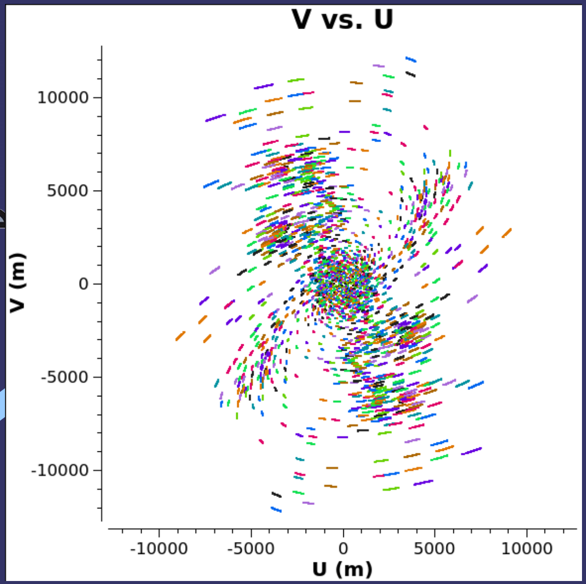
Fitting visibility data with UVMultiFit



Sébastien Muller

Onsala Space Observatory, Nordic ARC
Department of Space, Earth and Environment
Chalmers University of Technology, Sweden

I-TRAIN #3 - 2021 January 15

CHALMERS

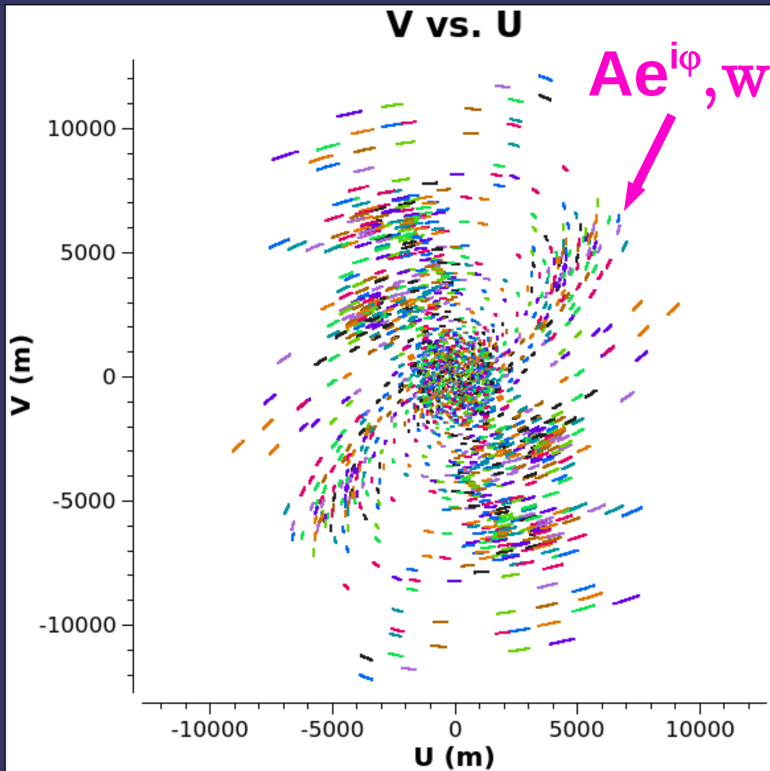
EUROPEAN ARC
ALMA Regional Centre || Nordic

Outline

- When and why using visibility fitting ?
- UVMultiFit a versatile tool for visibility fitting
- How to use it?

  Examples

  Live demo on a real ALMA dataset
- Q&As



FOURIER PLANE

Minimum baseline = 100 m
Maximum baseline = 13 km

Sparse coverage (~20 min on source)

At each u-v coordinate point,
a measurement of the complex visibility
(Amplitude, Phase, Weight)

