after the new map is generated. The names of the restart files are derived from the name of the directory where auto3dem was launched and are labeled with both the iteration number and either the letter a or b to indicate the first or second restart file for the iteration. For example:

Virus_restart_1a
Virus restart_1b

The restart files will contain all of the keyword parameters. Default values are supplied as necessary and other parameters are updated to reflect the progress of the reconstruction (e.g. improvements in resolution, naming of particle parameter files to reflect iteration number, etc.)

If auto3dem runs to completion, the restart files are moved to a directory that is automatically created and named using the name of the directory where auto3dem was launched (e.g. Virus_RESTARTS). A continue file is also generated in the run directory (e.g. Virus_continue). The only difference between the continue file and the restart files is that the former will have the number of iterations reset to whatever was used in the initial auto3dem parameter file whereas the latter will be configured to complete the original run.

To restart or continue a run, just launch auto3dem using the appropriate input file. For example:

auto3dem -ncpu ncpu -input Virus_continue
auto3dem -ncpu ncpu -input Virus_restart_5b

Constructing a starting model

Note – starting with version 4.0, a new and more reliable algorithm is used for the random model calculations. To revert to the previous version, use the –trad tag when running setup_rmc.

When a starting model is not available, one can easily be constructed for icosahedral viruses using the random model method. By default, setup_rmc will interrogate the particle parameter files in the dat directory and setup the directories and scripts required to generate ten random models using 150 images from the furthest from focus micrographs. It will also construct a minimal auto3dem input file that can be used to launch the full reconstruction. Executing the resulting RMC_run script will launch the random model calculations and RMC_cleanup will remove all intermediate files.

The number of random models to construct, the number of images to use, and other parameters can be controlled through command line arguments to setup_rmc. Running setup_rmc without any arguments provides a complete list of options. For example, to create 5 random models using 200 images

setup_rmc -nmodels 5 -nimages 200

If a good model is found using the criterion that the FSC never drops below 0.5 for all spatial frequencies less than 1/25 Å, the calculations are terminated and the current model is designated as the starting model. Otherwise, all calculations are allowed to run to completion and the best model is identified. In either case, the starting model is named rmc.pif and copied to the directory containing the particle parameter files (default = dat). At the end of the calculations, all intermediate files and directories are also moved into a new directory named RMC_temp

A sample session is shown below

```
% ls
dat pif

% setup_rmc
[output not shown - enter yes at prompt]

% ls
dat            RMC2_master  RMC6_master  RMC_cleanup
pif            RMC3_master  RMC7_master  RMC_logfile_list
RMC10_master   RMC4_master  RMC8_master  RMC_run
RMC1_master    RMC5_master  RMC9_master  Virus_master

% ls dat
file1.dat_000  file4.dat_000  file7.dat_000  RMC10  RMC4  RMC7
file2.dat_000  file5.dat_000  file8.dat_000  RMC2   RMC5  RMC8
file3.dat_000  file6.dat_000  RMC1           RMC3   RMC6  RMC9
```

```
% ./RMC_run

% ls
dat  pif  RMC_cleanup  RMC_temp  Virus_master

% ls dat
file1.dat_000  file3.dat_000  file5.dat_000  file7.dat_000  rmc.pif
file2.dat_000  file4.dat_000  file6.dat_000  file8.dat_000

% ls RMC_temp
RMC1             RMC2_RESTARTS  RMC5_log       RMC8
RMC10            RMC2_summary   RMC5_master    RMC8_continue
RMC10_continue   RMC3           RMC5_RESTARTS  RMC8_log
RMC10_log        RMC3_continue  RMC5_summary   RMC8_master
RMC10_master     RMC3_log       RMC6           RMC8_RESTARTS
RMC10_RESTARTS   RMC3_master    RMC6_continue  RMC8_summary
RMC10_summary    RMC3_RESTARTS  RMC6_log       RMC9
RMC1_continue    RMC3_summary   RMC6_master    RMC9_continue
RMC1_log         RMC4           RMC6_RESTARTS  RMC9_log
RMC1_master      RMC4_continue  RMC6_summary   RMC9_master
RMC1_RESTARTS    RMC4_log       RMC7           RMC9_RESTARTS
RMC1_summary     RMC4_master    RMC7_continue  RMC9_summary
RMC2             RMC4_RESTARTS  RMC7_log       RMC_bestmap_summary
RMC2_continue    RMC4_summary   RMC7_master    RMC_logfile_list
RMC2_log         RMC5           RMC7_RESTARTS  RMC_run
RMC2_master      RMC5_continue  RMC7_summary

% ./RMC_cleanup

%ls
dat  pif  Virus_master
```

