UV Plane Analysis

Frédéric Gueth

IRAM Grenoble

UV Plane analysis

- The data are now **calibrated** as best as we can
  - Caution: data are calibrated, but not perfect: phase fluctuation remains!
- Next step: produce a (multichannel) **image** by Fourier transform

Calibration (CLIC) $\longrightarrow$ uv-table $\longrightarrow$ Imaging & Deconvolution (MAPPING)

- **Some work can/must be done directly in the uv plane**
  - data checking and editing before imaging
  - analysis (model fitting, continuum subtraction, ...)

uv tables (1)

- *uv* tables are used to store the calibrated data needed to produce a map
- **Why this intermediate file?**
  - **many parameters are not needed any more** at this point: atmospheric parameters, total powers, correlator setup, image side band visibilities, ...
  - **data selection** (e.g. with the `Data quality assessment` tool)
  - **small files**
  - **flexible data format**, fully editable in GILDAS
- A *uv* table is created for each line and continuum band

uv tables (2)

- GILDAS **.uvt** format, very similar to `UVFITS`
- Content for each visibility point:
  1. **u in meters**
  2. **v in meters**
  3. Scan number
  4. Observation date (CLASS day number)
  5. Time in seconds since above date
  6. Start antenna of baseline
  7. 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, ...

uv table creation in CLIC (1)

```
!
! Select correlations on NGC 2264
!
FIND /SOURCE NGC2264 /PROCEDURE CORRELATION
!
! Apply the calibrations
!
SET RF          ANTENNA ON
SET PHASE       ANTENNA RELATIVE INTERNAL ATMOSPHERE
SET AMPLITUDE ANTENNA RELATIVE
!
! Create the UV table
!
SET SELECTION LINE LSB L01
TABLE hcn.uvt NEW /RESAMPLE 19 10 -27 2.12 V
```

uv table creation in CLIC (2)

- `SET SELECTION LINE LSB L01`
  - the table will be a line table (i.e. with more than one spectral channel)
  - the lower side band data will be used
  - only the first subband of the correlator: `L01`

- `TABLE hcn.uvt NEW /RESAMPLE 32 10 -27 2.12 V`
  - the data is **resampled** to a velocity grid of 32 channels
  - the reference channel 10 corresponda to the LSR velocity -27 km/s
  - the channel spacing will be 2.12 km/s

uv table creation in CLIC (3)

- Without `/RESAMPLE`, all channels in the subband L01 are written with their original velocity separation
- `/RESAMPLE` option
  - **avoid creating tables with too many channels** (unused parts of correlator subbands are discarded)
  - choose the resolution that is actually needed: **improves signal to noise ratio**

  It is better to resample the data before imaging than after!
- Other useful option of `TABLE` command: `/FREQUENCY` to redefine reference frequency in *uv* table
  - if several lines were observed simultaneously
  - if exact line frequency was unknown (high-z sources...)

uv table creation in CLIC (4)

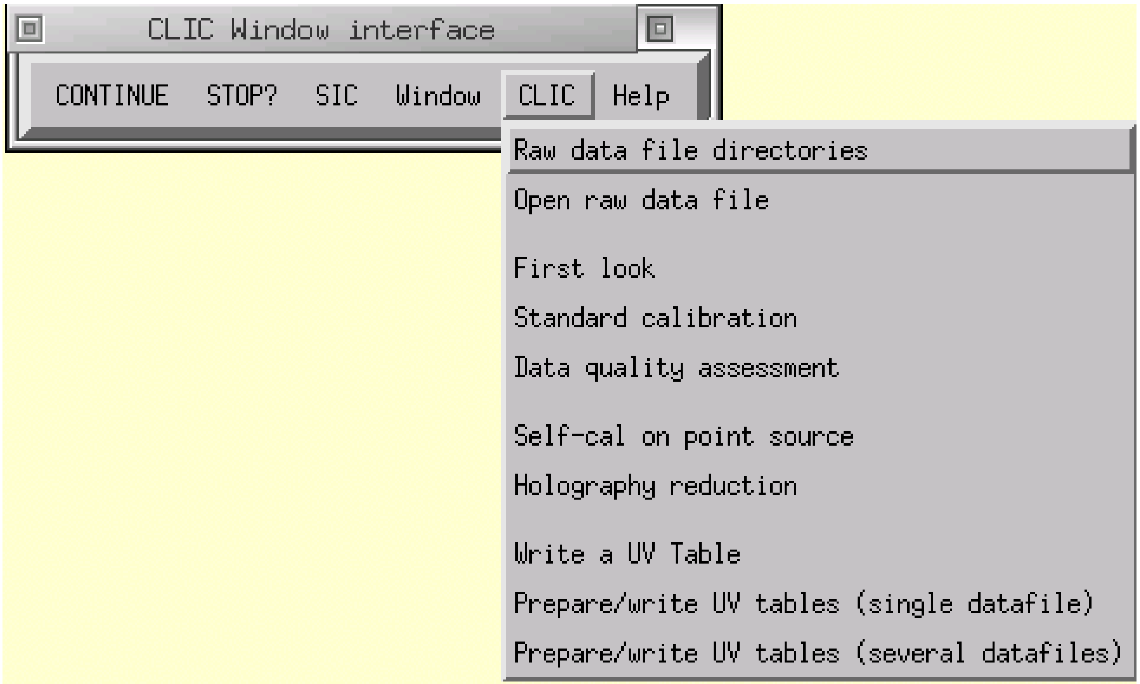
- Continuum table:

```
SET SELECTION CONTINUUM DSB L01 TO L05 -
   /WINDOW   214405 214726   217476 217796
TABLE cont-1mm NEW
```

- Use the correlator subbands #1 to #5, DSB data, but only in two **frequency windows**
  - 214.405 to 214.726 GHz
  - 217.476 to 217.796 GHz

  This is of course to **discard the line emission** of some molecules

uv table creation in CLIC (5)



uv data plots: UVALL

- Procedure **UVALL** is used to do various plots:
  - U, V, radius, angle, time, date, scan, number
  - amplitude, phase, real, imaginary, weight

- Examples:

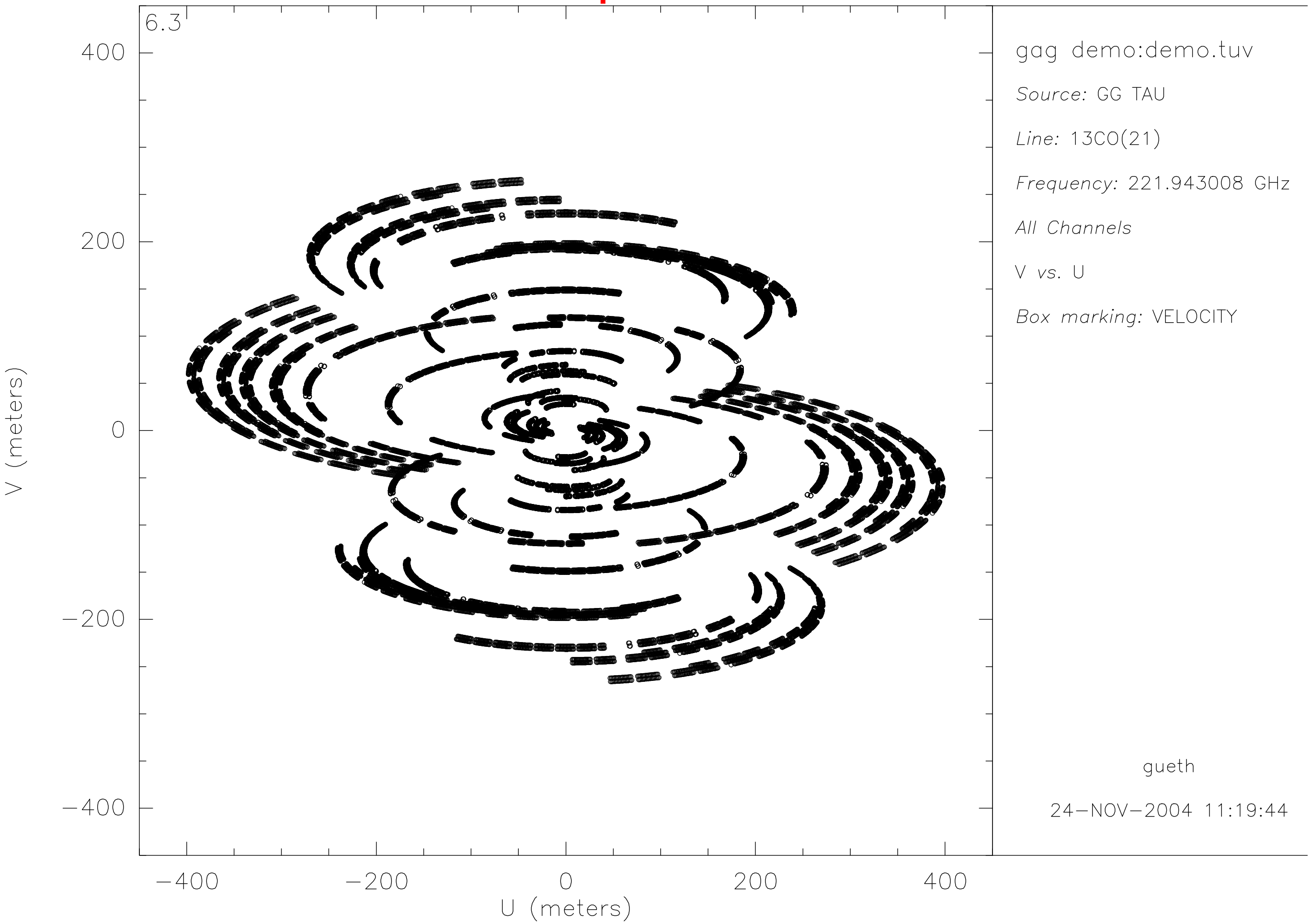
  uv coverage (U vs. V): basic property of the observations