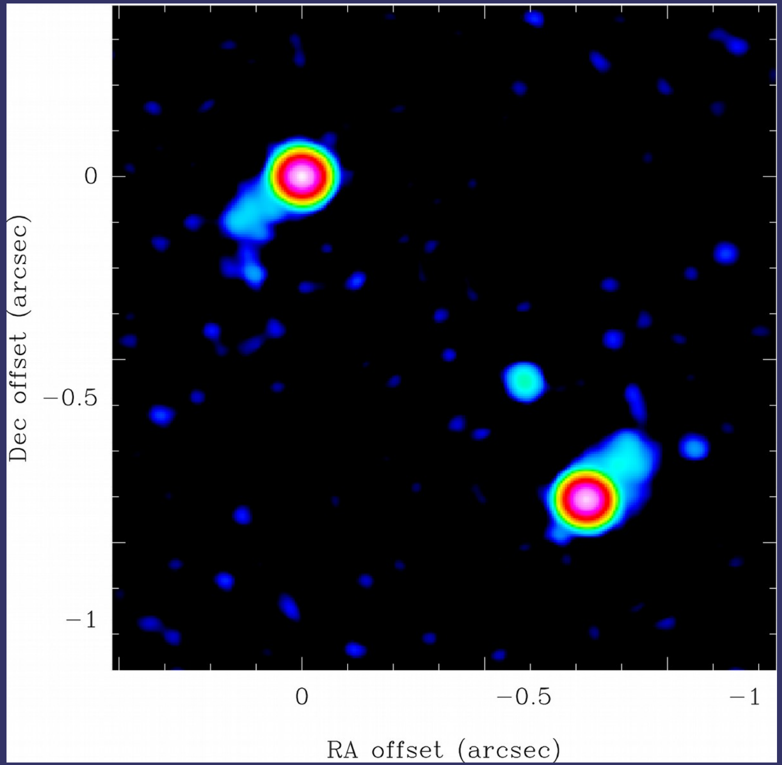
FT

IMAGE PLANE

CLEAN deconvolution
= interpretation (or model) of visibility data
(possible artefacts, sidelobes)

Beam resolution: 50 mas
Field of view: 40 arcsec

At each pixel, the intensity
Sensitivity 0.2 mJy/beam



What is uv-fitting?

Least-square minimization between **observed visibilities** and the corresponding **visibilities of a source model**

$$\chi^2 = \sum_{\text{over all visibilities}} \left(\frac{V_{obs} - V_{mod}}{\sigma} \right)^2$$

