Instructions for running Doppler at the CfA

In the following instructions, computer output is in small type**, prompts end with ":" or ">", comments are in boldface and that which the user should type is in** *italics*:

In order to run Doppler, you will need an .sdf file created in molly from spectra trimmed around the line of interest. The process consists of creating a starting image, scaling it to the optimal starting image for the data, then iterating it with the data to produce a tomogram. The iteration employs the 'Minimum Entropy Method', by which it produces a series of image fits with a reduced chi-squared in relation to the data, then chooses the image with the least entropy, under the assumption that this is the most likely image to be correct.

Starting Doppler

To make Doppler commands available, source the aliases using this command in an xterm window:

> source /data/saku10/software/doppler/doppler_aliases

Creating a Starting Image

Doppler requires an initial image to be created using the command 'makimg', which is then modified to fit the data and produce a meaningful tomogram. Depending on where the line in questions is being emitted, it may help to start with a Gaussian instead of a flat image, by setting 'fancy' to 'TRUE', but in most cases a flat image works equally well. If you are creating tomograms for lines which are blended or very close, you may need to set the number of images and lines higher. The Parameters are explained below:

> *makimg*
NSIDE - Number of pixels/side /100/ > 200 **The length, in pixels, of each side of the square tomogram image**
NIMAGE - Number of images /2/ > **The number of images you want to produce, usually one for each line**
IMAGE - Image to create /@twobintest/ > *startimage* **The image filename**
VPIX - Pixel size in km/s /6/ > **The velocity resolution of the image per pixel – VPIX multiplied by NSIDE must be great enough to cover the whole velocity range of the line**
NWAVE - Number of lines /2/ > **The Number of lines in the input spectrum**
WAVE - Wavelength(s) of line(s) /4685/ > **The wavelength(s) of the line(s)**
1 more value is still needed.
WAVE - Wavelength(s) of line(s) /4685/ > 4641
FWHM - FWHM resolution (km/s) /16/ > **The velocity resolution of your spectra, or slightly lower**
GAMMA - Systemic velocity (km/s) /-140/ > **The radial velocity of the system**
NSUB - Phase blurr factor /1/ > **Accounts for phase smearing – set to 1 if exposures are short compared to the period**
OFFSET - Phase offset /0/ > **If the phases are offset from the ephemeris, use this for corrections**
NDIV - Pixel splitting factor /24/ > **The factor by which data pixels are divided before being binned onto the image array – must be larger than FWHM**
USE - Use HJDs or Phases? /'PHASES'/ > **Use Phases**
FANCY - Do you want to set fancy image? /FALSE/ > TRUE **Set this to TRUE if you want a Gaussian starting image -**
XCEN - X centre of gaussian (km/s) /0/ > **the describing parameters of the gaussian are self**

YCEN - Y centre of gaussian (km/s) /0/ > **explanatory**
WIDTH - FWHM of gaussian (km/s) /300/ >
PGAUSS - Peak of gaussian /3/ >
XSP - X centre of power law+spiral (km/s) /0/ > **Parameters relating to spiral arms, if present**
YSP - Y centre of power law+spiral (km/s) /0/ >
EXPON - Exponent for power law+spiral /0/ >
VLOW - Lower velocity limit (km/s) /0/ > **Minimum velocity of the line(s)**
VHIGH - Upper velocity limit (km/s) /1000/ > **Maximum velocity of the line(s)**
ANGLE - Opening angle of spiral (degrees) /2.5/ >
CURVE - Total curvature of spiral (deg) /120/ >
PPOWER - Peak of power law background /1/ >
PSPIRAL - Peak of spiral shocks /1/ >

The main parameters which require attention are FWHM and NDIV – if FWHM is too small, the image will be messy, whilst if NDIV is too large, bearing in mind it must be larger than FWHM, the program will run very slowly, or not at all.

Scaling the Starting Image

In order to save having to iterate the starting image many times just to rescale it before any actual changes can be made, the command 'optscl' is used. This *only* scales the image, to the optimal scaling to start iterations:

> *optscl*
IMAGE - Image to rescale /@startimage/ > **Filename of starting image**
DATA - Reference data file /@twodaysbin/ > **The spectrum data file produced by molly**
GLOBAL - Do you want global re-scaling ? /TRUE/ > **If set to TRUE, all images in the set will be scaled equally**
Initial reduced Chi-squared = 2.94335
Final reduced Chi-squared = 2.6425
Image 1 scaled by 9.92018
Image 2 scaled by 9.92018

Possible error: If you receive the error message "FAST buffer too small", this means that the combination of the size of the image you are trying to scale (detemined by NSIDE) and the chosen pixel splitting factor (NDIV) is too high for the programme to deal with. Reducing either should prevent this error.

