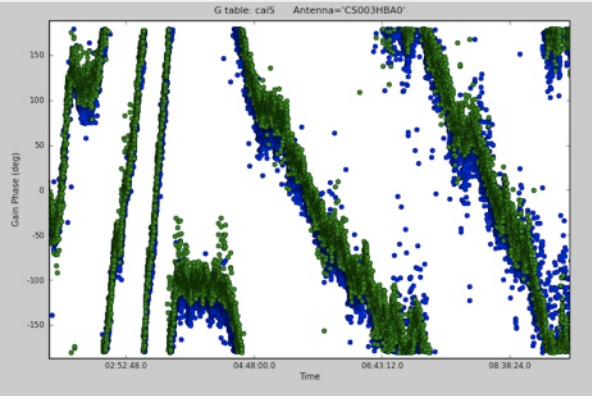
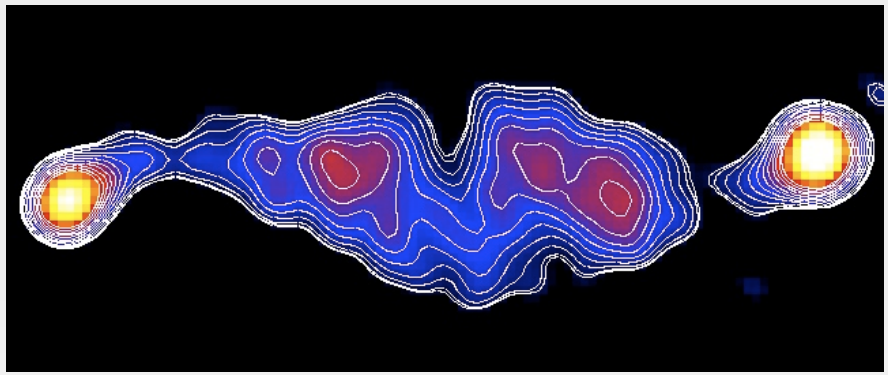
From raw visibilities to images



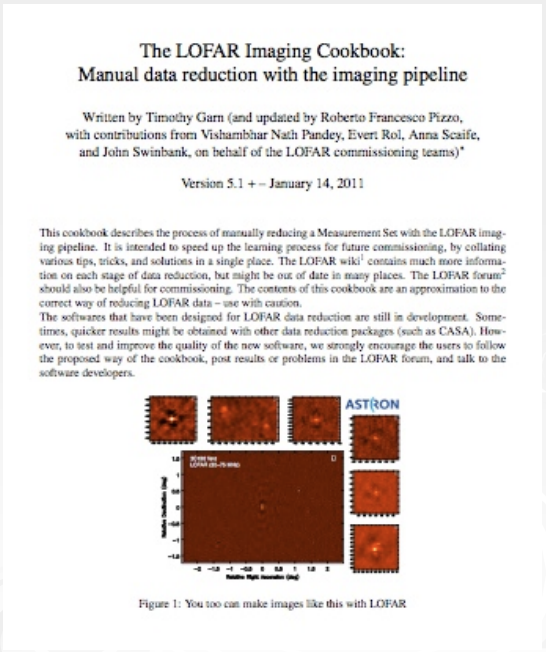


R. J. van Weeren
Leiden Observatory

Overview of processing

- 1. flagging (RFI console)
- 2. NDPPP (flagging, compression)
- 3. BBS (calibration)
- 4. imaging

(see the Cookbook)



The LOFAR Imaging Cookbook:
Manual data reduction with the imaging pipeline

Written by Timothy Garn (and updated by Roberto Francesco Pizzo, with contributions from Vishambhar Nath Pandey, Evert Rol, Anna Scaife, and John Swinbank, on behalf of the LOFAR commissioning teams)*

Version 5.1 + – January 14, 2011

This cookbook describes the process of manually reducing a Measurement Set with the LOFAR imaging pipeline. It is intended to speed up the learning process for future commissioning, by collating various tips, tricks, and solutions in a single place. The LOFAR wiki[1] contains much more information on each stage of data reduction, but might be out of date in many places. The LOFAR forum[2] should also be helpful for commissioning. The contents of this cookbook are an approximation to the correct way of reducing LOFAR data – use with caution.
The softwares that have been designed for LOFAR data reduction are still in development. Sometimes, quicker results might be obtained with other data reduction packages (such as CASA). However, to test and improve the quality of the new software, we strongly encourage the users to follow the proposed way of the cookbook, post results or problems in the LOFAR forum, and talk to the software developers.

