# Chapter 12

# *UV* Plane Analysis

**Robert Lucas**

IRAM, 300 rue de la Piscine, F-38406 Saint Martin d'Hères

## 12.1 *uv* tables

After calibration with CLIC, the calibrated data may be stored in a particular file called a '*uv* table'. This is useful because much of the data in the CLIC data file are not needed any more: atmospheric parameters, total powers, image side band visibilities, data from other receivers may be discarded at this stage. All that counts is: the data that are needed to describe the source itself, the sky frequency that was observed, ... One may for instance create a *uv* table for the continuum and one for each line that was observed.

These *uv* tables are just special GILDAS tables suited for *uv* data handling that are created by CLIC. Mapping consists of transforming these tables into something more meaningful for the astronomer, either images or numbers like positions, flux densities, sizes, etc. However a good part of the data evaluation and analysis can be directly performed on the *uv* data itself, before performing any of the complex operations involved in creating an image (Fourier transform and deconvolution). Direct analysis of the *uv* data is the subject of this Lecture.

### 12.1.1 *uv* table contents

A *uv* table is a file in the Gildas Data Format, of dimensions $[3N_c+7, N_V]$, for $N_c$ spectral channels and $N_V$ visibilities. The $3N_c+7$ lines contain:

1. $u$ in meters
2. $v$ in meters
3. Scan number
4. Observation date (integer CLASSday number)
5. Time in seconds since above date

6. Number of start antenna of baseline
7. Number of end antenna of baseline
8. First frequency point (real part)
9. First frequency point (imaginary part)
10. First frequency point (weight)
11. Same for second frequency point, and so on

Thus for a given scan with $N_A$ antennas, $N_A(N_A-1)/2$ visibilities are recorded.

The table header has the standard form of a GILDASImage. The header is available (for instance) by declaring:

```
GRAPHIC> SIC\DEFINE HEADER T co10.uvt READ
GRAPHIC> EXAMINE T%
```

For a table named `co10.uvt`. Some keywords convey a more precise meaning for *uv* tables:

**T%NDIM** should be 2

**T%DIM** contains $3N_c$+7 and $N_V$

**T%RA,T%DEC** coordinates (radians) of the pointing center (the center of the primary beam).

**T%A0,T%D0** coordinates (radians) of the phase tracking center (a point source at this point should have zero phase); they are identical to RA and DEC when a table is first produced.

**T%EPOCH** The epoch of those coordinates. Should be 2000.0

**T%VELOFF, T%VELRES** The velocity of the reference channel, and the channel separation in velocity units (km/s)

**T%RESTFRE, T%FREQRES** The rest frequency, and the channel separation in frequency units (MHz)

**T%CONVERT[1,1]** the reference channel

**T%CONVERT[1,2]** the actual observing frequency at the reference channel (MHz); the one used to scale angular displacements from $u, v$ coordinates in meters.

One may also examine directly the header by typing simply :

```
GRAPHIC> HEADER co10.uvt
```

### 12.1.2 How to create a *uv* Table

*uv* Tables are created by CLIC using the command **TABLE**.

A set of commands to create a *uv* table may look like:

```
! Reset the default options:
  SET DEFAULT
! find the useful scans:
  FILE IN 21-JAN-1998-H126
  SET SOURCE IRC+10216
  SET RECEIVER 1
  SET PROCEDURE CORRELATION
  SET QUALITY AVERAGE
  FIND
! calibration options:
```

```
  SET AMPLITUDE ANTENNA RELATIVE
  SET PHASE ANTENNA RELATIVE INTERNAL ATMOSPHERE
  SET RF ANTENNA ON
! table creation:
  SET SELECTION LINE LSB L01
  TABLE HCN NEW /FREQUENCY HCN  88631.85 /RESAMPLE 19 10 -27 2.12 V
```

All but the last two commands should be familiar at this point.