over
all
visibilities

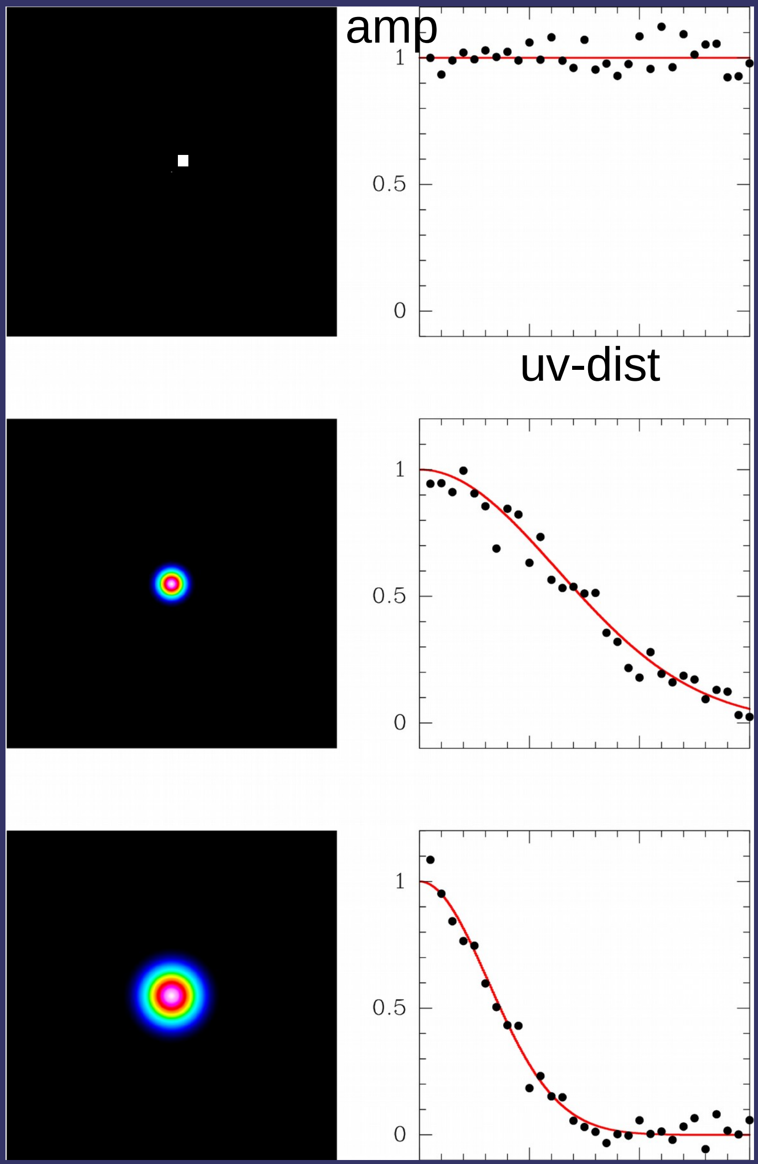
i.e., working in the Fourier plane

Image plane to Fourier plane

Point source

Narrow Gaussian

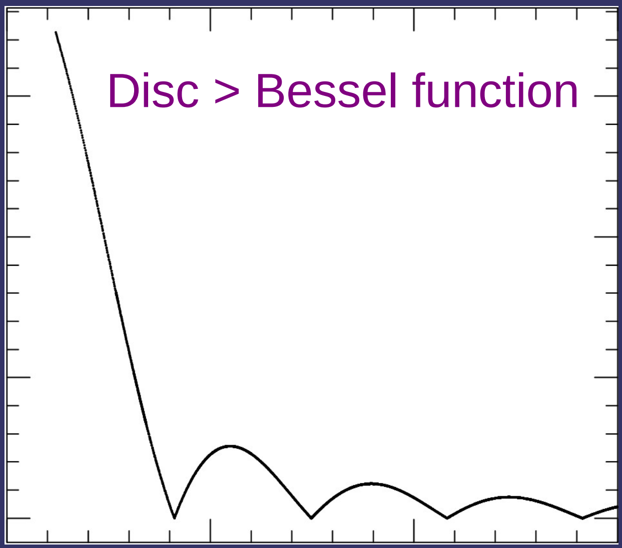
Broad Gaussian

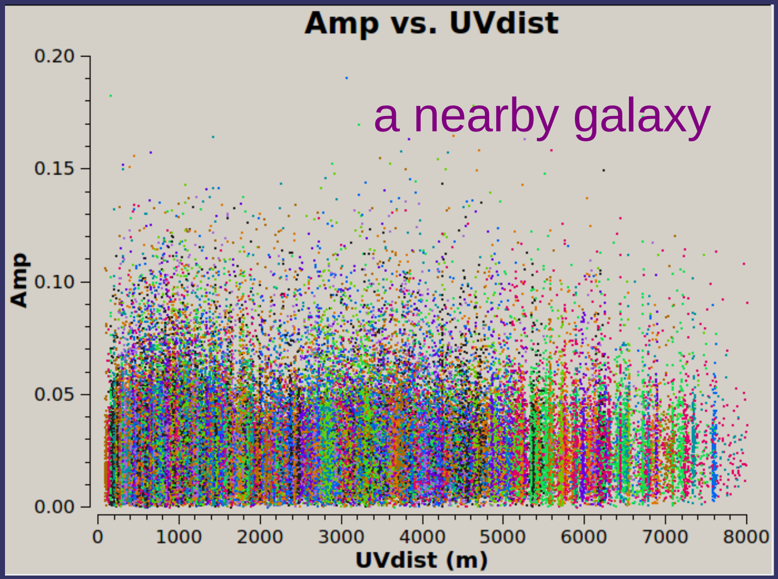


Constant amplitude

Broad Gaussian

Narrow Gaussian





When to uv-fit or to image ?