ASTRON

Figure 1: You too can make images like this with LOFAR

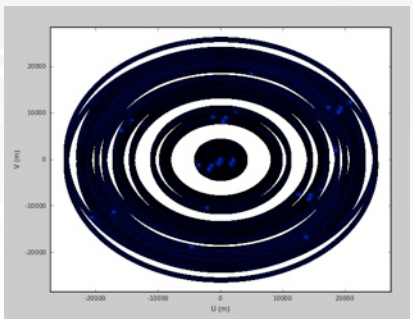
RFI console & NDPP

- dataset passes through RFI console and NDPP
- the good news: usually RFI is removed very efficiently
- averaging strategy often used:
  - from 256 (or 64) chan to 1 chan of 0.18 MHz
  - time averaging also often employed (i.e., from 1 sec to 3 sec)
- the bad news: bandwidth smearing issues

Before starting....

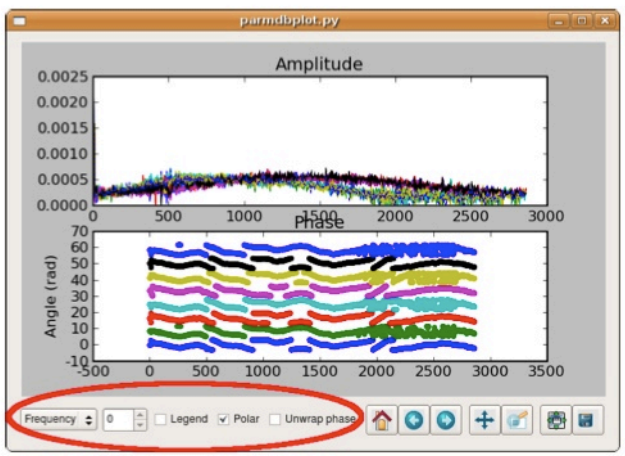
- check a few long/short baselines (plotxy, plotms), one baseline at the time
  - get an idea of the A-team effects
  - use BBS to “predict”

- What sources in the field center (FWHM of station beam)?
- How far away are the A-team sources?
- check uv-coverage, what are the longest baselines?
- as always: do not(!) forget the “clearcal” if you’re ever going to use casapy on the data (you probably will)



BBS

- monitor progress with (Strategy.ChunkSize = 100)
  `grep -i 'Reading chunk' ~/kernel_control.log| wc -l`
- ~~casa/aips/difmap selfcal~~ use BBS (some exceptions)
- always enable the station beam for `solving'
  Beam.Enable = T
- always check the solutions with parmdbplot.py (do they make sense?)
- two options for the "correct" step:
  - *Beam.Enable=T, for casa clean()*
  - *Beam.Enable=F, for the "vdtol" imager*



Bandwidth smearing

- affects A-team subtraction !!
- in LBA (some) A-team sources always cause “problems” at the short baselines (short baseline signal is not RFI)
- bandwidth smearing: also a problem in the central FOV, beyond the FWHM of the station beam
- also time smearing (if you time-average too much in time)

Table 11: Reduction in Peak Response Due to Bandwidth Smearing

$\frac{\Delta\nu}{\nu_0}\frac{\theta_0}{\theta_{HPBW}}$	Peak	Width
0.0	1.00	1.00
0.50	0.95	1.05
0.75	0.90	1.11
1.0	0.80	1.25
2.0	0.50	2.00

BBS solutions

if the solutions looks reasonable :

(if not, go back to BBS)

- manually remove high data points, flagdata(),
  - be careful (i.e., extended source)
- solution outliers: “solflag” (see cookbook)

3C196 LBA example

- 1. Cas A, at 60 deg (Cyg A > 90 deg)
- 2. simulate Cas A, Cyg A with BBS: conclude Cas A is the “problem”
- 3. skymodel contains Cas A & 3C196 (point)
  - Cas A dominates short baselines
  - 3C196 dominates long baselines > 2km
- use directional gain and solve 3C196 + Cas A, subtract Cas A

3C196 LBA example

- 1. Cas A, at 60 deg (Cyg A > 90 deg)
- 2. simulate Cas A, Cyg A with BBS: conclude Cas A is the “problem”
- 3. skymodel contains Cas A & 3C196 (point)
  - Cas A dominates short baselines
  - 3C196 dominates long baselines > 2km
- use directional gain and solve 3C196 + Cas A, subtract Cas A