- The first new command, `SET SELECTION`, prepares the last one `TABLE`. It selects that the next table to be created will be a line table (i.e. with more than one spectral channel). The lower side band data will be used, and only the first subband of the correlator: `L01`.
- The last command `TABLE`, actually creates the table named `hcn.uvt`. The rest frequency, 88631.85 MHz, is set to be the reference used for the velocity scale. The data will be resampled to a velocity grid of 19 channels; the reference channel 10 will correspond to the LSR velocity -27 km/s; the channel spacing will be 2.12 km/s. Without `/RESAMPLE`, one would have got all the channels in the subband L01 with their original velocity separation. Without `/FREQUENCY`, the rest frequency present in the data (in the observing procedure) would have been used.

Using `/RESAMPLE`, one may avoid creating tables with too many channels (by discarding unused parts of correlator subbands) and choose the resolution that is actually needed.

If the data is spread on several files, one may go on by opening the other files, finding the data scans, and appending to the table:

```
  FILE IN 12-FEB-1998-H126
  FIND
  TABLE
  FILE IN 21-FEB-1998-H126
  FIND
  TABLE
...
```

(the arguments to `TABLE` need not to be repeated).

For continuum tables one may use:

```
SET SELECTION CONTINUUM DSB L01 TO L05 -
  /WINDOW 214405 214726 217476 217796 217837 217875
TABLE CONT-1MM NEW
```

Here we are using data from all the line subbands, but only in the three frequency windows: 214405 to 214726 MHz, 217476 to 217796MHz, and 217837 to 217875. This is of course to avoid the line emission of some molecules.

A standard menu is available under the CLIC main menu ("Create a UV Table"). After execution, a specific procedure is created to keep track of the options and parameters used. This procedure can subsequently be edited to add new data files (data files can also be added from the menu).

## 12.2 *uv* data plots

A procedure is available to do various plots from a continuum or line table. Its name is `UVALL` and it is called by clicking on "Interferometric UV operations" in the GRAPHIC standard menu. One has to select the first and last channel to be plotted (0 0 to get all channels) and the name of the parameters to be plotted in abscissa and ordinate. The following examples are the most useful plots:

***uv* coverage:** to get an idea of the imaging quality that may be obtained, to check if one configuration has been forgotten, ...

