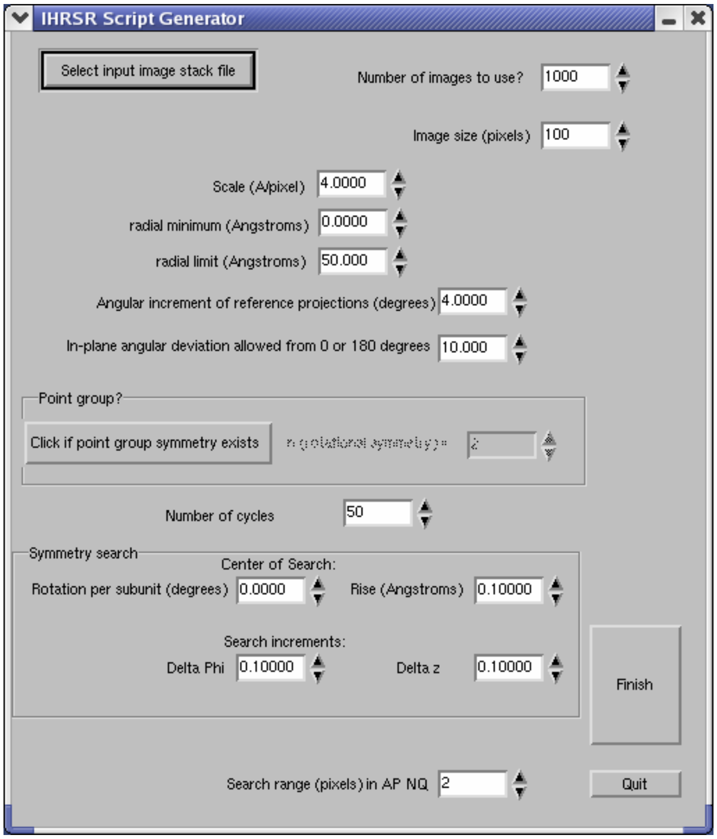
IHRSR Programs


IHRSR Script Generator
Select input image stack file
Point group?
Click if point group symmetry exists
Symmetry search
Finish
Quit

Number of images to use? 1000

Image size (pixels) 100

Scale (A/pixel) 4.0000

radial minimum (Angstroms) 0.0000

radial limit (Angstroms) 50.000

Angular increment of reference projections (degrees) 4.0000

In-plane angular deviation allowed from 0 or 180 degrees 10.000

n (rotational symmetry) = 2

Number of cycles 50

Center of Search:

Rotation per subunit (degrees) 0.0000

Rise (Angstroms) 0.10000

Search increments:

Delta Phi 0.10000

Delta z 0.10000

Search range (pixels) in AP NQ 2

generator – graphical interface for generating the IHRSR script

This program (generator) is invoked in the directory where you will want the IHRSR script written. The output file created by this program is named b25.spi, and is a SPIDER script. *This program requires that the file script1 be placed in /usr/local/bin.* Here is an explanation for the requested parameters:

Select input image stack file – this is self-evident, and provides the script with the name and path for the raw images that will be used for the IHRSR procedure. As discussed below, these filament segments need to be pre-centered (in the x-direction) to make the subsequent processing efficient. This can be done by alignment against some initial reference, using only integer shifts in x to move them. The integer shifts avoid any interpolation that will potentially degrade resolution. A SPIDER script showing such pre-alignment (b03.spi) is included in the IHRSR distribution.

Number of images to use? – this specifies the number of images for processing, which can be less than the stack size.

Image size (pixels) – Images must be square, and this specifies nx and ny (the number of pixels in x and y, respectively).

Scale (A/pixel) – self-evident

radial minimum (Angstroms) – this specifies the inner radius for the helical search and imposition. In general, this will be 0.0, except for hollow tubes, where it will have a value greater than this.

radial limit (Angstroms) – this specifies the outer radius of the filament for the helical search and imposition.

Angular increment of reference projections (degrees) – this value is determined from the geometry of a single-axis tilt series, where the number of equally spaced projections needed for an object of diameter D to obtain a resolution of d is πD/d (Crowther, DeRosier and Klug, 1970). The minimum increment (in degrees) is therefore (180*d)/(πD), but one would typically oversample by ~ a factor of 2.