  weight vs. scan: e.g. to look for spurious weights

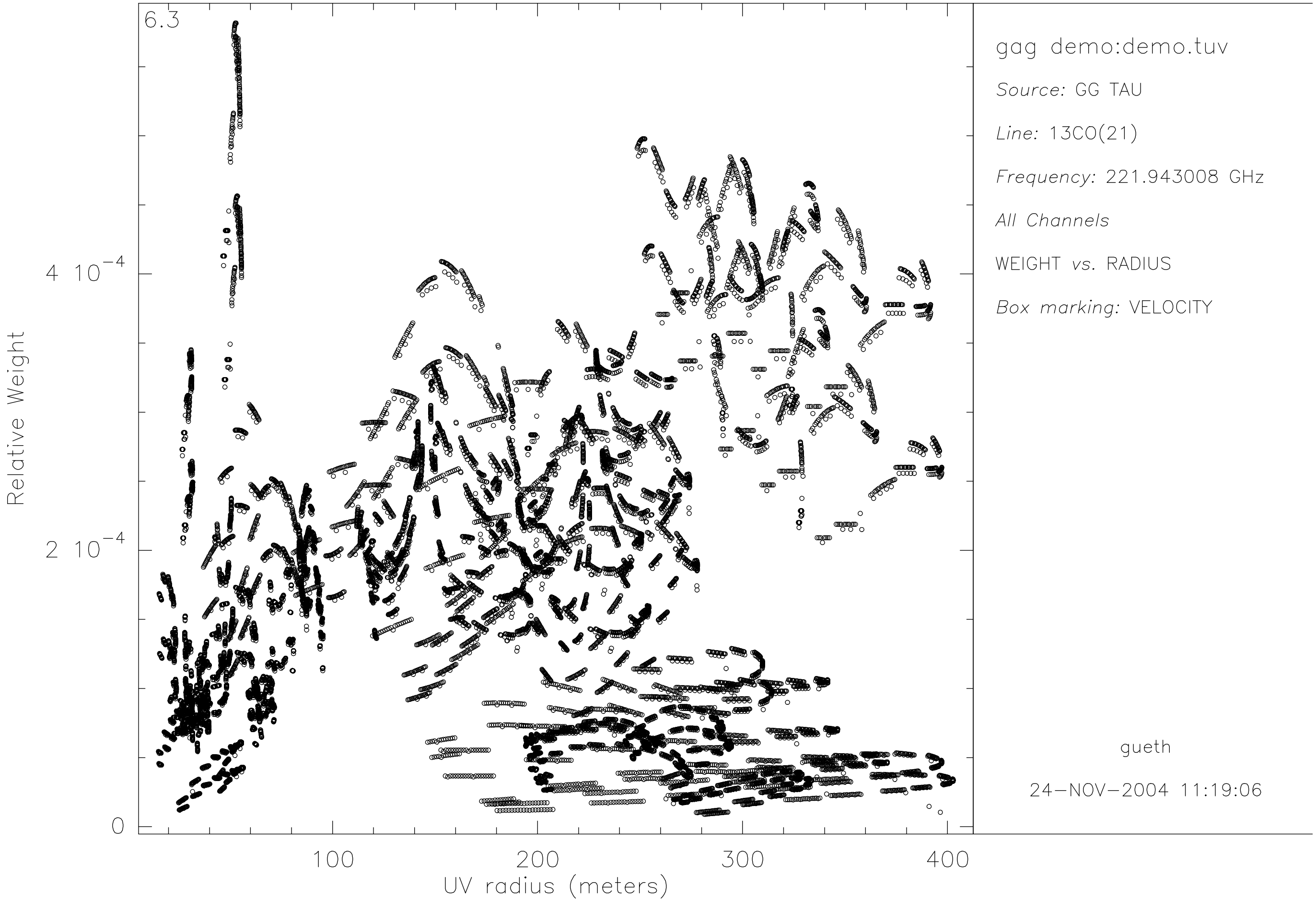
  amplitude vs. antenna spacing: useful if the source is strong

  amplitude vs. weight: spurious high-amplitude points with non-negligible weights can cause a lot of harm in a map

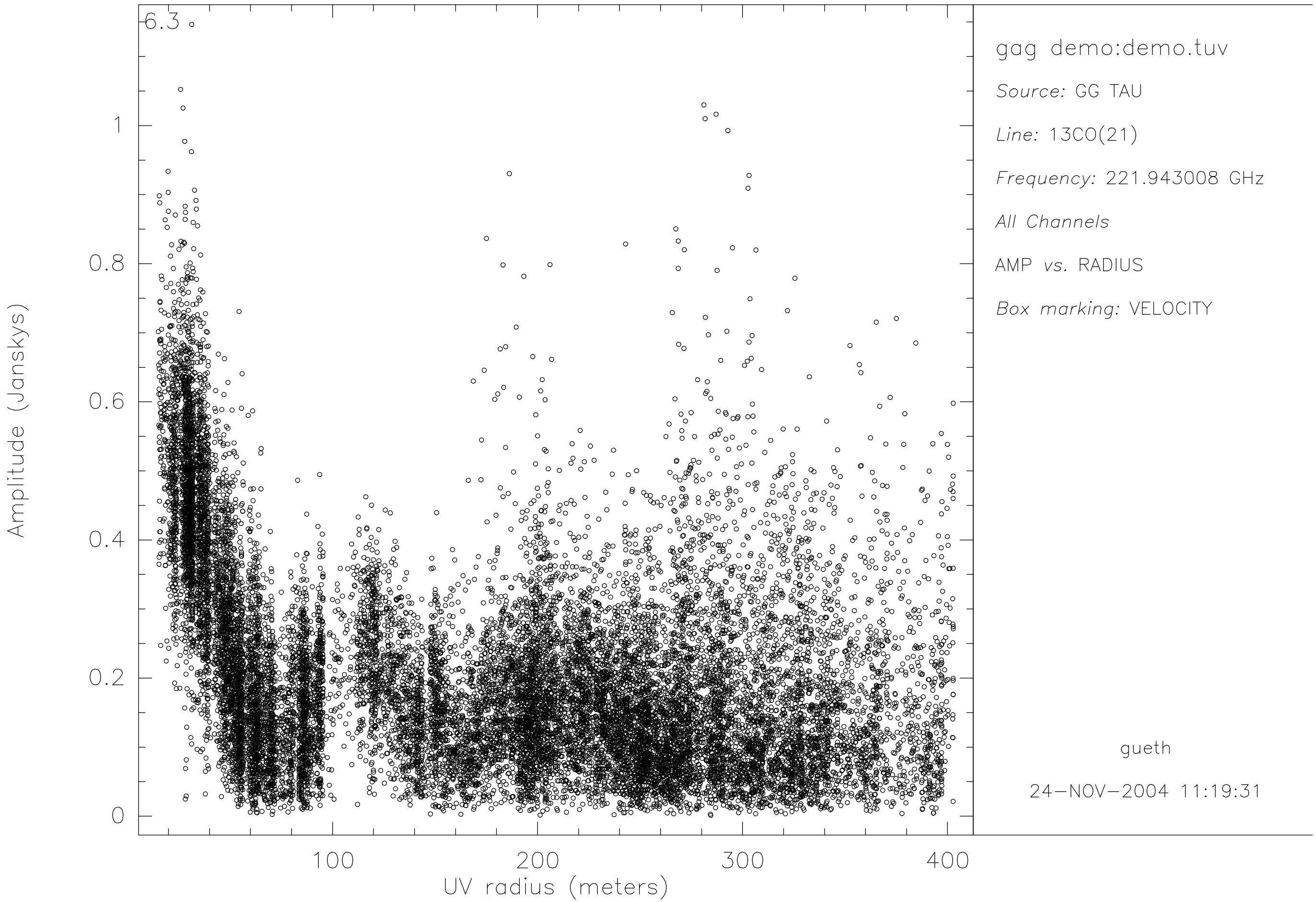
uv data plots: u vs. v



uv data plots: weight vs. antenna spacing



uv data plots: amplitude vs. antenna spacing



Data editing (1)

