Radio Astronomy Software

presented by
Victor de Souza Magalhães

07/07/2016
Summary

- GILDAS
- AIPS
- CASA
- MIRIAD
- Python
IRAM - GILDAS

- GILDAS - Grenoble Image and Line Data Analysis Software

  - Core:
    - SIC; GreG; pySic

  - Calibration:
    - MIRA; CLIC; MRTCAL

  - Data reduction:
    - MAPPING; CLASS

  - Preparation for observations:
    - ASTRO
SIC: Simple interpreter of commands

- It is the core of GILDAS:
  - Command line interpreter
  - Parser
  - Scripting language.
  - syntax similar to FORTRAN.
pySic

• It is the python binding to GILDAS.

• enables the python prompt inside GILDAS.

• Enables the use of python capabilities in GILDAS (functions, modules, plotting, list manipulation, etc).

• Enables the use of GILDAS functionalities in python (plotting, spectrum manipulation, data handling, etc).

• works through a python module called pySic.
MIRA | MRTCAL | CLIC

- MIRA is the software currently in use for the calibration of IRAM-30m data.
- MRTCAL is the software in development to replace MIRA, as faster, more reliable software.
- CLIC is the software for the calibration of IRAM-NOEMA data, it produces plots so you can follow what is happening during the pipelines.
GreG - Grenoble Graphic

- It is a very powerful plot generator.
- It can be called from the other software inside GILDAS.
- Can produce very beautiful plots.
- Can even produce interactive graphics.
CLASS

- It is a subset of GILDAS fine tuned for spectral line analysis.
- It contains tools for:
  - Line identification
  - Line fitting
  - Visualisation of spectra
A typical CLASS session
MAPPING

- Reduction of interferometric data.
- UV plane analysis.
- Dirty images.
- Deconvolution.
- Comprehensive plots to understand the data
h13cn sub corr ave.tuv

Source: mwc 4800

Line: H13CN3–2

Frequency: 259.042878 GHz

Channels: 1 to 1

v vs. u

Box marking: VELOCITY
h13cn sub corr ave.tuv

Source: mwc 4800
Line: H13CN3–2
Frequency: 259.042878 GHz
Channels: 1 to 1
amp vs. radius
Box marking: VELOCITY

Amplitude [Jy]

UV radius [m]
Source: TW Hya

Line:

Frequency: 340.031549 GHz
Vsys: 0 km/s LSR
Beam: 0.9 x 0.54 PA 86°
Level step: 100 mJY/BEAM

Box marking: VELOCITY
Channels: [292,315]

desouzav
05–JUL–2016 15:48:20
Help you prepare observations.
Check source visibility.
Prepare tuning setups for the observations.
Check UV coverage for interferometric observation.
Dealing with sources

Example 3: source visibility

ASTRO> TIME 0 15-AUG-2016
ASTRO> TYPE issum.sou

Source1 eq 2000 04:00:00 -05:00:00 lsr 0 FL 2 1
Source2 eq 2000 08:00:00  10:00:00 lsr 0 FL 1 0.1
Source3 eq 2000 12:00:00  25:00:00 lsr 0 FL 0.6 0.9
Source4 eq 2000 16:00:00  40:00:00 lsr 0 FL 0.5 0
Source5 eq 2000 20:00:00  55:00:00 lsr 0 FL 0.2 0.2

ASTRO> CATALOG issum.sou
ASTRO> HORIZON /SOURCE /PLANET

Source1         Sun distance   86.1   Avoidance 25-APR-2017 to 09-JUN-2017
Source2         Sun distance   24.5   Avoidance 19-JUN-2016 to 24-AUG-2016
Source3         Sun distance   34.9   Avoidance 18-AUG-2016 to 06-OCT-2016
Source4         Sun distance   85.0   No Avoidance
Source5         Sun distance  107.8   No Avoidance
Example 1: NOEMA single pointing case

ASTRO> TIME 0:0:0 15-AUG-2016
ASTRO> SOURCE MySource EQ 2000 12:00:00 24:24:24 LSR 0
ASTRO> UV_TRACKS W27 W09 E68 E23 E12 N46 N29 N20 /FRAME /HORIZON 40 /TABLE mytab.uvt

W27—W09—E68—E23—
—E12—N46—N29—N20

Frequency 100.0 GHz
Declination 24.3°
To know more

- IRAM provides a website to support GILDAS: http://www.iram.fr/IRAMFR/GILDAS/
- There are some tutorials on how to use GILDAS software.
- Documentation on all the tasks.
- And a Download area, where you can fetch GILDAS for yourself.
AIPS is a very old software from NRAO from the 1970s.