Auto3dem input file

Auto3dem uses a single keyword-based file for specifying the auto3dem input. We describe below the general rules for the input file and the allowed format for the two types of records: auto3dem control parameter records and data records.

General formatting rules

The following rules apply to the entire input file.

- The ordering of lines is unimportant. However, if a keyword is specified more than once, the value associated with the later occurrence will override that for the earlier occurrence.
- Extra whitespace is ignored. Leading/trailing whitespace and blank lines are ignored. Contiguous blocks of whitespace are treated the same as a single space. Embedded whitespace is not allowed in character string input. For example, 'data file 1.pif' would not be a valid string.
- The hash/number (#) sign indicates start of comment. Entire lines can be commented out, or comments can be added to the end of a line. Hash signs are not allowed in character strings.

- Extra fields are ignored. Input parameters are specified using three fields, while data files only use two fields. Any data appearing after the end of the last required field is ignored. The one exception to this rule is that the email recipient field may consist of multiple addresses separated by whitespace and/or commas.
- Fields are case insensitive, except for character string input. Identifiers and keywords are internally converted to lowercase. Character strings specifying directories, file names, and binaries must be typed using the correct case.

Auto3dem control parameter records

These lines control the overall behavior of both auto3dem and the underlying image reconstruction codes (P3DR, PCTFR, PCUT, PO2R, PPFT, and PSF) that are called by auto3dem. With the exception of the email recipient line, which may contain an arbitrary number of recipients, all records have a three-field format

```
identifier key value
```

The first field (identifier) is used to distinguish whether the record contains an auto3dem control parameter or input for one of the image processing programs. The second and third fields form key-value pairs corresponding to the name of the input parameter and its values. The following case-insensitive values are allowed:

auto - auto3dem control parameter
p3dr - P3DR input parameter
pctfr - PCTFR input parameter
pcut - PCUT input parameter
po2r - PO2R input parameter
por - PO2R input parameter (provided for back compatibility)
ppft - PPFT input parameter
psf - PSF input parameter

Data records

All data files are specified using a two-field format.

```
data filename
```

The first field in the record must be the keyword data. The data lines can appear anywhere, but as a matter of convenience they are normally located at the end of the file. The file names are case sensitive. It is not necessary to provide full paths to the data files since the directory containing these files is specified using an auto3dem control parameter.

Minimal required input

Default values can be used for the majority of the auto3dem input parameters, but some values must still be supplied. The listing below shows an example minimal input file.

```
auto  mode        search      # Search mode (using ppft)
auto  niter       10          # 10 iterations
auto  start_map   start.pif   # Starting map
p3dr  res_min     8.5         # Resolution to which map is computed
data  file1.dat_000           # At least one data file required
```

Full input

When launching a new run, it is typically easiest to start with a minimal input file. Restart and continuation files will contain all keywords and can easily be edited.

Better Performance

Auto3dem can make a reasonable estimate for the inner and outer diameters of the capsid. If these radii are known (e.g. from inspecting central section of starting map), higher resolution can sometimes be achieved by specifying the following parameters

```
auto_freeze_annulus  1 # Keep annulus_low/high, in_rad/out_rad fixed
ppft annulus_low     n # inner radius of capsid (incl protrusions)
ppft annulus_high    n # outer radius of capsid (incl protrusions)
pcut in_rad          n # inner radius of capsid (excl protrusions)
pcut out_rad         n # outer radius of capsid (excl protrusions)
```

If the images have a small pixel size, can often use the following combination of parameters to start search mode calculations using binned image data

```
auto bin_reduce      1 # Automatically reduce bin_factor
ppft bin_factor      2 # Start with 2x2 binning of images
```

Note – as of version 4.0, it is no longer necessary to specify "ppft verbose -1" since this is now the default behavior for PPFT.