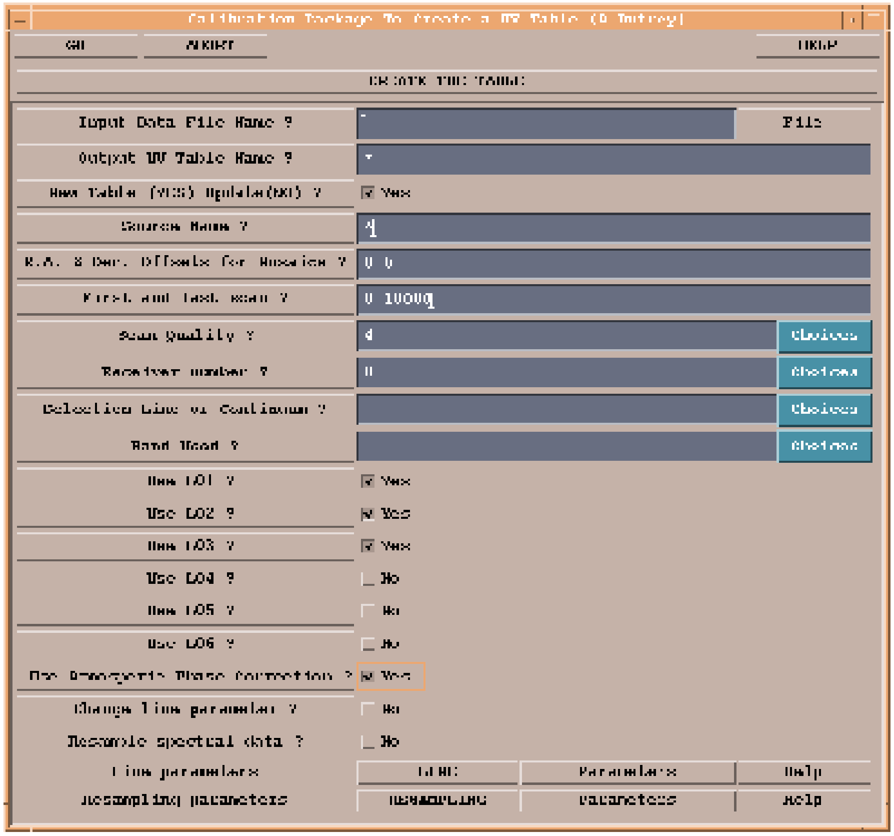


Figure 12.1: "Create a UV Table" menu in CLIC

**weight vs. number:** check if some data got strange weights (e.g., zero) for any reason

**Amplitude vs. antenna spacing:** quite useful if a source is strong to see if it looks resolved. Also check for spurious high amplitude points.

**Amplitude vs. weight:** another useful check: spurious high-amplitude points with non-negligible weight can cause a lot of harm in a map.

These plotting facilities are also implemented in the MAPPING program as a command (`SHOW UV`).

## 12.3 Data editing

Editing the interferometer data is not very flexible when done in the *uv* tables. If a problem occurs, is is not easily diagnosed since many of the parameters associated with the data acquisition are not present: atmospheric data, total powers, ... It is however useful when something strange occurs in the mapping process to do *uv* plots of the input table to look for the faulty data. Using time or scan number as X coordinate is recommended. One then may go back to the CLIC program and flag the faulty data, tag the corresponding scans, and keep a log of these problems so that they are not encountered again when the *uv* tables are reconstructed for any reason.

Two tasks have been written that may directly edit the data in *uv* tables:

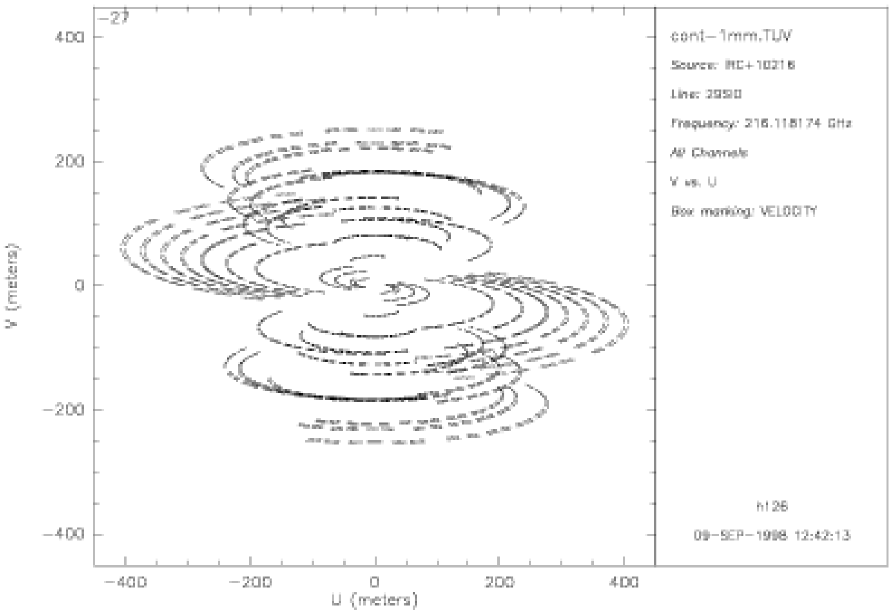


Figure 12.2: Example of a *uv* coverage plot

`UV_CLIP` flags all visibilities larger than a given flux: this deletes visibility points with totally wrong numbers, if any.

`UV_FLAG` deletes visibility points in a give time interval for a given baseline.

Both tasks work by setting the corresponding weight to zero: their action is irreversible (you will have to reconstruct the *uv* table to go back).

The MAPPING program provides a more efficient, simpler and reversible interactive tool to flag parts of a *uv* data set (command `UV_FLAG`).

## 12.4 Position shift

For the purpose of further data reduction it may be necessary to change the phase center of the *uv* data. This is done by a simple rotation of the phases (multiplication by a complex factor). One uses the task `UV_SHIFT` for this purpose.