Iterating the image

The scaled starting image is now ready to be iterated over the data, using the minimum entropy method iterating protocol 'memit':

> *memit*
IMAGE - Image to iterate /@startimage/ > **The starting image filename**
DATA - Reference data file /@twodaysbin/ > **The spectrum data file produced by molly**
CAIM - Reduced chi-squared to aim for /1/ >
NITS - Number of iterations /1/ >
RMAX - Maximum step/iteration /0.2/ >
NDEF - Default option /1/ > 2 **The type of iteration you wish to use**
BLURR - FWHM of gaussian blurr default /10/ > **The blur factor to use when employing NDEF = 2**

The iteration will then commence as below, editing the starting image and giving you the new chi-squared value:

```
Image pixels = 80000, data values = 16140

 FORMATTING CORE STORAGE FOR DATA AND IMAGE VECTORS.
 IMAGE DIMENSION (X,Y,Z)= 200 200 2
  DATA DIMENSION = 16140
 9  IMAGES IN 9  FILES
 1 80001 160001 240001 320001 400001 -9999999 -9999999 -9999999 -9999999 -9999999 -9999999 -9999999 -9999999
-9999999 -9999999 -9999999 640001 480001 560001
 8  DATA VECTORS IN 8  FILES
 720001 736141 752281 768421 784561 800701 816841 832981 -9999999 -9999999 -9999999 -9999999 -9999999
-9999999 -9999999 -9999999 -9999999 -9999999 -9999999 -9999999
 IMAGE STORE: 720000  = 84.79367  %
  DATA STORE: 129120  = 15.206331  %
 TOTAL STORE: 849120  = 4.2456  % OF BUFFER: 20000000
Loading initial image.
Loading observations.
Loading inverse variances.

 MEM iteration: 1
Gaussian default being computed
Finished setting default
     OPUS  1  --->  23
                                    C    ===  2.64250E+00
   TROPUS  6  <---  28
     S    === -2.70009E-05    TEST ===  0.98582
     OPUS  2  --->  24
     OPUS  3  --->  25
     SNEW === -2.00315E-02    dist ===  0.20000    CNEW ===  2.59695E+00
   TROPUS  6  <---  28
     OPUS  4  --->  26
     SNEW === -2.00286E-02    dist ===  0.20000    CNEW ===  2.59560E+00
```

NDEF is the most significant parameter here – if '1' is entered, the iterations will change the large scale of the image to fit the data, whilst '2' will smooth the image and change the smaller structure to fit the data. It is therefore best to start with NDEF = 1 and iterate a few steps step at a time, checking the image using Gaia (the starting image file itself is edited, so just keep reopening it) after each set of iterations, to ensure the image is improving and that you don't iterate too much and start to degrade the image. When the image is as good as possible using NDEF = 1, change to NDEF = 2 and repeat the procedure to produce the final image.

Once your tomogram is finished, it is possible to add contours, using the 'contouring' command under the 'Image-Analysis' menu. An even set of contours can be generated under the 'Generate' tab, then the values, colours etc. of each can be adjusted in the 'Levels' tab. The contours can then be displayed by clicking the 'Draw contours' button at the bottom of the window. The 'Region' tab controls the amount of the diagram on which the contours are drawn, allowing you to exclude contours of any fluctuations away from the main structure, whilst including everything you do want.

It is also possible to add a grid and axes, using the 'overlay axes grid' command in the 'Image-Analysis' menu. As with the contouring, use the various tabs to control the line styles, positions, colours, fonts etc. and click 'Draw' to display. The default settings put the axes through the middle of the image - untick the 'Interior' box, under 'Suggested axis placement' in the 'Position tab' to put the axes at the edge of the image.

Gaia

The files used by Doppler are all in the .sdf format. This is a graphical format, but is not readable by most graphical programmes, such as DS9. The program 'Gaia', a version of Starlink, can, however.

Using this program, it is possible to check the trail created by molly and track the changes to your tomogram image as it is iterated. It has a graphical interface which is easy to use, although the 'Autocut' in the main interface often needs to be adjusted in order for images to be displayed correctly.

This ability is not currently available at the CfA.