It is has new releases, basically annually.

It was created to handle radio interferometric data.

It was built to be fast.

It has been mainly superseded by CASA, but it is still faster than CASA.
CASA - Common Astronomy Software Applications

Its development started as a rewrite of AIPS into more modern software.

It is a GUI oriented software for radio astronomy data reduction.

It is the software with the ALMA calibration pipeline.

Can be scripted in Python, its prompt is actually a Python prompt, called casapy.
NRAO - CASA

- Holds all data treatment related functionality in one interface:
  - Calibration, reduction.
- Posses various data viewing GUIs, UV plane viewing, image viewing, data cube analysis.
- Some tasks are realised using a GUI, ex: Cleaning.
- Drawback: Some tasks can be very slow (ex: a few hours per cleaning).
INFO casa::casa ---
INFO casa::casa CASA Version 4.2.2 (prerelease r30986)
INFO casa::casa Tagged on: Thu, 21 Aug 2014

*** Loading ATNF ASAP Package...
*** ... ASAP (4.2.0a rev#30794) import complete ***

Major interface changes to SINGLE DISH tasks have been taken place in CASA 4.2.2 release

The interface of the following tasks are modified: sdbaseline, sdcal, sdcal2, sdfit, sdflag, sdgrid, sdimaging, sdmath, sdplot, sdreduce, sdsave, and sdstat. Additionally, a new task called sdaverage is available. Task sdsmooth has been incorporated in the new task and removed.

The tasks with old interfaces are available with name {taskname}old. They will be kept until CASA 4.3 release and removed from later releases. Users are advised to update existing scripts.

#-----------------------------------------------

CASA <2>:
To know more

- NRAO maintains a website for CASA: https://casa.nrao.edu

- There is also a good amount of tutorials (some with data to follow them) at: https://casaguides.nrao.edu/
Its is the software for the reduction of data from the Sub-Millimetre array.

It can be configured to be used with other observatories as well.

It integrates into the OS shell, no separate prompt.

Its scripts are Shell scripts.
Python itself has seen the development of Astronomy oriented modules.

These are not specific for radio astronomy but are very useful to make plots and do some data treatment or analysis.

Astronomy specific python modules:

- astropy | pyfits | aplpy | pyspeckit

Useful modules:

- numpy | scipy | matplotlib | pandas
```python
import aplpy
from matplotlib import pyplot as plt
radius = 0.0014  # Radius of the plots in degrees
fig = plt.figure()  # matplotlib figure to enable subplots
# Creating a subplot from the fits file
figcn = aplpy.FITSFigure('cnb_uvall.fits', figure=fig, subplot = [0.1,0.1,0.35,0.5])
figcn.show_colorscale()  # Displaying the fits file in color
figcn.add_beam()  # adding the beam size
xw, yw = figcn.pixel2world(133, 128)  # Getting the position of the center in WCS
figcn.recenter(xw, yw, radius)  # Recentering and resizing the plot
# Decreasing the precision of the WCS displayed on the plot
figcn.tick_labels.set_xformat('hh:mm:ss')
figcn.tick_labels.set_yformat('dd:mm:ss')
# Hiding RA(J2000) and DEC(J2000)
figcn.axis_labels.hide()
## Same as for CN
figc15n = aplpy.FITSFigure('c15nb.fits', figure=fig, subplot = [0.45,0.1,0.35,0.5])
figc15n.show_colorscale()
figc15n.add_beam()
figc15n.recenter(xw, yw, radius)
figc15n.tick_labels.set_xformat('hh:mm:ss')
figc15n.tick_labels.set_yformat('dd:mm:ss')
figc15n.axis_labels.hide()
# Overlaying the CN image as grey contours
figc15n.show_contour('cnb_uvall.fits', colors='grey')
### Hide the tick labels.
figc15n.tick_labels.hide()
figcn.save('CN_C15N.eps')```
$\chi^2$ map for:

$l1498-500\text{um-nw} + l1498-350\text{um-nw}$
Model:
Plateau = 54.0, α = 2.92, χ² = 0.0013