Full listing of auto3dem keywords

auto

For the sake of clarity, the AUTO3DEM input parameters are divided into two sets. The first set contains the general parameters, while the second contains the parameters related to particle selection criteria.

general parameters

bin_reduce: controls whether or not PPFT bin_factor should be reduced when resolution of reconstruction fails to improve.
boxrad: radius (pixels) of image box
delete_maps: non-zero value specifies that the maps generated at intermediate stages of the reconstruction should be deleted.
estimate_res: non-zero value indicates that resolution estimation is performed.

freeze_annulus: non-zero value freezes inner and outer radii of the annulus defining the ordered region of the map. This affects the parameters annulus_low and annulus_high in PPFT and in_rad and out_rad in PCUT.
freeze_res: non-zero value freezes the resolutions used in PSF, P3DR, PPFT, and PO2R.
fsc_hithresh: cutoff value for FSC used in estimating map resolution.
fsc_lothresh: cutoff value for FSC used to set resolution limits in P3DR, PSF, and PO2R.
gauss_adj: parameter used to set width of Gaussian falloff in P3DR.
have_map: non-zero value indicates that starting map is available.
hollow_cut_step: number of steps used by masking algorithm when generating a hollow map
hollow_cut_weight: weight used by masking algorithm when generating a hollow map
hollow_in_rad: inner radius of hollowed map
hollow_map: flag specifying whether or not map should be hollowed
hollow_out_rad: outer radius of hollowed map
iter_start: starting iteration
mode: AUTO3DEM mode of operation. Allowed values = (search, refine).
new_ptles: flag specifying whether or not new particles should be oriented relative to existing map without updating the map
niter: maximum number of iterations of AUTO3DEM main loop
noctf: if true, disables CTF correction. Overrides CTF mode and sets to zero for programs P3DR, PO2R, PPFT, and PCTFR. Used primarily with image sets which have already been CTF corrected.
noise_suppression: apply Rosenthal and Henderson JMB 333 721-745 (2003) noise suppression algorithm
outfile: base name used to construct names of log, summary, restart, and continuation files
per_ptle_ctf: apply CTF correction on a per-particle basis in P3DR, PO2R, and PPFT. Setting to one overrides per_ptle_ctf parameter set for individual programs.
quit_early: set to non-zero value to have AUTO3DEM quit if the FSC curve never drops below fsc_hithresh (usual value is fsc_hithresh=0.5). This option is normally only used for random model calculations where it is set automatically by setup_rmc.pl
refine_ctf: refine CTF parameters at end of each iteration when running in refine mode
res_adj: parameter that determines the higher resolution to which map will be calculated beyond upper resolution limit used in PO2R.
restart: set to 1 to continue calculation
rundir: directory containing input data (maps, images, particle parameters)
start_map: name of starting map used by AUTO3DEM.
switch_mode: if true, allows auto3dem to automatically switch from search to refine mode
symm_code: symmetry code

term_refine: allow automatic termination of run when in refine mode (functionality not currently active, added as placeholder)
term_search: allow automatic termination of run when in search mode (functionality not currently active, added as placeholder)