In-plane angular deviation allowed from 0 or 180 degrees - this provides a means of excluding “bad” images during the subsequent processing by taking advantage of the fact that the initial segments were windowed out of longer boxes in which the filaments are approximately vertical. If all of the segments were perfectly vertical, then the only values found for in-plane rotation would be 0° or 180° (depending upon whether a segment has the same or opposite polarity, respectively, as the reference), plus random values (anywhere between 0° and 360°) for segments that cannot be properly aligned against a reference. One could thus exclude all segments having rotations that are not exactly 0° or 180° to get rid of these bad segments. However, given curvature and flexibility, some real deviations from 0° or 180° will always be found for good segments. But a segment found with 90° rotation with respect to the reference is certainly one that should be excluded. The greater the degree of curvature, the larger this parameter needs to be so as to not exclude good segments. A reasonable value might be 10-20°.

Click if point group symmetry exists – The generation of a helical filament involves a screw symmetry (a coupled angular rotation and axial translation), as well as a possible point group symmetry, which would be an n-fold rotation about the filament axis. If the structure can be completely described by a rotation and translation of an asymmetric unit along a 1-start helix, this item does not need to be set.

Number of cycles – this is simply the number of iterations to run the IHRSR script. The minimum value is 2 (one cycles), since the convention used is that cycle *n* begins by using the volume created in cycle *n-1* as a reference. Thus, the script created here (b25.spi) will require that there be a reference volume named volume001.dat (or whatever extension you intend to use for data files) in the current directory.

Rotation per subunit (degrees) – this will be the initial value for the angular component of the helical symmetry, and will be refined over the course of the iterations. This value should correspond to the symmetry of the initial reference volume, volume001.

Rise (Ångstroms) - this will be the initial value for the axial translation component of the helical symmetry, and will be refined over the course of the iterations. This value should correspond to the symmetry of the initial reference volume, fzzsksk001.

Delta phi – this is the increment (in degrees) that is used when searching about the rotation per subunit value entered above. The search is made over a range of -10 to +10 times the search increment.

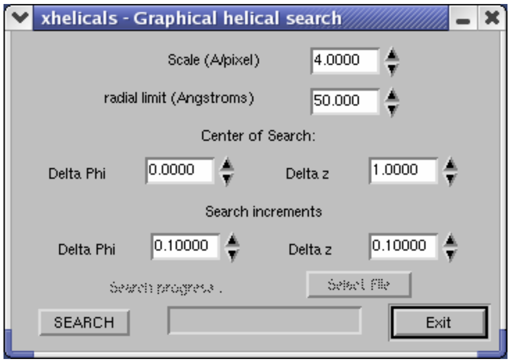
Delta z - this is the increment (in Å) that is used when searching about the rise per subunit value entered above. The search is made over a range of -10 to +10 times the search increment.

Search range (pixels) in AP NQ – The SPIDER routine AP NQ is used for the multi-reference alignment of the raw images against the different projections of the reference volume, and this routine accounts for most of the time taken by each iteration. The time required for AP NQ is quadratic in the size of the search range. Thus, having a search range of 6 pixels will require a ~ 9-fold increase in time compared to having a search range of 2 pixels. The minimum size of this parameter is dictated by the axial rise per subunit (in pixels), so that any image segment may be properly aligned to a reference. For example, if the axial rise per subunit is 15.0 Å, and a sampling of 3.9 Å per pixel is used in the images, then the furthest any image segment can be (in the y-direction) from a reference projection is 3.85/2 pixels, and a search range of 2 pixels is adequate.

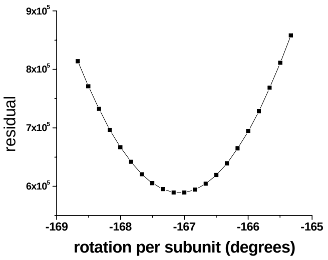
X-window interactive programs:

xhelicals

This is an interactive version of hsearch, and might be used during the first cycle to ensure that the correct symmetry is found. The input_volume is chosen interactively, as are the parameters for the search. This program will graphically display the residual plots as a function of both azimuthal rotation and axial rise. There should be a clear minimum found, as in Ultramicroscopy **85**, 225-234 (2000), or in the plot on the right. The difference in the residual between the minimum and maximum values should be significant, as a very shallow minimum might be the cause for some concern.