## 12.5 Averaging

### 12.5.1 Data compression

One may simply wish to average several channels to increase the signal to noise ratio (use tasks `UV_COMPRESS` and `UV_AVERAGE`).

### 12.5.2 Circular averaging

For sources with circular symmetry it may be necessary to obtain the variation of amplitude with antenna spacing, in order to compare the amplitude data with models. For this purpose, with task `UV_CIRCLE` one

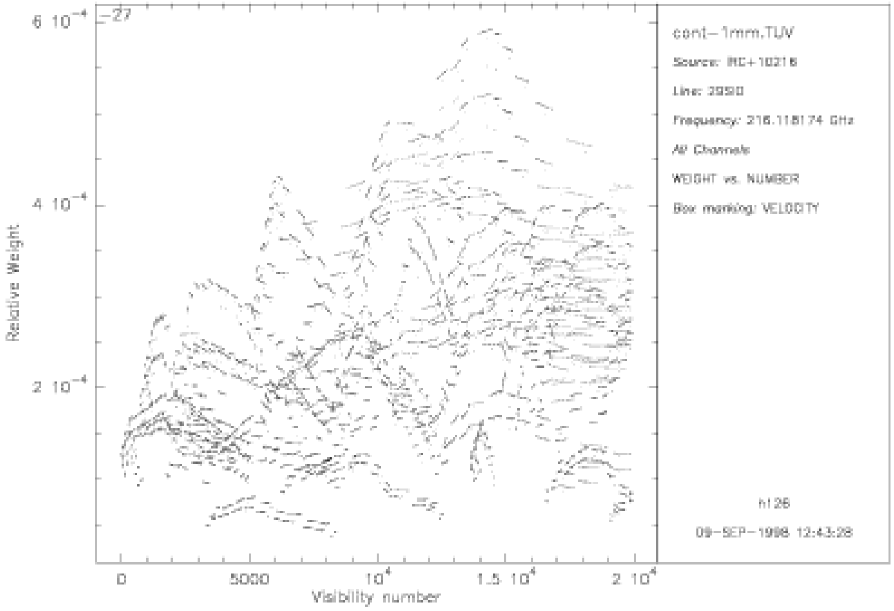


Figure 12.3: Weight versus visibility number plot

takes the mean of all the visibilities in concentric rings in the $uv$ plane. The output has the format of a $uv$ table (except that all $v$'s are zero), and may be plotted with UVALL (fig. 12.5).

Use this sort of averaging with caution: the phase center must accurately coincide with the source position or the amplitude of the visibility average will decrease on long spacings (use UV_FIT and UV_SHIFT. One may also do this kind of averaging in separate sectors in the $uv$ plane, to check for asymmetries, provided the $uv$ plane is well sampled (task UV_CUTS).

## 12.6 Model fitting

Model fitting is the oldest way of analyzing interferometer data. It was effectively used in the times where the coverage of the $uv$ plane was too scarce to even think of creating an image by Fourier transform. One assumes a simple source model depending of a few parameters (source position, flux, size) and fits the visibility function of that model to the visibility data. Of course one may use a linear combination of several source models since the Fourier transform is linear. This is performed using the GILDAS task UV_FIT. The result may be displayed with the procedure PLOTFIT. Both are available in the panel "Interferometric UV operations" from the GRAPHIC standard menu.

Table 12.1 gives examples of a few models and their visibility functions. For source models with a circular symmetry, the visibility function is split into a radial dependent amplitude and a phase factor which depends only on the source position.

Some sources are actually so simple that this method may be used to a good accuracy (fig. 12.6).