- Do imaging:
  - First look of a unknown source
  - Complex sources, with lots of structures
- Do uv-fitting:
  - For simple (analytical) sources
  - If you have good reasons from prior source knowledge (morphology, type, observations at other λ or better resolution)
  - Source with size close to or smaller than the beam resolution and observed with good SNR
  - Sparse uv-coverage, poor calibration

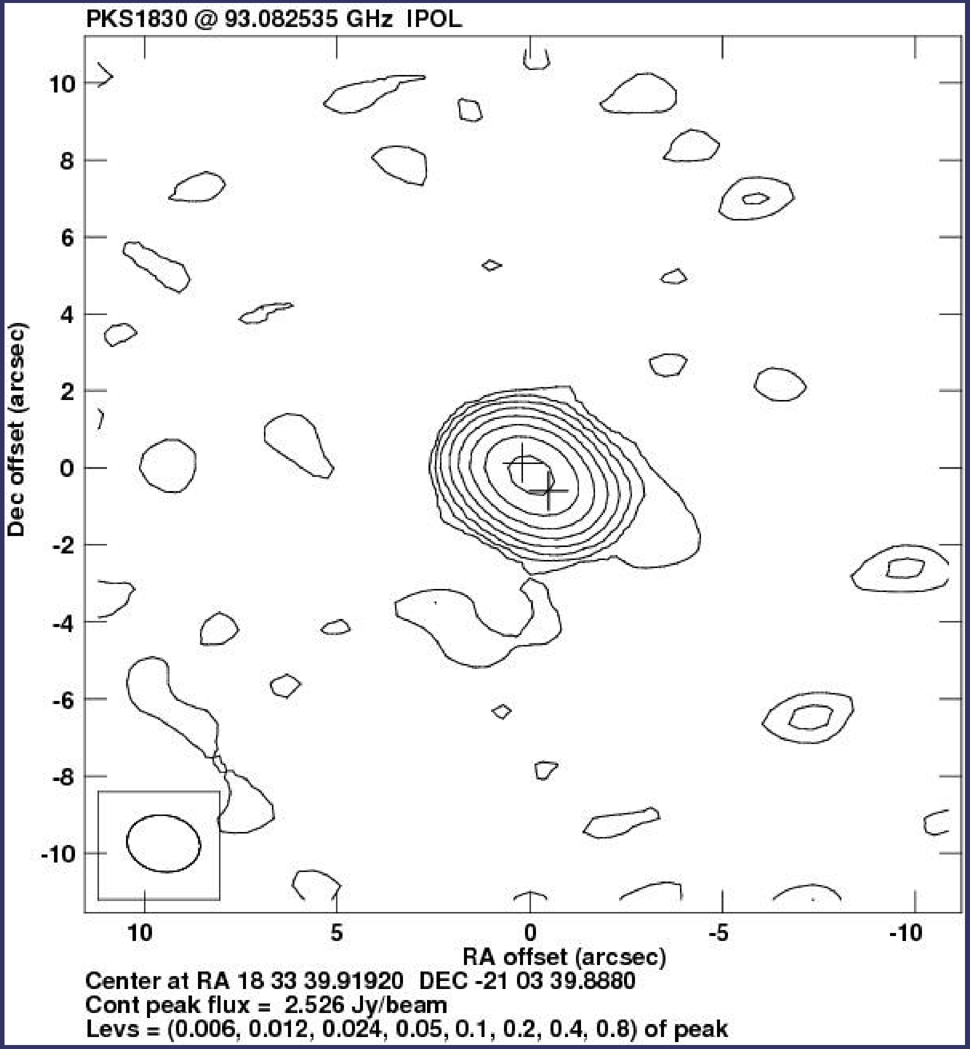
Why uv-fitting ?

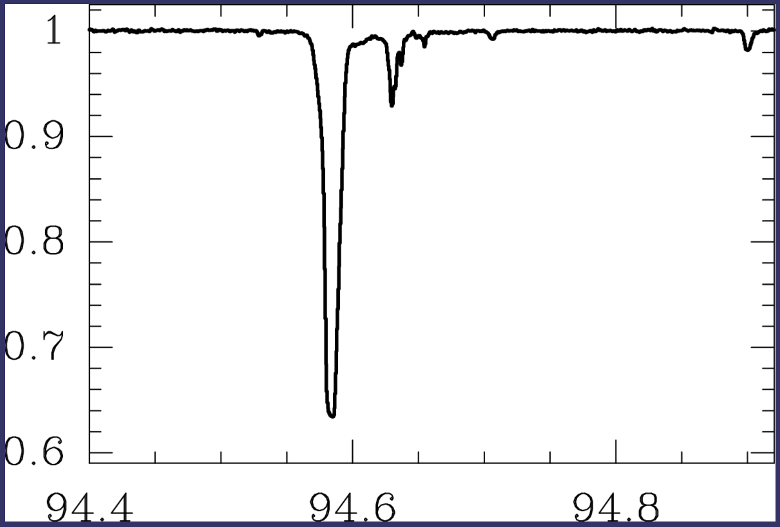
- uv plane: domain in which measurements are done
- Quick estimate of source parameters (e.g., to further self-cal)
- Robust handling of parameter uncertainties
- Remove sources of potential problems in the deconvolution/imaging process (next slide)