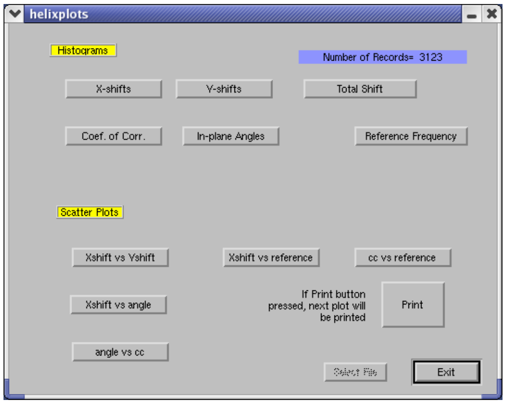


Scale (A/pixel) 4.0000
radial limit (Angstroms) 50.000
Center of Search:
Delta Phi 0.0000 Delta z 1.0000
Search increments
Delta Phi 0.10000 Delta z 0.10000



hplot

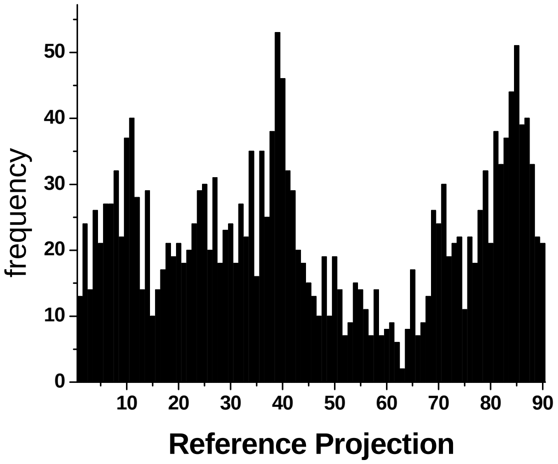
This program provides a graphical display for all of the parameters found in the AP NQ search. It is crucial that this program be run to examine these parameters, as they are very rich in information that can point to artefactual results. The x-shift plot should be centered at 0.0 (showing that the filament segments are centered), and it should fall off before the limit of the search. That is, if a search range of 2 pixels is given in AP NQ, most of the x-shift values should be between -1.5 and +1.5. There should be no correlation between x-shift and in-plane rotation, as such a corelation can reveal miscentered segments (a filament might be rotated 180° in order to better center it). Similarly, there should be no correlation between reference frequency and any other parameter. The in-plane rotation angles should be near either 0 or 180.

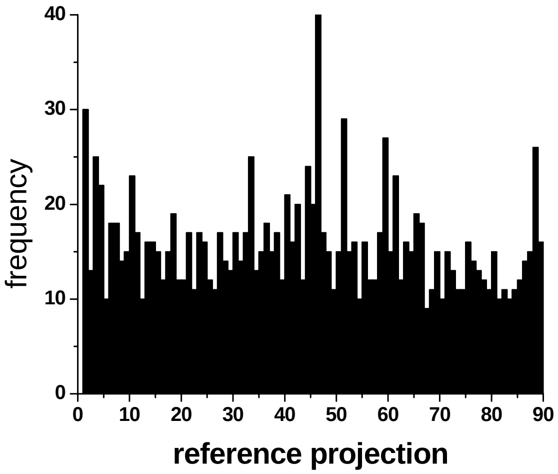


Number of Records= 3123

If Print button pressed, next plot will be printed

The distribution of reference frequencies should be flat (within statistical fluctuations), as distributions that are considerably different from this (with certain references never seen) can show artefactual results. In particular, if a wrong symmetry has been used, one can get a stable solution, but the reference distribution will not be flat. But heterogeneity can also lead to a stable solution, with the same type of uneven frequency of reference projections that are seen when the wrong symmetry is used for an initial model. To show this, we have intentionally mixed images of filament segments from UvsX-DNA complexes that were previously separated into two different groups: one with a pitch of 66 Å and 6.4 units/turn, and the other with a pitch of 94 Å and 6.1 units/turn. These images converge to a stable solution with ~ 58.8° per subunit (6.12