Quite often this method is used for sources that are unresolved or not well resolved at a given frequency; for instance a SiO maser may consist of several point-source components at different velocities. Fitting a point source in each channel one derives a "spot map" (figs 12.7,12.8).

| Parameters: | | Variables: | |
|---|---|---|---|
| $x_0$ | RA position | $x, y$ | sky position |
| $y_0$ | DEC position | $r$ | $\sqrt{(x-x_0)^2+(y-y_0)^2}$ |
| $S$ | Source flux | $u, v$ | projected spacing |
| $b$ | HP size | $q$ | $\sqrt{u^2+v^2}$ |

| Name | Model | Visibility |
|---|---|---|
| Point source | $S\ \delta(x-x_0, y-y_0)$ | $S\ e^{-2i\pi(ux_0+vy_0)}$ |
| Gaussian | $\frac{4S}{\pi b^2 \log 2}\ e^{-4\log 2 \frac{r^2}{b^2}}$ | $S\ e^{-\pi^2/4/\log 2\ (bq)^2}\ e^{-2i\pi(ux_0+vy_0)}$ |
| Disk | $\frac{4S}{\pi b^2}$ where $\lvert r \rvert < b$ | $S J_1(\pi bq)\ e^{-2i\pi(ux_0+vy_0)}$ |

Table 12.1: A few source simple source models and their visibility functions

### 12.6.1 Position measurement

For a source with central symmetry the task `UV_CENTER` determines the source position by using only the phases. Alternatively the task `UV_FIT` may be used to fit the amplitudes and phases at the same time, or e.g. to simultaneously fit a pair of sources.

## 12.7 Continuum source subtraction

It is straightforward to subtract a point source at the phase center in the *uv* data: one simply subtract a real number (the source flux) from all the visibilities.

The task `UV_SUBTRACT` subtracts a time-averaged continuum *uv* table from a spectral line table (this assumes that the continuum and the line have been observed simultaneously), providing a new table with the line emission. Note that if the source is too complex, the time averaging (needed to avoid increasing the noise level in the resulting table), may affect the structure of the subtracted continuum image.

If the continuum data was not observed in the same session, or is known only from other sources, one may build a *uv* table of the continuum using the task `UV_MODEL`. This tasks computes that table from a model image or data cube; it computes the corresponding visibilities at *uv* coordinates taken from a reference *uv* table (e.g. the table out of which one wants to subtract the continuum model).

## 12.8 Self calibration by a point source

In some cases the continuum is a point source, and is strong enough to be used to reference the phases and amplitudes of the line data. The phase and amplitude referencing can be done in CLIC command `STORE PHASE /SELF` or in the *uv* tables using task `UV_ASCAL`.

For this one creates separately a line table and a continuum table. Naturally both must have been observed simultaneously. It is strongly advised to self-calibrate the phase first, then the amplitude using a longer smoothing time. `UV_ASCAL` allows in addition to subtract (part of) the input reference source, typically to provide continuum-free spectral line maps.

Note that this operation will work also if the source is somewhat extended, but with central symmetry; in that case only the phases can be self-calibrated.