- Reminder: **data selection has occured before *uv* table creation** (e.g. using the `Data quality assessment` tool)
- Reminding problems: corrupted scans
- Data editing in *uv* table: remove bad points, e.g. by setting their weight to zero
- **Good practices:**
  - *uv* plots used to look for the wrong/corrupted data
  - go back to `CLIC` to do a full diagnostic (all relevant parameters are available)
  - flag the data
  - re-create the *uv* table

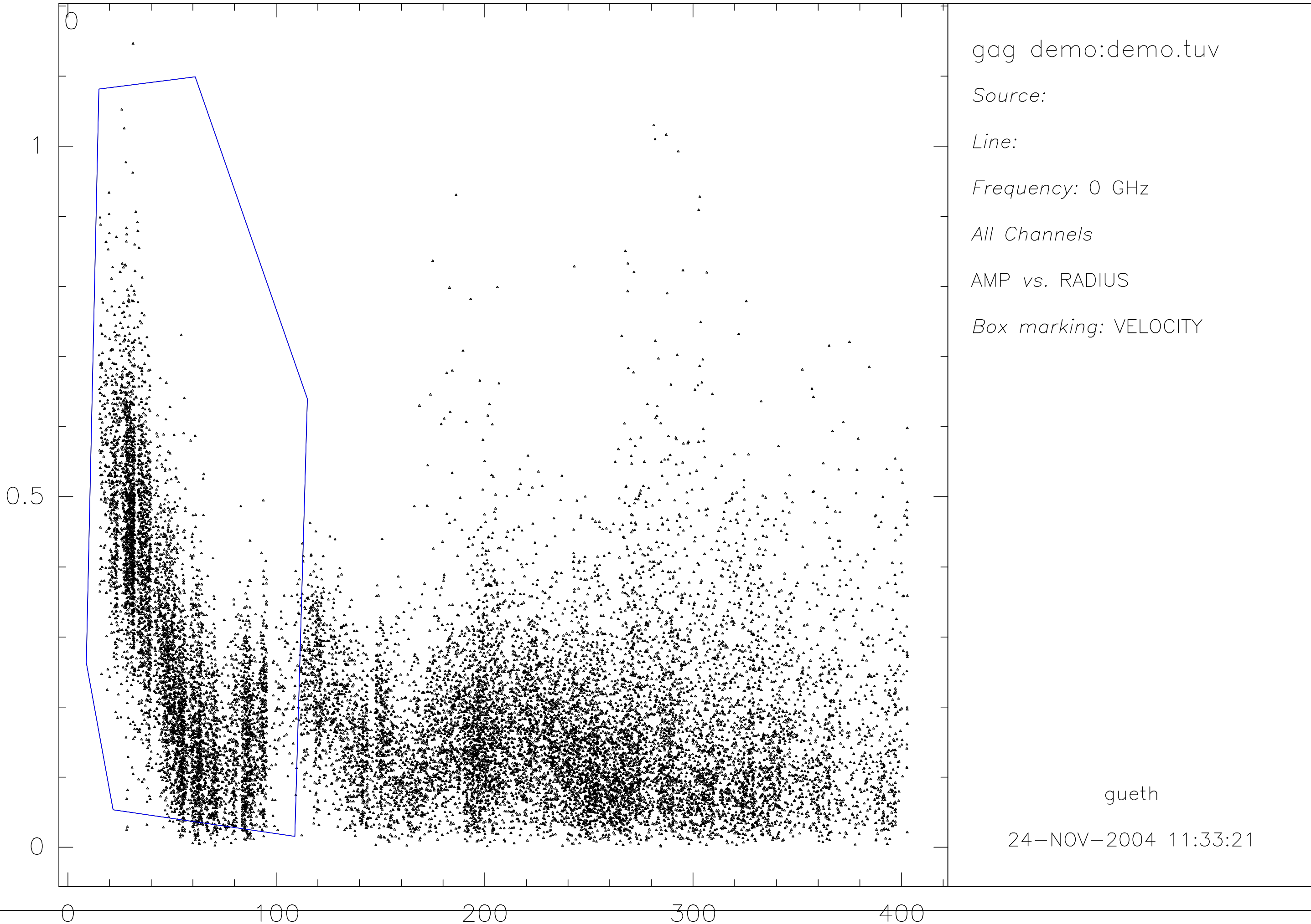
Data editing (2)

- **Interactive program:** `UV_FLAG` in `MAPPING`
- Tasks automatically edit the data in *uv* tables, e.g. to
  - flag all visibilities larger than a given flux (wrong values)
  - delete visibility points in a given time interval for a given baseline

  These tasks work by setting the corresponding weight to zero ⟶ their action is irreversible
- **Content of *uv* tables fully editable in GILDAS**

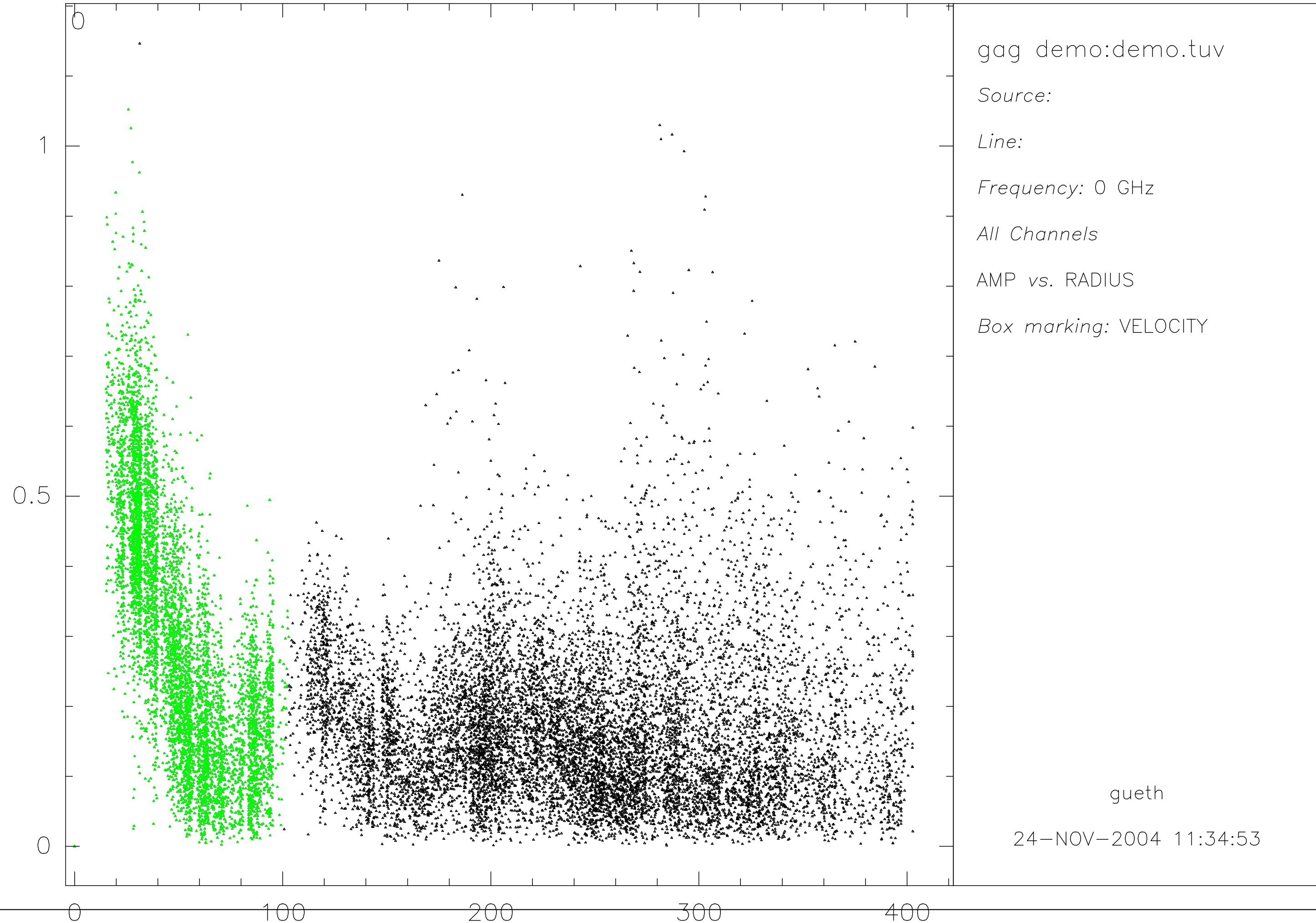
```
DEFINE TABLE AA mytable.uvt WRITE
LET AA[34,10] 2264
DELETE /VARIABLE AA
```