particle selection parameters

box_center_offset: maximum allowable distance between the center of the particle and the center of the box; applied separately to each coordinate. Particles with centers too far from center of box are excluded from the model.
cmp_cc_fraction: fraction of images to accept on the basis of the CMP correlation coefficient. Makes sense only when parsing particle parameter files generated in search mode.
cmp_cc_nstd: number of standard deviations to add to the average CMP correlation coefficient when setting cutoff. Negative values are less restrictive, positive values are more restrictive.
global_select: if set to true (non-zero) value, then selection criteria are applied globally across particle parameter files. Otherwise, selection criteria applied on a per file basis
nselect_offset: number of selection criteria to evaluate in each 'direction' from the central selection criterion. The total number of selection criteria to be evaluated is (2*nselect_offset + 1).
omega1, omega1_tol: select images with omega within omega1_tol of omega1. Must be used together
omega2, omega2_tol: select images with omega within omega2_tol of omega2. Must be used together
pft_cc_fraction: fraction of images to accept on the basis of the PFT correlation coefficient. Makes sense only when parsing particle parameter files generated in search mode.
pft_cc_nstd: number of standard deviations to add to the average PFT correlation coefficient when setting cutoff. Negative values are less restrictive, positive values are more restrictive.
phi_reject_lower / phi_reject_upper: range of azimuthal angles (phi_reject_lower ≤ phi ≤ phi_reject_upper) for which images will be excluded from map construction.
prj_cc_fraction: fraction of images to accept on the basis of the PRJ correlation coefficient. Makes sense only when parsing particle parameter files generated in search mode.
prj_cc_nstd: number of standard deviations to add to the average PRJ correlation coefficient when setting cutoff. Negative values are less restrictive, positive values are more restrictive.
score_fraction: fraction of images to accept on the basis of the score generated by program PO2R.
score_nstd: number of standard deviations to add to the average score when setting cutoff. Negative values are less restrictive, positive values are more restrictive.

select_delta: the size of the 'step' to be used when evaluating multiple selection criteria. For standard deviation-based criteria, adds a fixed number of standard deviations; for fraction-based criteria, adds a fixed fraction.
theta_reject_lower / theta_reject_upper: range of inclination angles (theta_reject_lower ≤ theta ≤ theta_reject_upper) for which images will be excluded from map construction.

p3dr

apo_border: width of border region for map apodization
bin: name of P3DR binary
ctf_ff1: 1st CTF filter factor
ctf_ff2: 2nd CTF filter factor
ctfmode: CTF mode (ctf_mode also accepted)
fsc_file_name: name of FSC file to be used when applying noise suppression algorithm. (File format: line 1 = number of FSC records; subsequent lines = spatial frequency (Å-1) FSC value)
filter: filter mode
magfactor: magnification factor
map_dim: map dimension
max_cpu: maximum number of CPUs to be used by P3DR
per_ptle_ctf: apply CTF correction on a per-particle basis
res_max: resolution at end of Gaussian falloff
res_min: resolution to which map is computed
symm_code: symmetry code
tempfac: temperature factor
zero_fill: zero fill for background pixels

pctfr

bin: name of PCTFR binary
ctf_ff1: 1st CTF filter factor
ctf_ff2: 2nd CTF filter factor
ctfmode: CTF mode (ctf_mode also accepted)
dangle: CTF parameter step size
filter: filter mode
funcmode: function mode
funcweight: function weight
max_cpu: maximum number of CPUs to be used by PCTFR
nangle: number of steps in CTF parameters taken in each direction
res_max: maximum resolution used in image/projection comparison
res_min: minimum resolution used in image/projection comparison
tempfac: temperature factor
zero_fill: zero fill for background pixels

pcut

bin: name of PCUT binary
cut_step: number of steps used by masking algorithm

cut_weight: weight used in masking algorithm
in_rad: inner radius for masking
max_cpu: maximum number of CPUs to be used by PCUT
out_rad: outer radius for masking

po2r

bin: name of PO2R binary
ctf_ff1: 1st CTF filter factor
ctf_ff2: 2nd CTF filter factor
ctfmode: CTF mode (ctf_mode also accepted)
dangle: angular step size (delta angle)
dcenter: spatial step size (delta xy)
filter: filter mode
funcmode: function mode
funcweight: function weight
global_por: enable/disable global orientation search
handtest: enable/disable handedness tests for images
magfstep*: size of magnification refinement step
max_cpu: maximum number of CPUs to be used by PO2R
nangle: number of angular steps taken in each direction
ncenter: number of spatial steps taken in each direction
nmagf*: number of magnification factor steps
per_ptle_ctf: apply CTF correction on a per-particle basis
quick_search: implement quick approximate search of orientation space
res_max: maximum resolution used in image/projection comparison
res_min: minimum resolution used in image/projection comparison
symm_code: symmetry code
tempfac: temperature factor
ticos_equiv: restrict search to the 60 symmetry related orientations
zero_fill: zero fill for background pixels