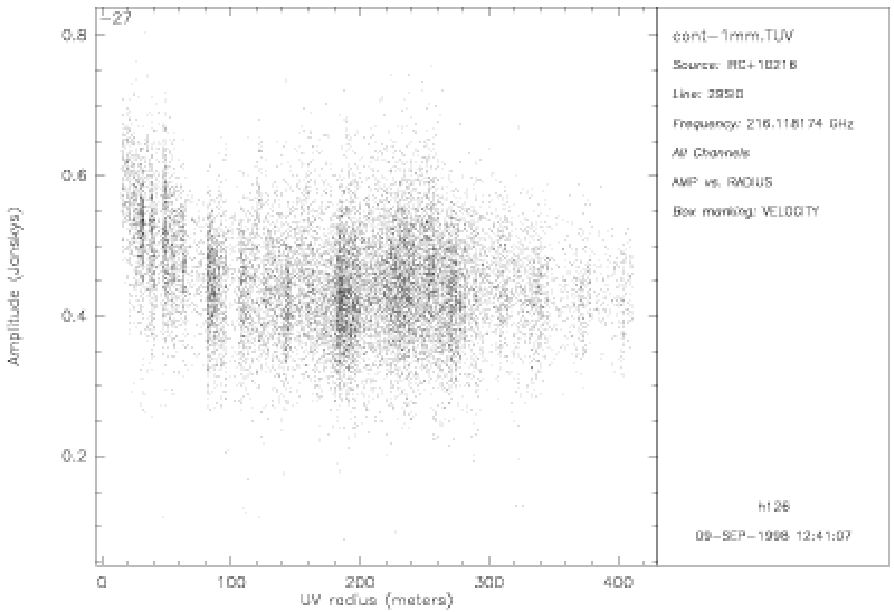


Figure 12.4: Amplitude versus antenna spacing plot

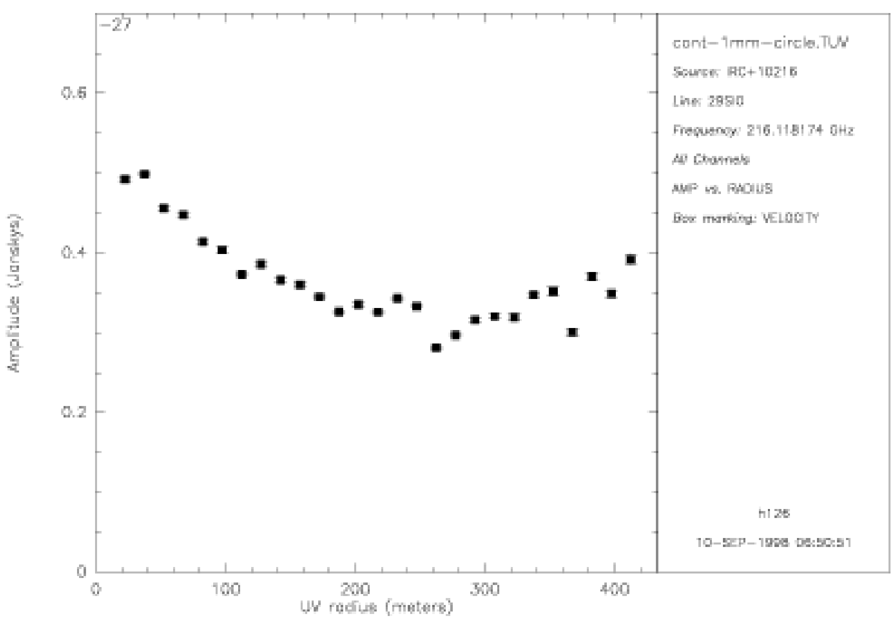


Figure 12.5: Example of a circular average plot (same data as fig. 12.4)

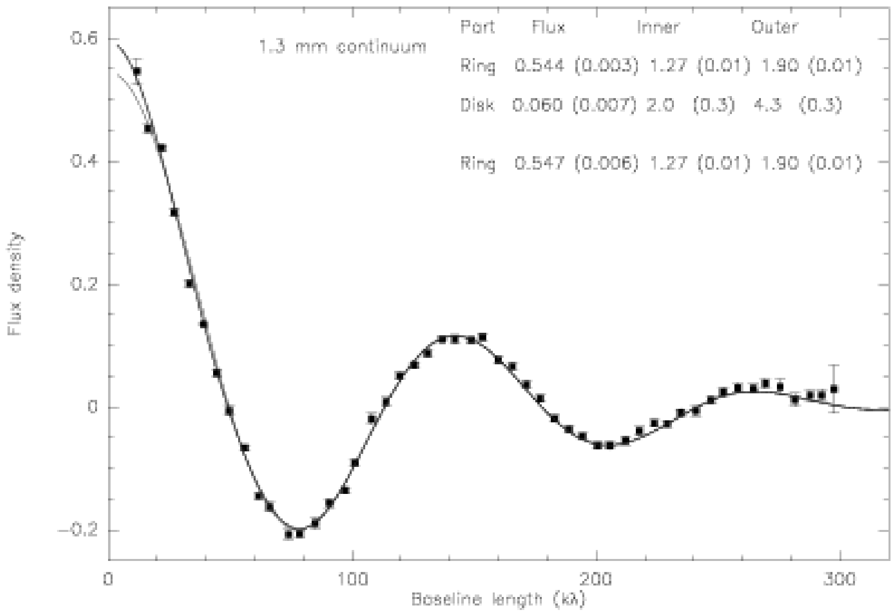


Figure 12.6: $uv$-plane fit to the disk around GG Tau at 1.3 mm (from [Guilloteau et al 1999]).