Behind imaging

- Gridding (pixel size)
- Weighting scheme (natural, uniform)
- Non-linear deconvolution process (CLEAN)
- Convolution with a clean beam (to be deconvolved if you want to measure a size of a source with size ~ beam)
- Deconvolution artefacts (sidelobes, missing flux)
- Systematics in frequency, epochs, configurations, instruments
- Image pixels within the beam are correlated (correlated noise)
- Image = non-unique interpretation of visibility data

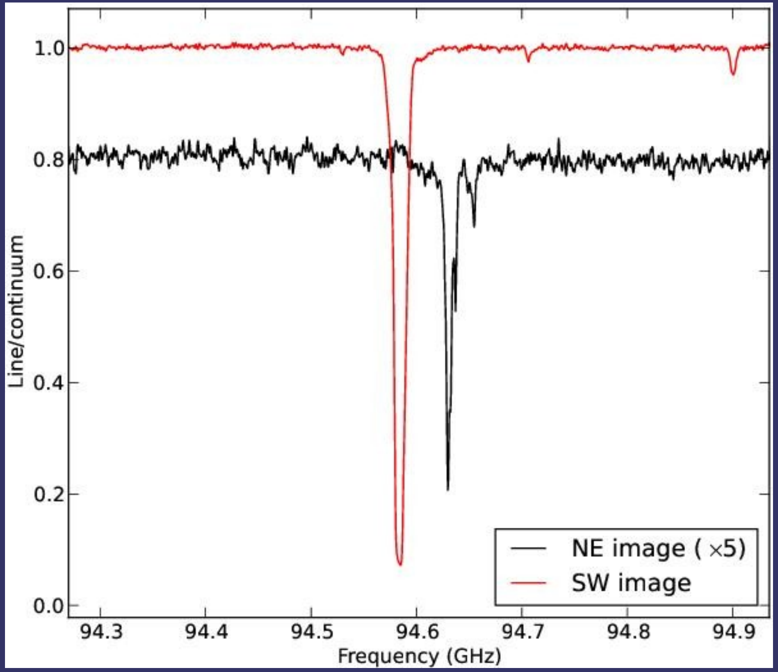
Example of uv-fitting over imaging





Spectrum out of the CLEAN image, central pixel

uv-fit with a two point source model



UVMultiFit in a nutshell

- Developed by Ivan Marti-Vidal (Marti-Vidal et al 2014)
- A versatile library for fitting visibility data
- Python-based and scriptable
- Many analytical models available
- Indefinite number of source components
- Possibility to tie parameters
- A generic frequency dependence is allowed

(e.g., spectral index measurements)

Installation

- Nordic ARC github repository:

  https://github.com/onsala-space-observatory/UVMultiFit

  for up-to-date instructions (Linux, MacOS)

  Contact the Nordic ARC for help !

Basic call to UVMultiFit

```
CASA>import uvmultifit as uvm

CASA>myuvfit = uvm.uvmultifit(
```

Data selection

```
vis=’mydata.ms’, spw=’0,1:10~100’,
column=‘data’, scans=[], field=’mysource’, ...
```

(Averaging)

```
timewidth=1, chanwidth=1,
```

Model & variables

```
model=[‘delta’], var=[‘p[0], p[1], p[2]’],
p_ini=[0.0, 0.0, 1.0], ...
```

Outputs

```
write = ‘ model’,
outfile=’results.uvfit’ )
```

Available models

Model	>	Variables
delta	>	RA, Dec, Flux
disc	>	RA, Dec, Flux, Major, Ratio, PositionAngle
ring	>	RA, Dec, Flux, Major, Ratio, PositionAngle
Gaussian	>	RA, Dec, Flux, Major, Ratio, PositionAngle
sphere	>	RA, Dec, Flux, Major, Ratio, PositionAngle
bubble	>	RA, Dec, Flux, Major, Ratio, PositionAngle
expo	>	RA, Dec, Flux, Major, Ratio, PositionAngle
power-2	>	RA, Dec, Flux, Major, Ratio, PositionAngle
power-3	>	RA, Dec, Flux, Major, Ratio, PositionAngle
GaussianRing	>	RA, Dec, Flux, Major, Ratio, PositionAngle, Sigma

And combinations e.g., model = [‘delta’, ‘Gaussian’]

FT is linear !