*Not settable through auto3dem. PO2R standalone feature

ppft

The PFTsearch/PPFT input parameters pftrads_filename, pftres1_filename, and pftres2_filename are not read from the input parameter file since they are set by AUTO3DEM. They are assigned the values ppft_iter_n.rads, ppft_iter_n.res1, and ppft_iter_n.res2, respectively, where n is the iteration number.

annulus_high: outer radius of annulus for image/projection comparison
annulus_low: inner radius of annulus for image/projection comparison
bin: name of PPFT binary
bin_factor: binning factor
ctf_mode: CTF mode (ctfmode also accepted)
delta_theta: step size for inclination angle theta
filter_factor_1: 1st filter factor

input_mode: input mode
jcut: minimum order Bessel function (Jn), default strongly recommended!
mag_cen: midpoint for magnification scale search
mag_norm: switch used to normalize the MAG scale factors so that the average MAG is 1.0.
mag_num: extent of magnification search window
mag_step: grid size of magnification scale search
max_cpu: maximum number of CPUs to be used by PPFT
per_ptle_ctf: apply CTF correction on a per-particle basis
pft_filename: name of PFT file
pftrad_hi: outer PFT radius
pftrad_lo: inner PFT radius
pftrad_step: PFT radius step size
prj_filename: file name for prj output
quick_omega: perform fast approximate search for omega
resolution_high: upper resolution limit
resolution_low: lower resolution limit
sigcut: threshold for variance mask when filtering PFT data, default strongly recommended!
symmetry: symmetry code
symm_code: alternative for specifying symmetry code
temperature_factor: temperature factor
verbose: verbose factor (controls level of output)

psf

The PSF input parameters pixel_size is not read from the input parameter file since it is obtained by parsing the first line of the particle parameter files.

bin: name of PSF binary
max_cpu: maximum number of CPUs to be used by PSF
res_max: maximum resolution used in FSC calculation
res_min: minimum resolution used in FSC calculation
res_step: resolution step size

Auto3dem keyword default values

The following tables list the auto3dem input parameters along with their default values. A missing default value means that no default is used. Required input shown in **bold red**.

auto	bin_reduce	0
auto	boxrad	Extracted from PIF1
auto	box_center_offset	10000
auto	cmp_cc_fraction	

auto	cmp_cc_nstd	
auto	delete_maps	1
auto	estimate_res	1
auto	freeze_annulus	0
auto	freeze_res	0
auto	fsc_hithresh	0.5
auto	fsc_lothresh	0.3
auto	gauss_adj	0.01
auto	global_select	1
auto	have_map	1
auto	hollow_cut_step	12
auto	hollow_cut_weight	0.001
auto	hollow_in_rad	
auto	hollow_map	0
auto	hollow_out_rad	
auto	iter_start	1
auto	**mode**	
auto	new_ptles	0
auto	**niter**	
auto	noctf	0
auto	noise_suppression	0
auto	nselect_offset	0
auto	omega1	0
auto	omega1_tol	360
auto	omega2	0
auto	omega2_tol	0
auto	outfile	Current working directory
auto	per_ptle_ctf	0
auto	pft_cc_fraction	
auto	pft_cc_nstd	
auto	phi_reject_lower	360
auto	phi_reject_upper	-360
auto	prj_cc_fraction	
auto	prj_cc_nstd	
auto	quit_early	0
auto	refine_ctf	0
auto	res_adj	0.01
auto	restart	0
auto	rundir	dat
auto	score_fraction	
auto	select_delta	0
auto	select_nstd	
auto	**start_map**	See note2
auto	switch_mode	1