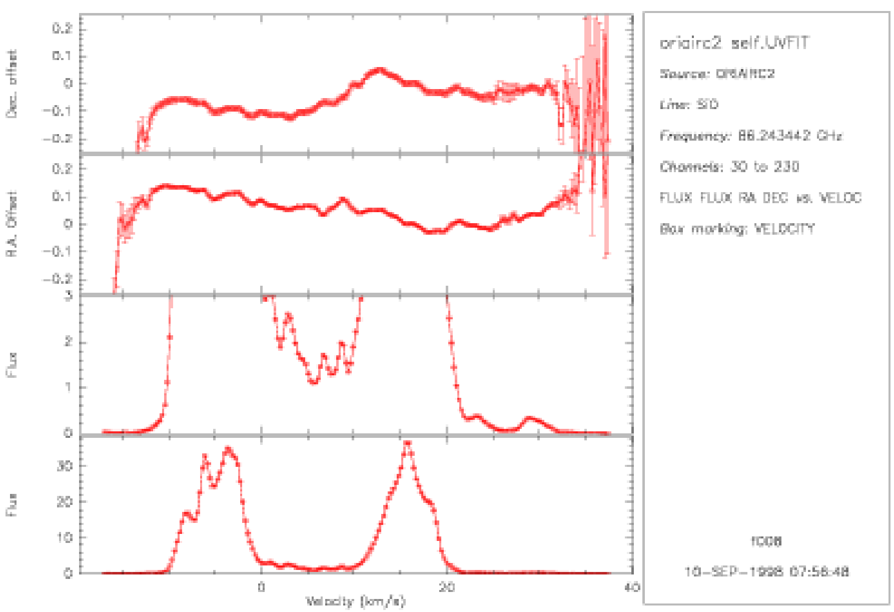


Figure 12.7: Result of a multi-channel point source fit to the Orion SiO (v=2, J=2-1) maser

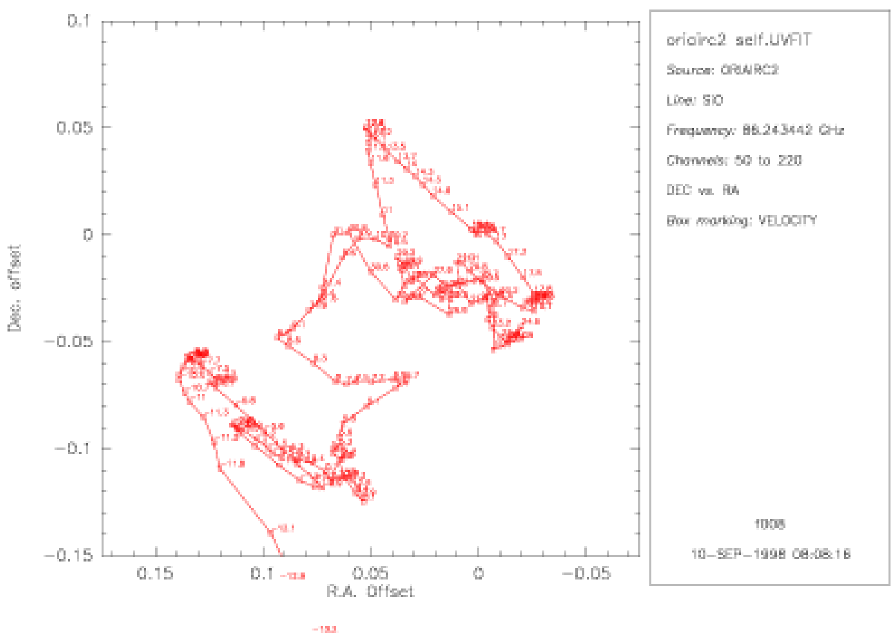


Figure 12.8: "Spot map" of the Orion SiO (v=2, J=2-1) maser