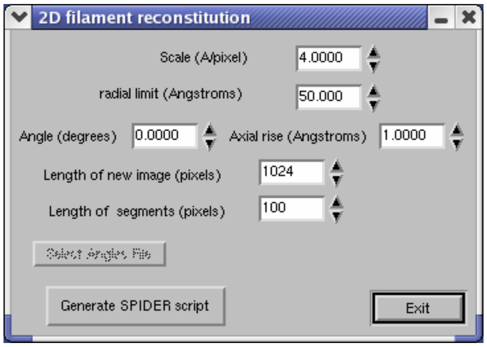




units/turn), and the pitch of this solution is 90 Å. A reconstruction from this solution looks quite reasonable. However, an inspection of the frequency distribution for reference projections (above, left) immediately shows us that something is wrong, since it is far from flat. For comparison, in a “good” distribution taken from an actin-cofilin data set (above, right), most of the fluctuations can be explained by a random statistical model.

filrecon

This program is for **fil**ament **recon**stitution and is used to generate a 2D averaged image of a filament. It requires the angles file (used as input for BP 3F) and the helical symmetry (azimuthal rotation and axial rise). It will ask for the length (in pixels) of the reconstituted filament image, and the length of the segments. From all of this, it will generate two histograms, one showing the frequency of segments along the reconstituted image, and the other showing the frequency of deviations (in degrees) from the ideal helical geometry if segments are placed into a reconstituted image of the chosen length. If one is using 4° angular increments for the reference projections in IHRSR, then there is an initial uncertainty of up to 2° in all of the angular assignments. That is, in principle an image segment cannot be more than 2° away from the azimuthal orientation that has been assigned to it. If one reconstitutes these segments into a very short image (say 100 pixels), then there may be a very large error in angles, but there will be many segments overlapping. If one chooses a very long image, there will be less error in angular position, but there may be very few overlapping segments.



2D filament reconstitution

Scale (A/pixel) 4.0000

radial limit (Angstroms) 50.000

Angle (degrees) 0.0000 Axial rise (Angstroms) 1.0000

Length of new image (pixels) 1024

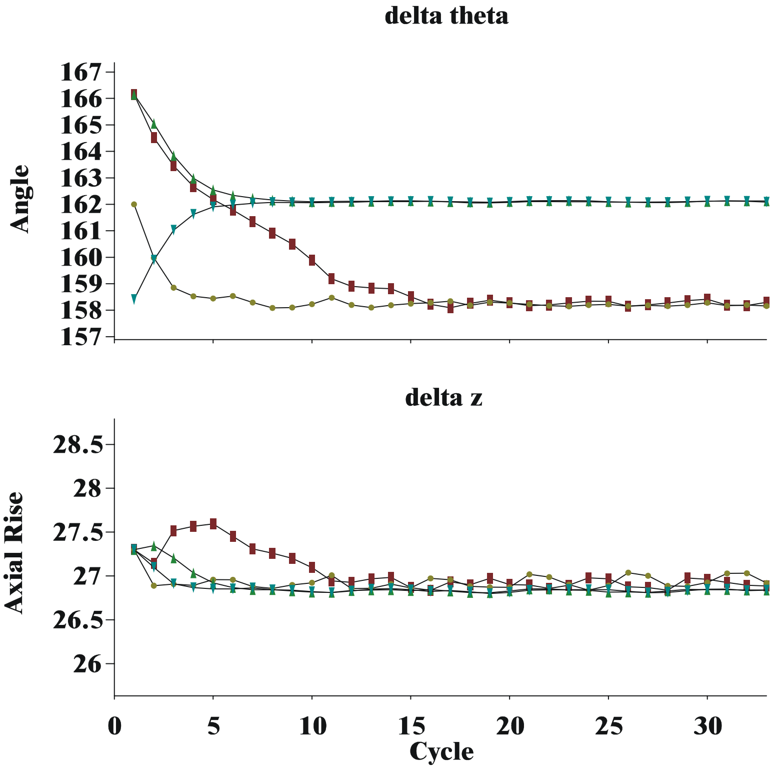
Length of segments (pixels) 100

The output of filrecon is a SPIDER script which will ask for the name of the image stack, and a SPIDER document file zshift.dat that will be read by the SPIDER script. *The file filrecon.txt must be placed in /usr/local/bin for this program to work.*

cplot

This is a contour plotting program that will display sections. It is assumed that the input file will be the symmetrized volume output by helicali, where the sections are perpendicular to the filament axis. This program will also print these plots, using the linux ‘lp’ command to send a postscript file to the printer.

splot



This is a program for displaying the “symdoc” file. It also allows for displaying multiple symdoc files, which is very useful for comparisons. For example, one needs to ensure that the same symmetry is reached after starting from different initial values. This is an important and crucial test of the result. The failure to find this is usually due to heterogeneity in the data. The examples shown (right), for F-actin, illustrate such a stable convergence for two different sets of filament segments, a “158°” subset and a “162°” subset. The plot shows that from two different starting points the 158° subset of images converge to the same solution, and for two different starting points the 162° subset converges. We have seen many examples where this convergence does not occur (different solutions are reached from different starting points), and this can be explained in most cases by the presence of heterogeneity.