- alternative approach:
  - skymodel only contains 3C196, but now limit uvrange > 2 km when solving
    - non-directional gain
    - you can add more source to the model close to 3C196
  - first approach gives better results (can use most of the short baselines also)

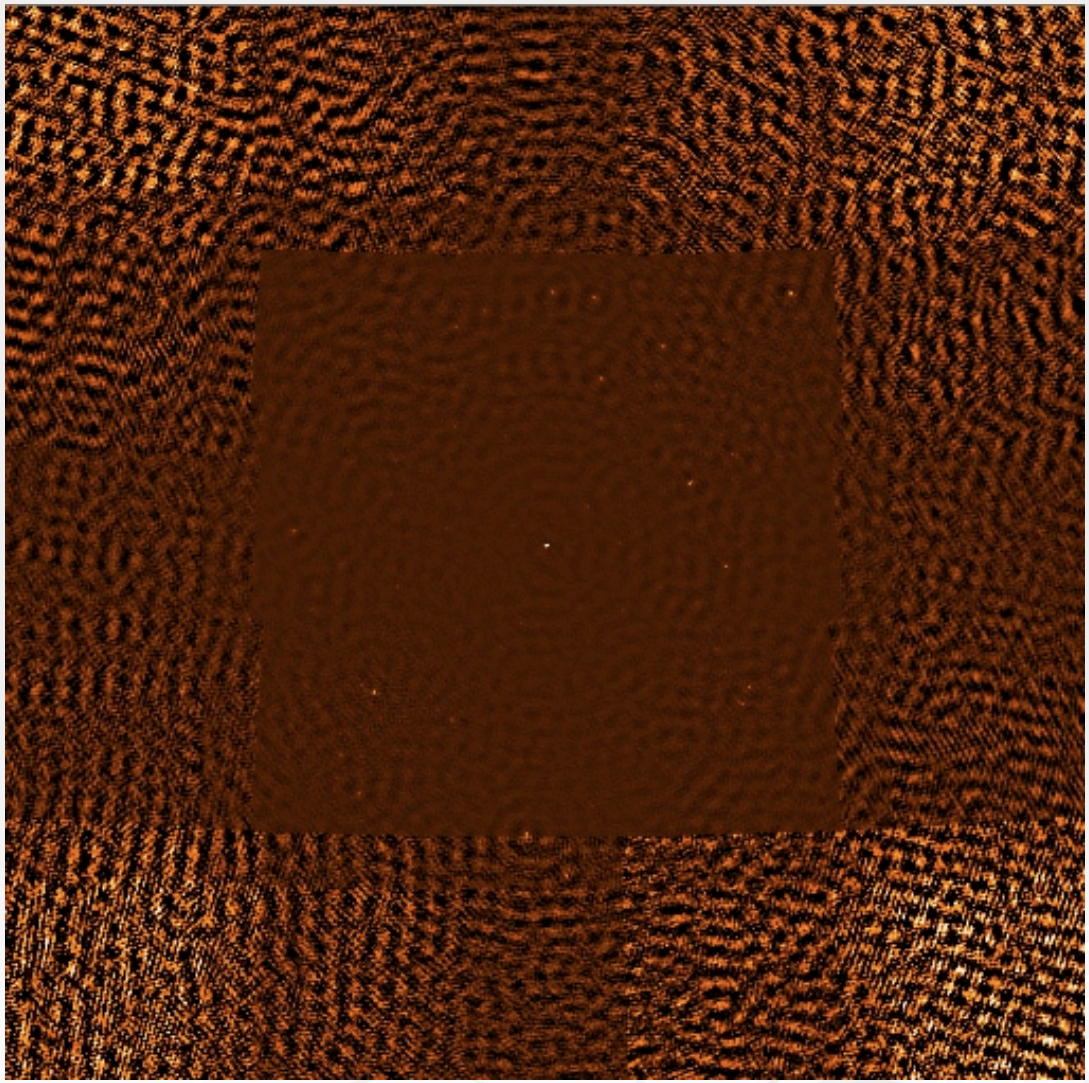
BBS: warnings

- incomplete skymodels:
  - you cannot solve for Cyg A only if the Cas A effects are of the same order as Cas A)
- be careful about the “patch” specification in the skymodel
  - patch has to be “tiny” as beam is computed only once for the center of the patch
- note: A-team solutions will be rubbish for remote stations if there is bandwidth smearing (for 1 channel avg. data)
  - affects quality of selfcal solutions for other stations