UV_FLAG

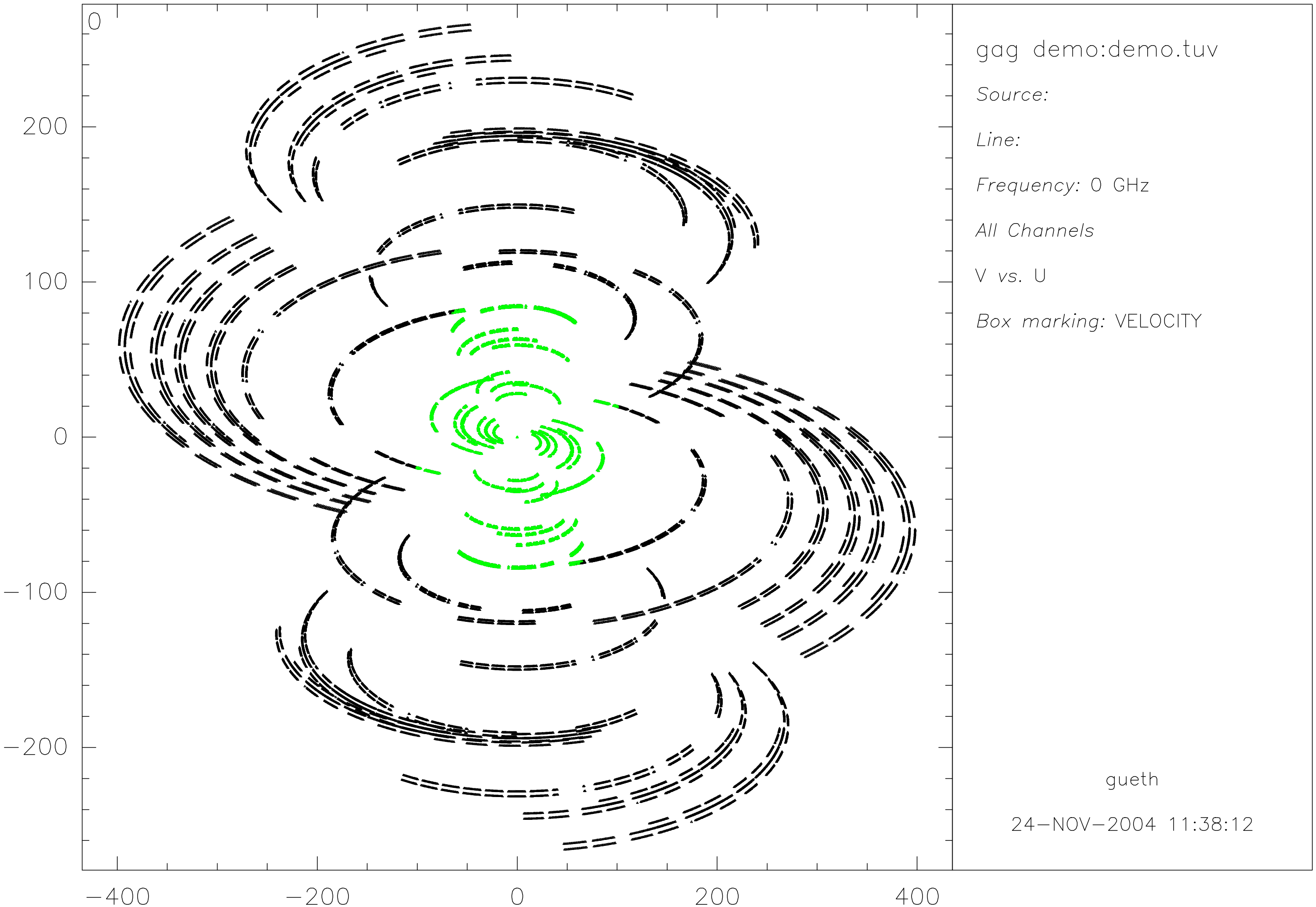


0
1
0.5
0
0
100
200
300
400
gag demo:demo.tuv
Source:
Line:
Frequency: 0 GHz
All Channels
AMP vs. RADIUS
Box marking: VELOCITY
gueth
24−NOV−2004 11:33:21

UV_FLAG


0
1
0.5
0
0
100
200
300
400
gag demo:demo.tuv
Source:
Line:
Frequency: 0 GHz
All Channels
AMP vs. RADIUS
Box marking: VELOCITY
gueth
24–NOV–2004 11:34:53

UV_FLAG


0
gag demo:demo.tuv
Source:
Line:
Frequency: 0 GHz
All Channels
V vs. U
Box marking: VELOCITY
gueth
24–NOV–2004 11:38:12
200
100
0
−100
−200
−400
−200
0
200
400

Data editing (2)

- **Interactive program:** `UV_FLAG` in `MAPPING`
- Tasks automatically edit the data in *uv* tables, e.g. to
  - flag all visibilities larger than a given flux (wrong values)
  - delete visibility points in a given time interval for a given baseline

  These tasks work by setting the corresponding weight to zero ⟶ their action is irreversible
- **Content of *uv* tables fully editable in GILDAS**

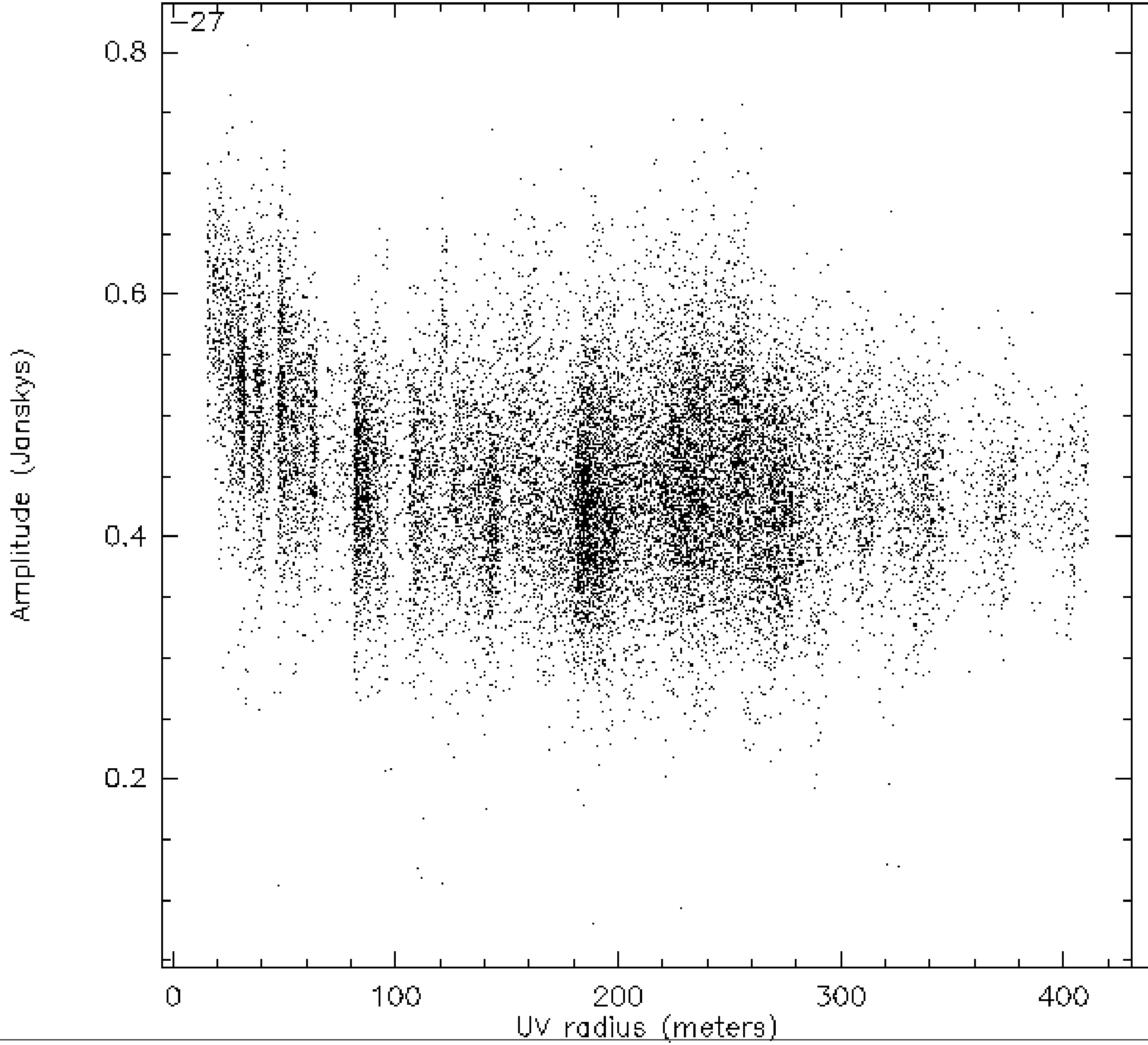
```
DEFINE TABLE AA mytable.uvt WRITE
LET AA[34,10] 2264
DELETE /VARIABLE AA
```