Some model definitions and their FT

Delta:

$$f(x,y) = \delta(x,y),$$
$$F(u,v) = 1.$$

Gaussian:

$$f(r) = \frac{1}{\sqrt{\pi/4\ln 2}a} \exp\left(\frac{-4\ln 2\, r^2}{a^2}\right)$$
$$F(\rho) = \exp\left(\frac{-(\pi a \rho)^2}{4\ln 2}\right),$$

Uniformly bright disc:

$$f(r) = \begin{cases} 4/(\pi a^2), & \text{if } r \le a/2 \\ 0, & \text{otherwise} \end{cases}$$
$$F(\rho) = \frac{2J_1(\pi a \rho)}{\pi a \rho},$$

Ring:

$$f(r) = \frac{1}{\pi a}\delta(r - a/2),$$
$$F(\rho) = J_0(\pi a \rho),$$

expo: $f(r) \propto \exp(-r/r_0)$

power-2: $f(r) \propto 1/(r^2+r_0^{\ 2})$

power-3: $f(r) \propto 1/(1+(2^{2/3}-1)(r/r_0)^2)^{3/2}$

Optically thin sphere:

$$f(r) = \begin{cases} 6/(\pi a^2)\sqrt{1-(2r/a)^2}, & \text{if } r \le a/2 \\ 0, & \text{otherwise} \end{cases}$$
$$F(\rho) = 3\sqrt{\pi/2}\,J_{3/2}(\pi a \rho)(\pi a \rho)^{-3/2}$$
$$= \frac{3}{(\pi a \rho)^3}\left[\sin(\pi a \rho) - \pi a \rho \cos(\pi a \rho)\right],$$

Variable meaning

- RA, Dec: shifts w.r.t. phase center (in arcsec)
- Flux: total flux density of the component (in Jy)
- Major: diameter along the major axis (in arcsec)
- Ratio: size ratio between the reference axis and the other axes
- Position angle: angle of the reference axis, from North to East (in deg.)
- Sigma: auxiliary variable for size-like parameter (e.g., width of the GaussianRing model)

Example 1: Fitting a point source with UVMultiFit

```
model = ['delta']

var = ['p[0], p[1], p[2]']
```

RA & Dec offsets and Flux density

Additional inputs:

```
p_ini = [ 0.0, 0.0, 1.0 ]     # guess values

bounds = [ [-1,1], [-1,1], [0,None]]
```

Useful way to measure the noise level for a non-detection:

```
var = [‘0.0, 0.0, p[0]’]
```

Example 2: Fitting two point sources

```
model = ['delta', 'delta']

var = [ 'p[0], p[1], p[2]', 'p[3], p[4], p[5]' ]
```

Or with tied parameters:

```
var = [ 'p[0], p[1], p[2]',
        'p[0]+p[3], p[1]+p[4], p[2]/p[5]'  ]
```

i.e., attaching the two point sources by fitting their separation

(ΔRA, ΔDec) = (p[3], p[4]) and flux ratio p[5]

Example 3: Fitting a spectral index

$F = F_0(\nu/\nu_0)^{-\alpha}$

```
model = ['delta']
var = ['0.0, 0.0, p[0]*(nu/100.e9)**p[1]']
p_ini = [ 1.0, -0.7]
```

p[0]: Flux density at 100 GHz

p[1]: Spectral index

(Need good SNR and/or large frequency coverage)

Handling of channel selection

Parameter OneFitPerChannel

- =True : one fit to each individual spectral channel (spectral mode)
- =False : fit to the whole channel selection at once (continuum mode)

Fit output #1

```
CASA>myuvfit = uvm.uvmultifit(...,
                    outfile = ‘filename.txt’)
```

- Output results in an ascii file ‘outfile’
- Create a Python dictionary myuvfit.result, containing:

```
FitValues = myuvfit.result[’Parameters’]
FitErrors = myuvfit.result[’Uncertainties’]
ReducedChiSquare = myuvfit.result['Reduced Chi squared']
```

among other information (frequency, degrees of freedom, ...)