auto	symm_code	
auto	term_refine	0
auto	term_search	0
auto	theta_reject_lower	360
auto	theta_reject_upper	-360

pcut	bin	PCUT
pcut	cut_step	12
pcut	cut_weight	0.001
pcut	in_rad	boxrad/3
pcut	max_cpu	8
pcut	out_rad	boxrad - 10

p3dr	apo_border	12
p3dr	bin	P3DR
p3dr	ctf_ff1	0.05
p3dr	ctf_ff2	0.1
p3dr	ctfmode	1
p3dr	fsc_file_name	
p3dr	filter	1
p3dr	magfactor	1.0
p3dr	map_dim	0
p3dr	max_cpu	256
p3dr	per_ptle_ctf	0
p3dr	res_max	(p3dr{res_min}-1 + auto{gauss_adj})-1 (See note3)
p3dr	**res_min**	
p3dr	symm_code	532
p3dr	tempfac	0
p3dr	zero_fill	1

pctfr	bin	PCTFR
pctfr	ctf_ff1	0.5
pctfr	ctf_ff2	0.1
pctfr	ctfmode	1
pctfr	dangle	0.05
pctfr	filter	1
pctfr	funcmode	3
pctfr	funcweight	0
pctfr	max_cpu	256
pctfr	nangle	4
pctfr	res_max	(p3dr{res_min}-1 + auto{gauss_adj})-1
pctfr	res_min	(2/5)·boxrad·pixel_size

pctfr	tempfac	0
pctfr	zero_fill	1

po2r	bin	POR
po2r	ctf_ff1	0.5
po2r	ctf_ff2	0.1
po2r	ctfmode	1
po2r	dangle	(1/2)·ppft{delta_theta}
po2r	dcenter	0.1
po2r	filter	1
po2r	funcmode	1
po2r	funcweight	0
po2r	global_por	0
po2r	handtest	1
po2r	max_cpu	256
po2r	nangle	4
po2r	ncenter	4
po2r	per_ptle_ctf	0
po2r	quick_search	1
po2r	res_max	(p3dr{res_min}-1 + auto{gauss_adj})-1
po2r	res_min	(2/5)·boxrad·pixel_size
po2r	symm_code	532
po2r	tempfac	0
po2r	ticos_equiv	0
po2r	zero_fill	1

ppft	annulus_high	boxrad/3
ppft	annulus_low	boxrad - 10
ppft	bin	PPFT
ppft	bin_factor	1
ppft	ctf_mode	3
ppft	delta_theta	0.5
ppft	filter_factor_1	0.1
ppft	input_mode	2
ppft	jcut	1
ppft	mag_cen	1
ppft	mag_norm	1
ppft	mag_num	1
ppft	mag_step	0
ppft	max_cpu	256
ppft	per_ptle_ctf	0
ppft	pft_filename	pft.pfts
ppft	pftrad_hi	auto{boxrad}
ppft	pftrad_lo	1.0

ppft	pftrad_step	1
ppft	prj_filename	pft.prjs
ppft	quick_omega	1
ppft	resolution_high	(p3dr{res_min}-1 + auto{gauss_adj})-1
ppft	resolution_low	(2/5)·boxrad·pixel_size
ppft	sigcut	0
ppft	symmetry	532
ppft	symm_code	532
ppft	temperature_factor	0
ppft	verbose	2

psf	bin	PSF
psf	max_cpu	8
psf	res_max	p3dr{res_min}
psf	res_min	60
psf	res_step	50

[1] Boxrad is normally extracted from the PIF file header field packRadius. The option to specify boxrad in the auto3dem parameter file is provided for use in those cases where this information is missing from the PIF header.

[2] If the have_map flag is false, then starting map is not required.

[3] Subject to Nyquist condition that resolution is not less than twice the pixel size.