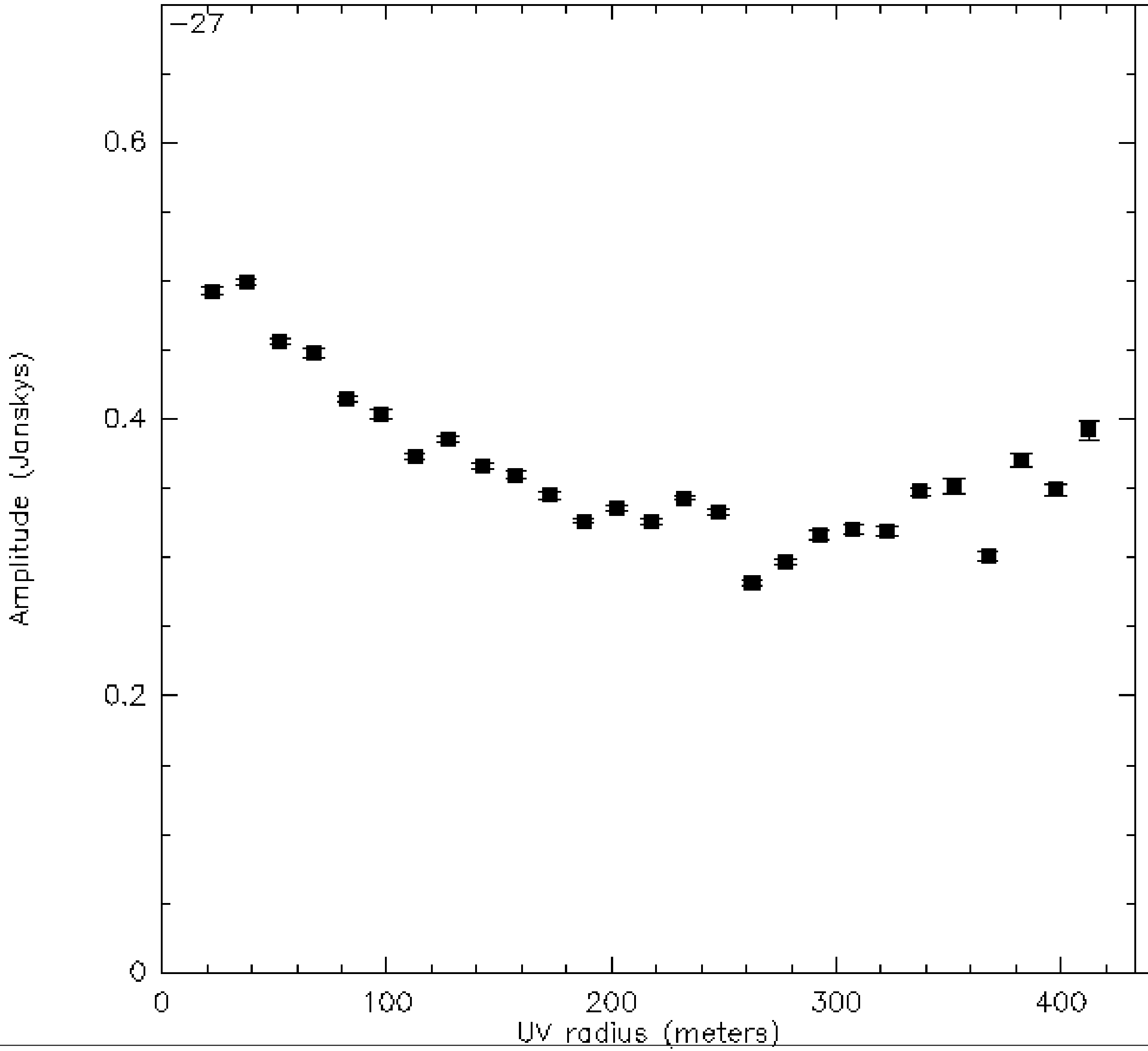
Circular averaging: UV_CIRCLE

- **Each visibility has a very low signal-to-noise** ⟶ difficult to see any signal
  - must average data in uv plane
  - or compute a map ($\sim$ weighted average)

- `UV_CIRCLE` **averages visibilities in rings in the** uv **plane**
  - only meaningful for sources with **circular symmetry**
  - **the source position must coincide with the phase center**, otherwise the amplitude of the visibility average on long spacings will decrease

- Output has the format of a uv table (but all v's are zero), may be plotted with `UVALL`


-27
0.8
0.6
0.4
0.2
Amplitude (Janskys)
0
100
200
300
400
UV radius (meters)


-27
0.6
0.4
0.2
0
Amplitude (Janskys)
0
100
200
300
400
UV radius (meters)

Model fitting

- Assume a **simple source model** that can be describes **analytically** with a limited number of parameters (source position, flux, sizes)
- Fit the FT of that model to the visibility data $\longrightarrow$ **avoid any artefact introduced by imaging/deconvolution process**
- A linear combination of several source models may be used
- Task **UV_FIT** knows
  - **point source**
  - **circular and elliptic gaussian**
  - circular and elliptic disk
  - ring
  - exponential and power laws

Model fitting

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**

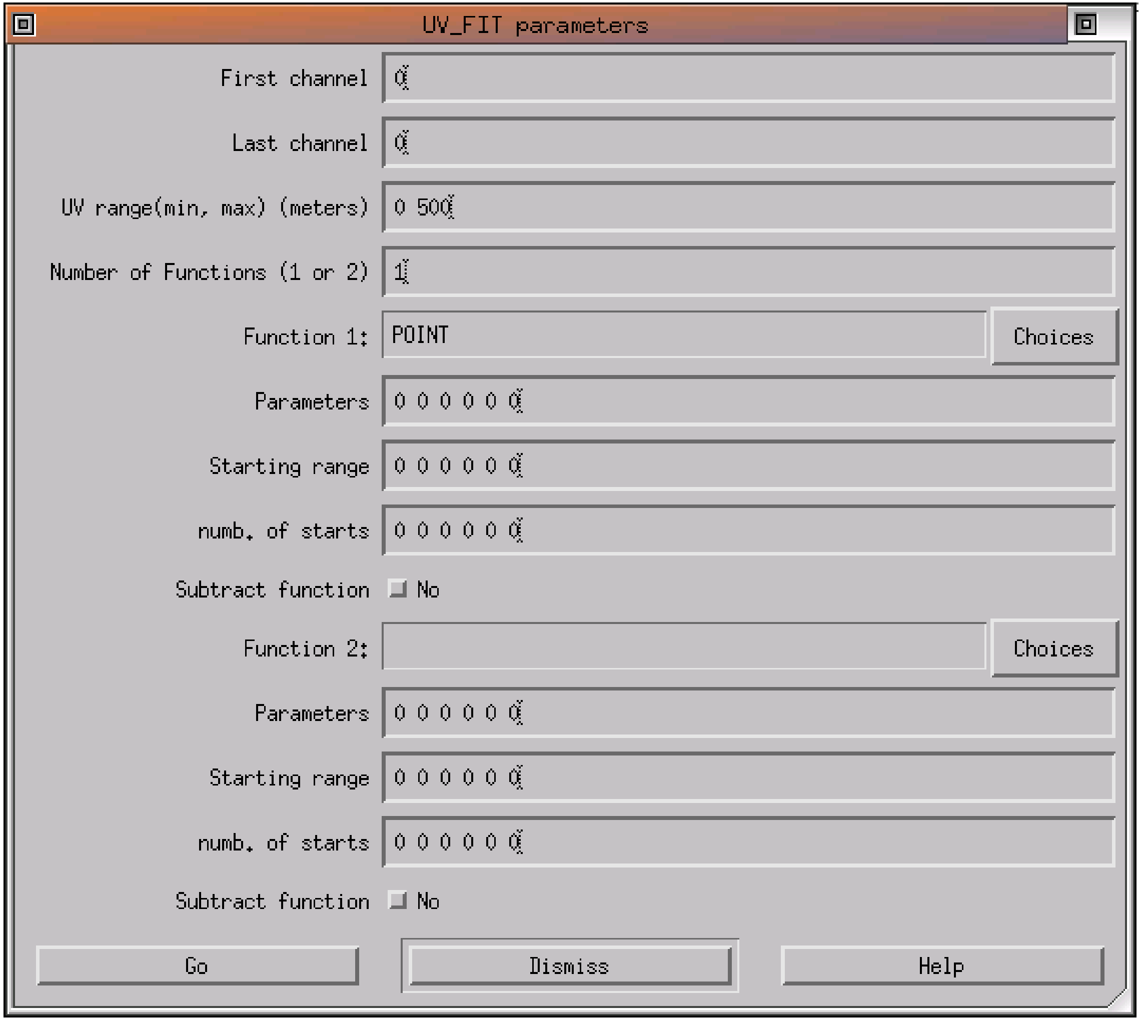
Parameters:

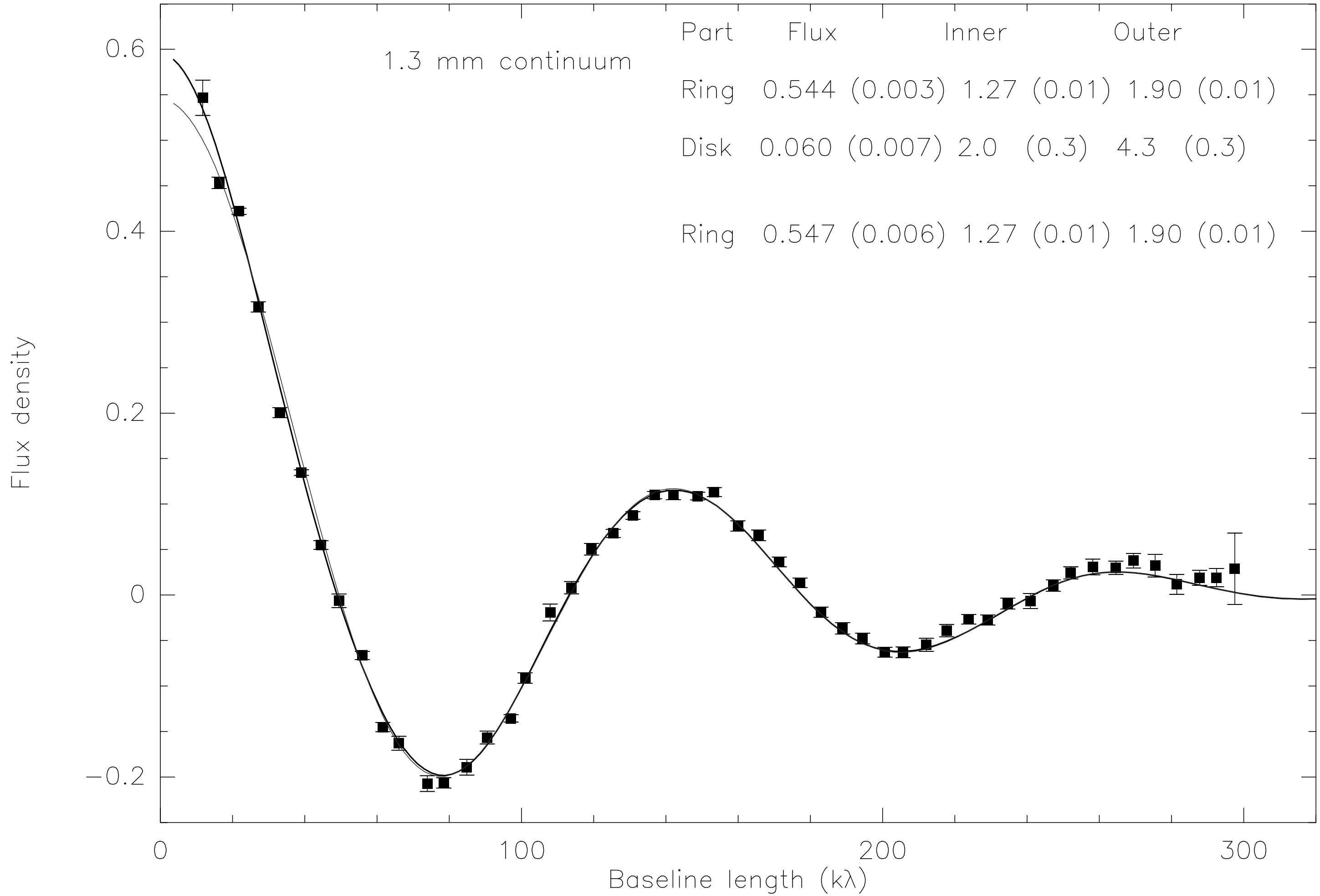
x_0	RA position
y_0	DEC position
S	Source flux
b	HP size

Variables:

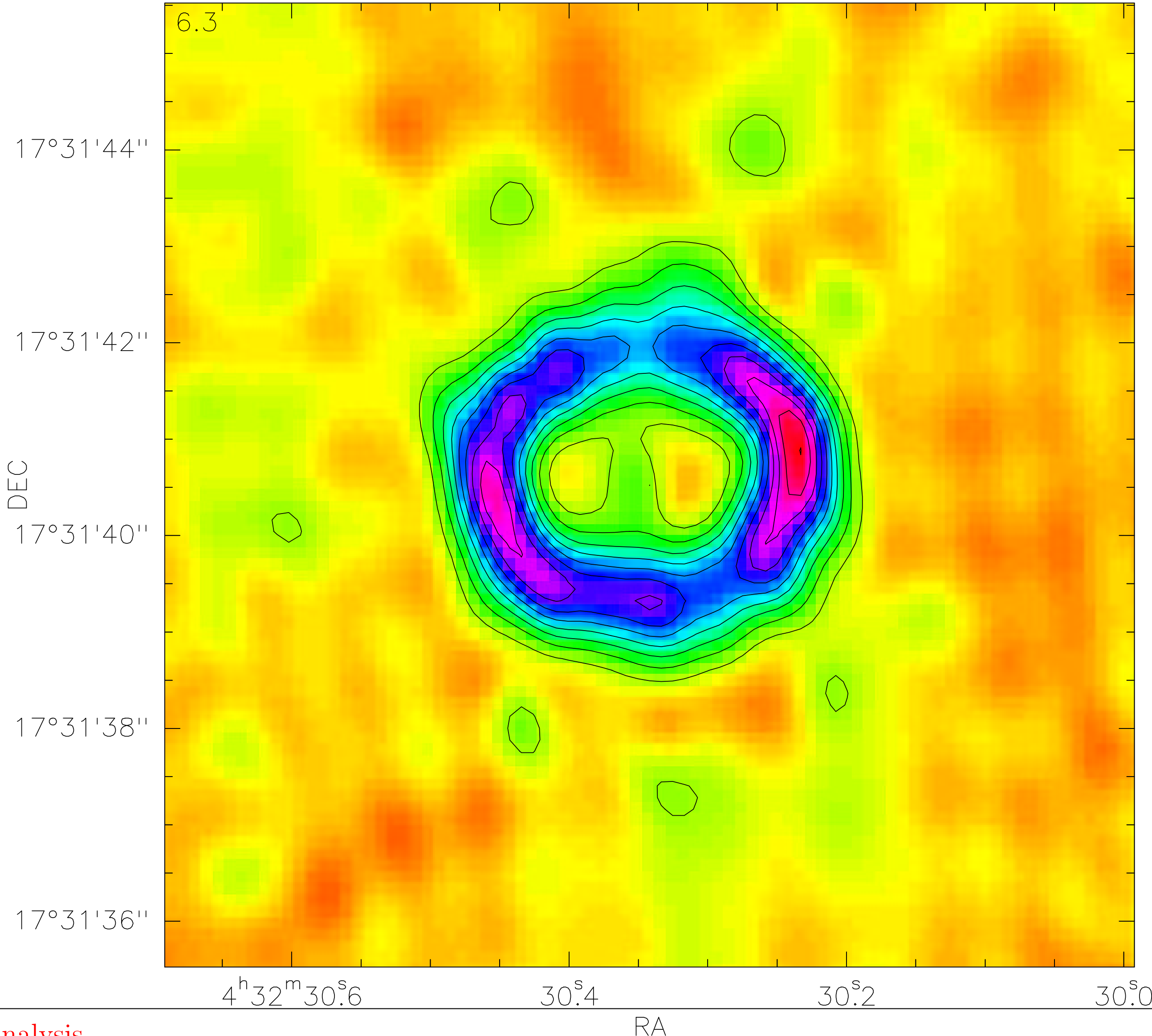
x, y	sky position
r	$\sqrt{(x-x_0)^2+(y-y_0)^2}$
u, v	projected spacing
q	$\sqrt{u^2+v^2}$

Name	Model	Visibility
Point	$S\ \delta(x-x_0, y-y_0)$	$S\ e^{-2i\pi(ux_0+vy_0)}$
Gaussian	$\frac{4S}{\pi b^2 \log 2}\ e^{-(4\log 2)\frac{r^2}{b^2}}$	$S\ e^{-\frac{\pi^2}{4\log 2}\ (bq)^2}$
Disk	$\frac{4S}{\pi b^2}$ where $\lvert r\rvert < b$	$2S\frac{J_1(\pi bq)}{\pi bq}$
Ring	$\frac{S}{2\pi\ b}\ \delta(r-b/2)$	$S J_0(\pi bq)$


UV_FIT parameters
First channel
0
Last channel
0
UV range(min, max) (meters)
0 500
Number of Functions (1 or 2)
1
Function 1:
POINT
Choices
Parameters
0 0 0 0 0 0
Starting range
0 0 0 0 0 0
numb. of starts
0 0 0 0 0 0
Subtract function
No
Function 2:
Choices
Parameters
0 0 0 0 0 0
Starting range
0 0 0 0 0 0
numb. of starts
0 0 0 0 0 0
Subtract function
No
Go
Dismiss
Help



Model fitting: RING (GG Tau)


6.3
17°31'44''
17°31'42''
17°31'40''
17°31'38''
17°31'36''
DEC
4h32m30.6s
30.4s
30.2s
30.0s
RA