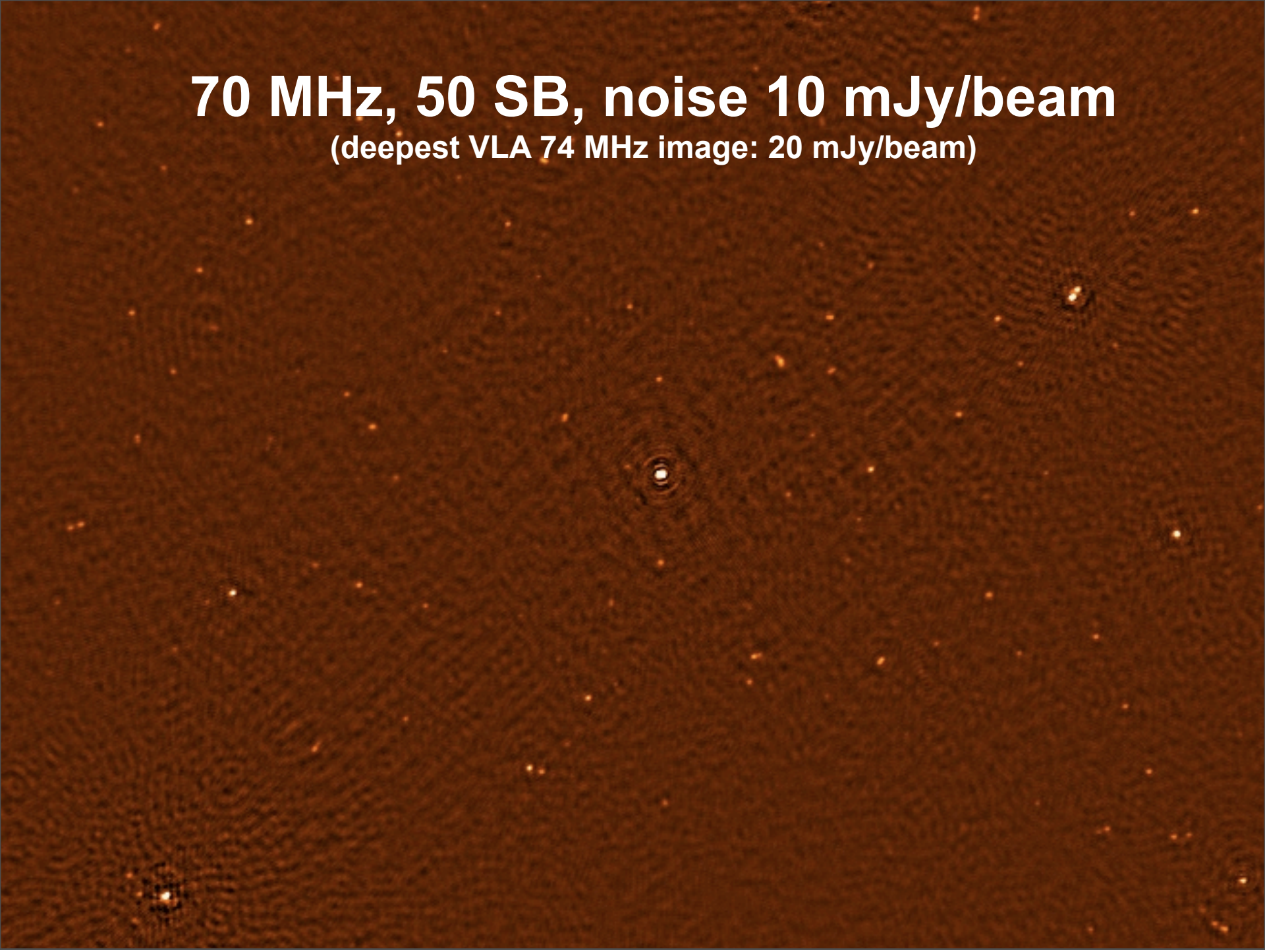
Imaging 3C196

- imagermode='csclean' (stabilizes clean)
  - limiting uvranges: can be very helpful
- When imaging do not use natural weighting with(!) the short baselines (use briggs in this case)
  - reason: subtraction Cas A is not perfect
  - Cyg A, Tau A, Vir A, ...
- if your model only contained 3C196: never image including the short baselines data (uvrange='>2km')

“vdtol” imager



55 MHz, 1 SB


70 MHz, 50 SB, noise 10 mJy/beam
(deepest VLA 74 MHz image: 20 mJy/beam)