Fit output #2

- To store directly into the measurement set:

  write = ‘’ : don’t write anything

  write=’model’: best-fit model saved into the **model** column

  write = ‘residuals’ : fit residuals save in the **corrected** column

  timewidth and chanwidth parameters need to be set to 1

  Be careful to use the same data selection when further processing

!!! Best practice: work on a copy of your measurement set

(Otherwise, it will over-write the model/corrected columns)

Fit the ‘data’ column

Need to create a model column if it doesn’t exist (e.g., with setjy)

Least square minimization

Choose a **correct** model: look at the visibilities (amp. vs uv-distance), use prior knowledge of the target

$$\chi^2 = \sum_{\text{over all visibilities}} \left(\frac{V_{obs} - V_{mod}}{\sigma} \right)^2$$

Reduced $\chi^2 = \chi^2 / (N-P)$

Best-fit parameter uncertainties estimated from the Jacobian matrix, scaled so that the reduced $\chi^2 = 1$

Judging the goodness of fit

- Check best-fit result parameters and their uncertainties
- Check the reduced Chi square
- Check the fit residuals

  Plot amplitude vs uv-distance of residual visibilities

  Make a dirty image of residuals stored in the corrected column by UVMultiFit parameter write='residuals'

- Have a critical mind *vs*

  uv-coverage (B_{min} - B_{max}), calibration systematics, dynamic range limits, fit robustness, absolute flux accuracy, ...

- Make simulations (OST, simobserve) and uv-fit them

Some rules of thumb

- Position measurement accuracy:

$$\delta(\text{position}) \sim \text{BEAM\_FWHM} / 2.\text{SNR}$$

- Size measurement accuracy

$$\delta(\text{size}) \sim \text{BEAM\_FWHM} / \text{SNR}^{1/2}$$

See e.g., Marti-Vidal et al. 2012, A&A 541, 135

Resources

A&A 563, A136 (2014)
DOI: 10.1051/0004-6361/201322633

Astronomy & Astrophysics

UVMULTIFIT: A versatile tool for fitting astronomical radio interferometric data

I. Martí-Vidal, W. H. T. Vlemmings, S. Muller, and S. Casey

- Nordic ARC github

  https://github.com/onsala-space-observatory/UVMultiFit

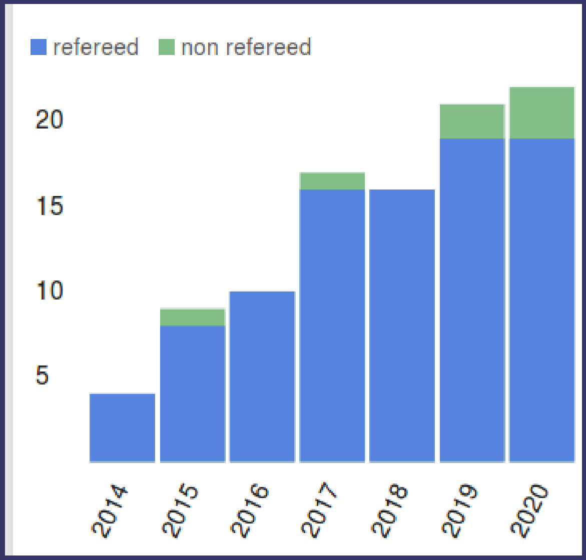
- Ivan's personal webpage and documentation

  http://mural.uv.es/imarvi/docums/uvmultifit/

- Test suite in the UVMultiFit distribution

Publications using UVMultiFit

Word cloud for titles of papers citing UVMultiFit



- 100 citations in the ADS as of 14 Jan. 2021
- increasingly being used
- covering a large range of science targets

Live demo

- Fitting a model of two point sources to real ALMA observations of the lensed quasar PKS1830-211
- Fitting the visibilities of PKS1830-211 to extract the absorption spectrum against the two lensed images