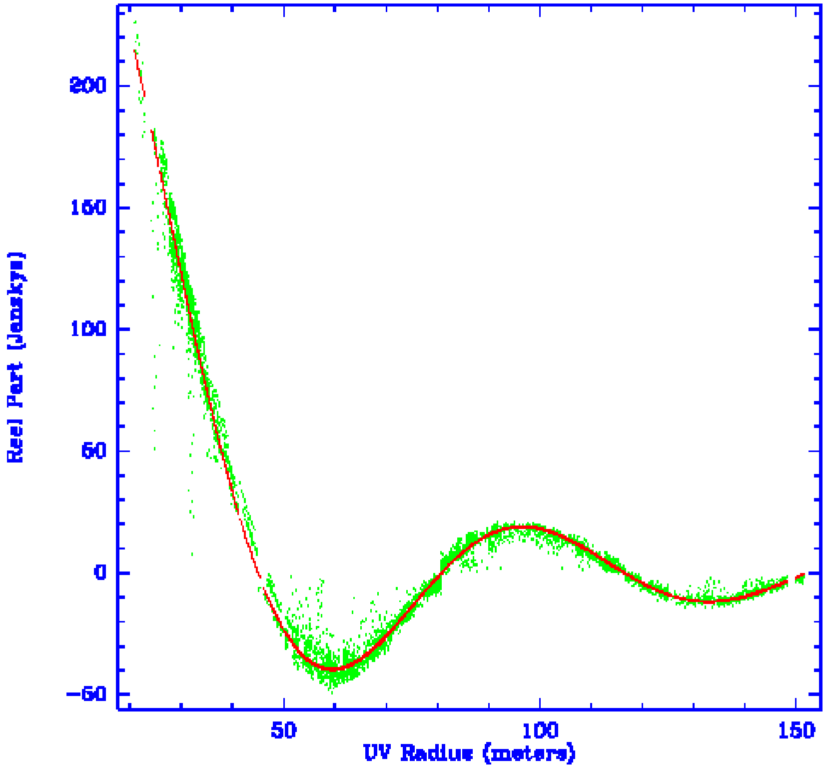
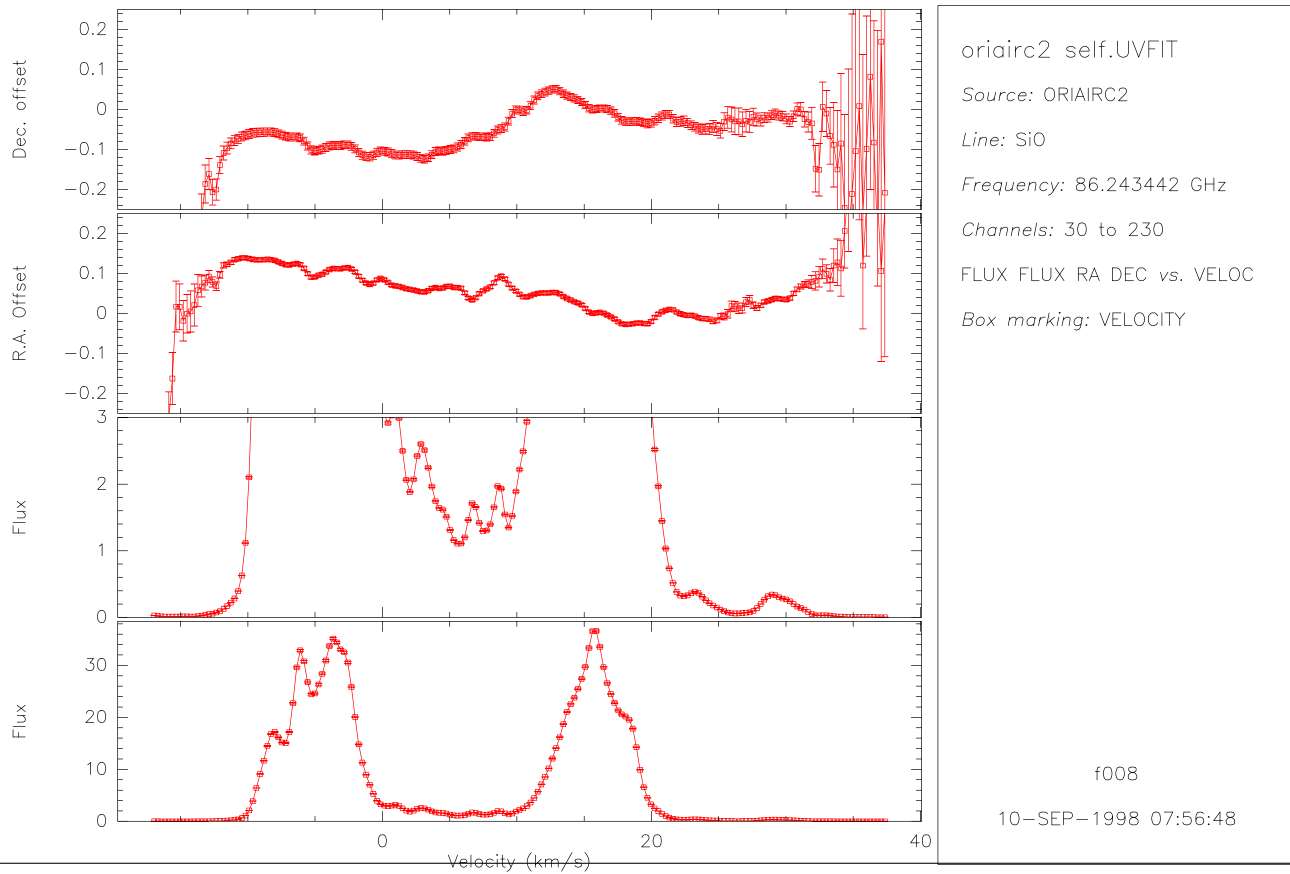


Figure 5: Real part of the CO(1 0) continuum visibilities observed at the IRAM Plateau de bure Interferometer after applying a self-calibration procedure and Mars continuum visibilities model (red points)

Model fitting: DISK (Mars)

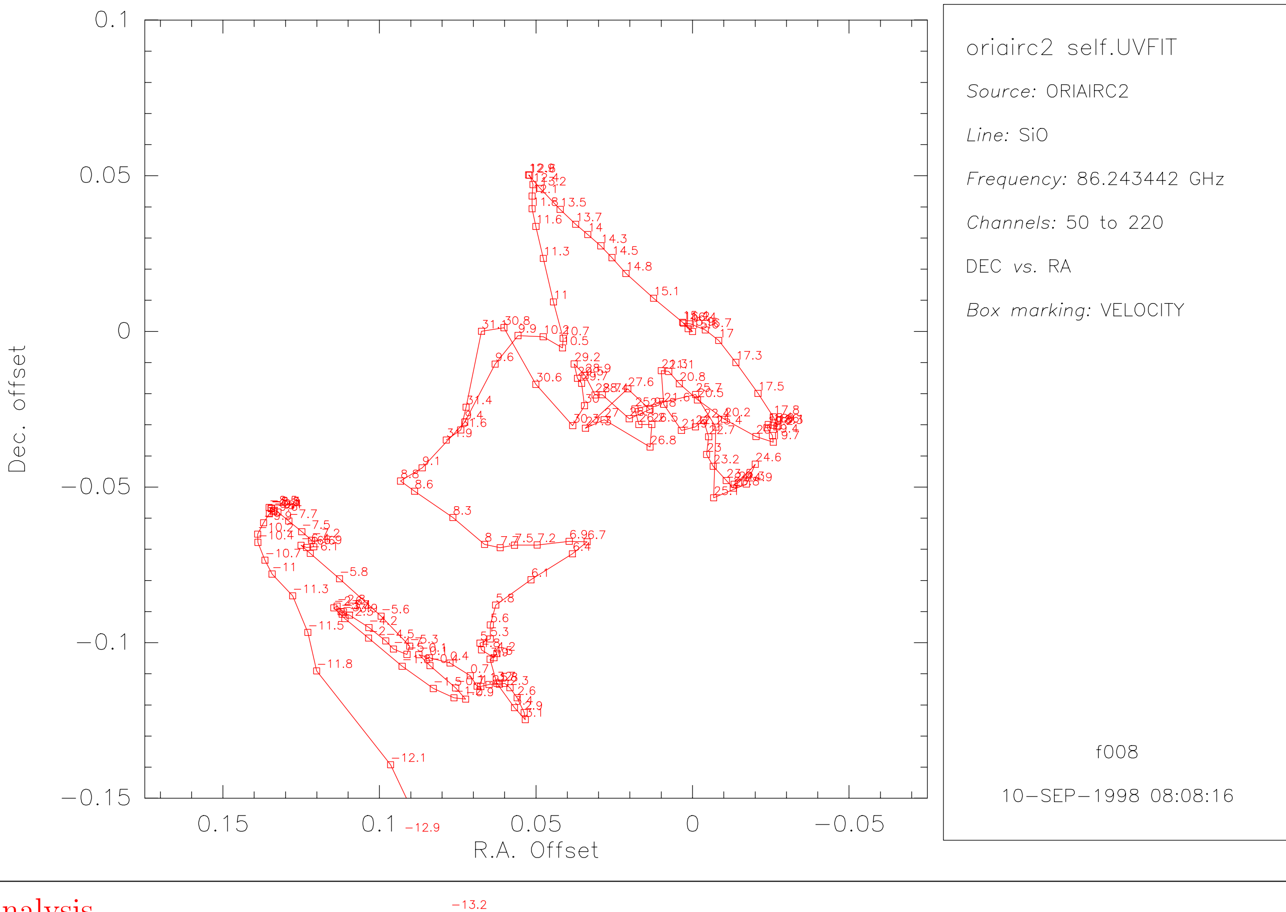
Model fitting: POINT SOURCE

Multi-channel point source fit to the Orion SiO maser:



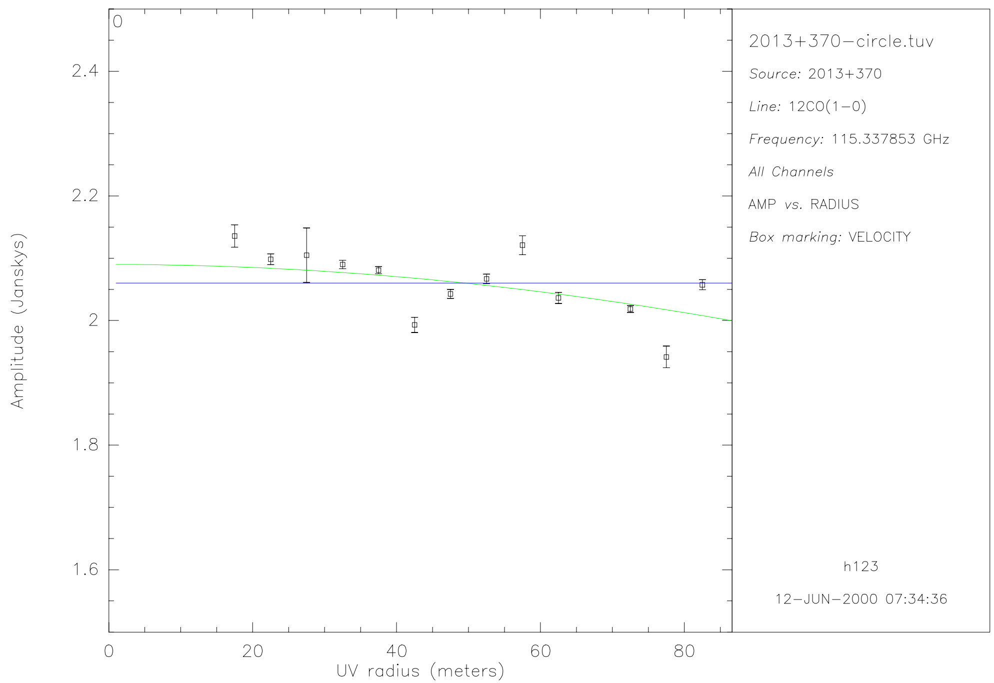
Model fitting: POINT SOURCE

Multi-channel point source fit to the Orion SiO maser:



Model fitting: POINT SOURCE flux measurement

- Possible **bias**: decorrelation from residual phase noise (e.g. on long baselines) will decrease the flux – this is the radio **seeing**
- Solution: fit a Gaussian (or compute a map)



Model fitting

```
I-UV_FIT,  420 data points for channel 1
I-UV_FIT, Starting minimization on channel 1   Velocity=  .0
I-UV_FIT, Starting from   .00000E+00  .00000E+00  2.0000
r.m.s.=      .7263 Jy.
POINT    R.A.        =     -.01289 (   .02275)  20:15:28.7089
POINT    DEC.        =     -.06380 (   .01770)  37:10:59.6262
POINT    FLUX        =     2.05698 (   .01754)
S-UV_FIT,  Successful completion

I-UV_FIT,  420 data points for channel 1
I-UV_FIT, Starting minimization on channel 1   Velocity=  .0
I-UV_FIT, Starting from   .00000E+00  .00000E+00  2.0000      1.0000
r.m.s.=      .7256 Jy.
C_GAUSS  R.A.        =     -.01276 (   .02280)  20:15:28.7089
C_GAUSS  Dec.        =     -.06363 (   .01783)  37:10:59.6264
C_GAUSS  Flux        =     2.09275 (   .03266)
C_GAUSS  F.W.H.P.    =      .68809 (   .26371)
S-UV_FIT,  Successful completion
```

Continuum source subtraction

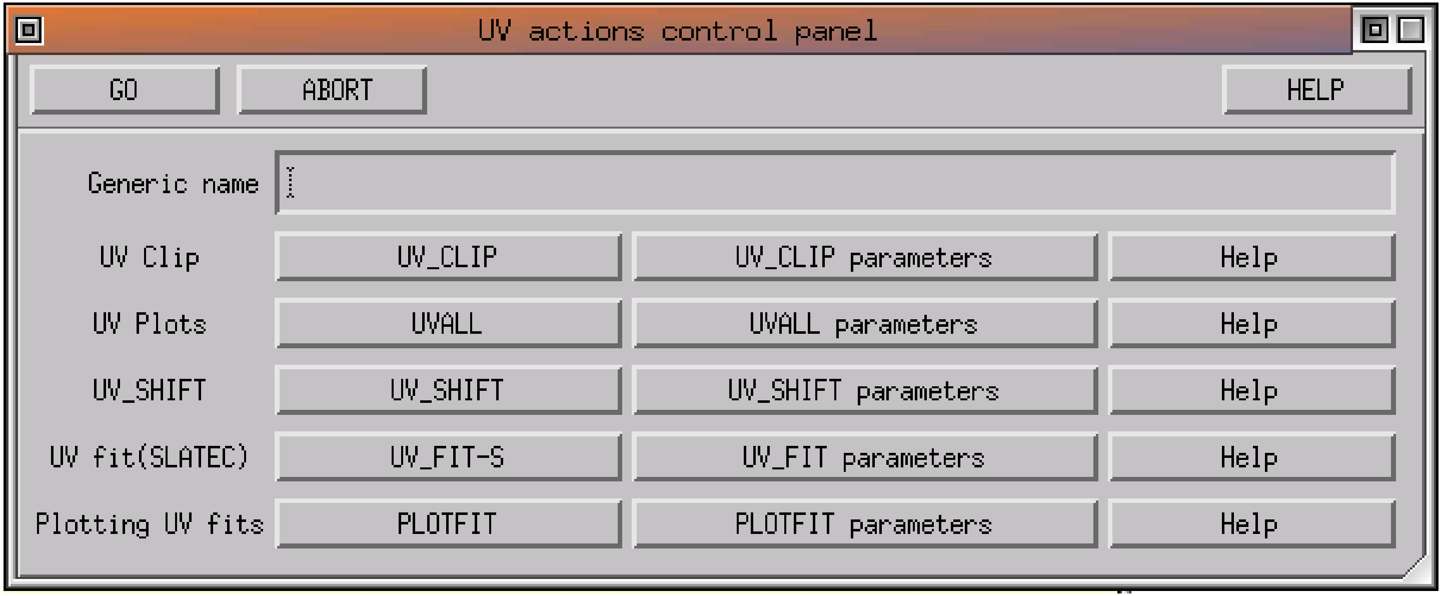
It is better to **subtract the continuum in the uv plane** to avoid error amplification due to the **non-linearity of the deconvolution**

- To subtract a point source at the phase center in the uv data: simply subtract a real number (the source flux) from all visibilities
- The task `UV_SUBTRACT` subtracts a time-averaged continuum uv table from a spectral line table observed simultaneously
  - If the source to be subtracted 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
- The task `UV_MODEL` can be used to build a uv table (from a model or another image) with the same uv coverage as the observations

Spectra of a point source

- Just compute a data cube and extract a spectrum
- Point source $\longrightarrow$ visibilities must be independant on baseline
  - amplitude = source flux
  - phase = zero
- **Just sum all visibilities!**
- `CLIC` can write spectra that can then be read by `CLASS`
- This is implemented in "self-calibration" procedure (often used for line absorption in spectra of strong quasars)
- Caution: **only meaningful for a point source located at the phase center**

MAPPING: Interferometric UV operations menu



uv_applyphase
uv_ascal
uv_atm
uv_average
uv_cal
uv_ccmodel
uv_cct
uv_center
uv_circle
uv_clip
uv_compress
uv_cuts
uv_dft
uv_extract
uv_fidelity
uv_fit-s
uv_flag
uv_fmodel
uv_gain
uv_hanning
uv_hybrid
uv_list
uv_map
uv_mcal
uv_merge
uv_mflag
uv_model
uv_mult
uv_noise
uv_observe
uv_pointing
uv_selfcal
uv_shift
uv_short
uv_single
uv_sinusphase
uv_solve
uv_sort
uv_splitfield
uv_stat
uv_subtract
uv_table
uv_timeaverage
uv_timebase
uv_track
uv_track_phase
uv_zero