AIPS MEMO NO. 6

M E M O

TO : Tim Cornwell, Ron Ekers, Jacqueline van Gorkom, Pat Palmer, and the Apes of Charlottesville

FROM : Arnold Rots

SUBJECT : Spectral Line Software Wish List for AIPS

DATE : 5 October 1981

This memo contains a wish list of spectral line software for AIPS. It does not contain any of the cube-handling routines described earlier by Ed Fomalont; it is understood that these are available or will be available in the near future.
Some of the items in this list may be in the system already, or may only need minor changes. They are included anyway to give a complete overview of the spectral line software that is needed.

1. MAPPING STAGE

1.1. Selfcal

There should be a way to do a self calibration on one pseudo continuum data base (one line channel), and apply the solution automatically to a collection of such data bases (i.e. to all other spectral line channels).

1.2. Mapping

It is assumed that multiple 2-dimensional maps will be made from a multi-file export tape, using the same mapping parameters. One should ensure that the Stokes parameters are set correctly.

1.3. Clean

It should be made possible to clean one map (channel) and use the components for cleaning all other maps (channels). It would be useful if Clean could operate on a dirty map cube, with associated beam cube (all in real format), and produce another cube.

2. MAP PLANE ANALYSIS

This section refers to processing of map cubes in (x,y,f) order.

2.1. Smoothing

A map plane smoothing function should be available. With the amount of disk space that spectral line data is going to require

it does not seem feasible to expect the user to remap the data, while restarting clean with a different restoring beam is too clumsy. One should be able to specify major axis, minor axis, and position angle of the RESULTING beam, and the program should work on the entire cube.

2.2. Map rotation

One should be able to rotate an entire cube around an axis perpendicular to the (x,y) plane.

2.3. Slice integral

When a one- or two-dimensional slice is taken through a cube, it should also calculate the integral.

2.4. Window integral

The user should be able to get the integral within an arbitrary polygon in any two-dimensional plane - at least the ones parallel to the first two axes.

2.5. Continuum subtraction

There should be a simple task subtracting a two-dimensional map from each plane of a map cube.

2.6. Optical Depth

There should be an optical depth calculation task, similar in set-up to 2.5.

2.7. Simple moments

One should at least be able to calculate zeroth, first and second moments of a map cube, using either a straight or an absolute value cut-off.

2.8. False color encoding

A task is needed to produce (R,G,B) maps on the basis of an intensity (zeroth moment), hue (first moment), and saturation (second moment) map.

3. PROFILE ANALYSIS

In this section we will assume a map cube in order (f,x,y) or (f,y,x). Some of the items are duplicated from section 2.

3.1. Display

It is assumed that the basic labeling of the "third axis" is in frequency, both because it is the basic unit, and because the grid is linear in this unit. The catalog header should contain:

- velocity of the reference pixel
- restfrequency
- velocity restframe (HEL, LSR, or OBS)
- velocity definition (optical or radio)

From these parameters it is very easy to convert from frequency to velocity.
It is further assumed that display programs offer a switch to either label this axis in frequency or in velocity.

3.2. Smoothing

A Hanning smoothing in frequency is needed.

3.3. Continuum subtraction

In addition to the simple continuum subtraction as in section 2.5., building and subtracting a continuum map based on arbitrary polygon selection in the (f,x) planes is needed.

3.4. Optical depth

The optical depth calculation from section 2.6. should also be capable of operating on the transposed cube.

3.5. Gaussian analysis

A task is needed to do gaussian analysis on profiles. It should be capable of fitting multiple one-dimensional gaussians.

3.6. Moments

It should be possible to calculate two-dimensional moment maps from profiles; both the simple moment calculation based on a cut-off, and a more sophisticated one based on arbitrary polygon selection in the (f,x) planes.