This program will also display the helical net, computed from the last value in the symdoc file.

surface

This program requires SceneViewer to be installed on the system. SceneViewer can be obtained freely from SGI as part of its OpenGL package.The program displays a surface as a function of a threshold value. The appropriate threshold value can be determined by entering the expected molecular weight to be enclosed within a given number of sections. It assumes a partial specific volume for protein of 0.75 cm^3/g. The program will also allow for a WaveFront object file (extension .obj) to be generated. This is an ASCII format file that can be imported into rendering programs such as NuGraf.

Command line routines (in the SPIDER script):

>hsearch_lorentz input_volume symdoc scale rad_min rad_max inc_phi inc_z

The program “hsearch” does the helical search. It will read the “symdoc” file and use the last record there to define the center point for a search in azimuthal rotation and axial rise that minimizes the residual between subunit densities in the input volume.

The input_volume is a SPIDER format 3D volume generated by BP 3F, in which the helical axis is parallel to the plane of the sections (that is, the helical axis will be within the section INT(nsec/2) +1, where nsec is the number of sections in the volume).

The “symdoc” file is a SPIDER format document file that will have a comment line at the top, and then have four columns: the key, the number 2 (for two data columns), the azimuthal rotation (in degrees), and the axial rise (in Angstroms). The convention is that a right-handed one-start helix has a positive azimuthal rotation, while a left-handed helix has a negative azimuthal rotation.

The scale is the sampling in Angstroms/pixel. It must be a floating point number!

The parameters rad_min and rad_max are in Angstroms, and determine the search radius. For a helix that is not a hollow tube, rad_min would be set to 0.0; for a hollow tube, having rad_min greater than 0.0 will speed up the search. They also must be floating point numbers!

The search is done over a range of +/- 10 steps in both phi (angle) and z (axial rise), about the last values found in the symdoc file. The step size is given by inc_phi (in degrees) and inc_z (in Angstroms).

>himpose input_volume symdoc output_volume scale rad_min rad_max

The program “himpose” does the helical imposition. The parameters are the same as for “hsearch”, except that an output_volume is also specified, and the search increments are not specified. The output volume is a SPIDER format file that will be generated by the program. The helical axis in this new volume will be perpendicular to the plane of the sections, and be located at position nx,ny = INT(nsam/2) +1, INT(nrow/2) +1.

>himpose_long input_volume symdoc output_volume scale rad_min rad_max nsec

This has the same parameters as himpose, but with an additional parameter (nsec) specifying the number of sections to generate in the output volume. If the normal volume generated by himpose was 80 x 80 x 80, specifying 500 for nsec would create an output volume of size 80 x 80 x 500.

General Points:

1) Segments should be extensively overlapped when cutting these from the filament images, as failure to do so will lose information. For a segment that is 100 pixels long, we might typically use a 10 pixel shift for each box, so that adjacent boxes have 90% overlap. As long as the search range in AP NQ is small enough, these adjacent segments will be assigned to different reference projections which will provide different views of the filament for the back projection.
2) Segments should be approximately centered in x by using a cross-correlation procedure against some reference model. The initial shifts to center should be done using NINT(xshift), where xshift is the shift needed. This will be an integer shift, avoiding any interpolation. Each cycle, there is a single interpolation (from RT SQ) to rotate and shift the segments in both x and y. Because one always goes to the original images to do this, there is no accumulated set of interpolations. Since one does not need to use a large search range (in AP NQ), the program can run much faster. The speed is ~ quadratic in the size of the search range, so halving it will lead to a 4-fold improvement in speed.
3) In general, the size of the search range must be larger than the axial rise per subunit. For example, if the axial rise (in pixels) is 4.5, one would need to use a search range of 3 pixels, so that the search will extend from -3 to +3 pixels. This will allow any segment to have a corresponding reference.