Calibrating LOFAR Data with CASA

Laura Bîrzan (Leiden University)

CASA vs. BBS

- CASA and BBS give identical solutions
- CASA is more interactive:
  - Can inspect and flag solutions
  - easier for, e.g., self calibration
- CASA cannot do directional gains or LOFAR beam corrections as BBS can

Overview of Calibration with CASA

- The following steps should be followed to calibrate LOFAR data (post-NDPPP) in CASA:
    1. Run “clearcal” in CASA to add columns to measurement set
    2. Run BBS to simulate sky model – BBS puts Fourier transform of sky model in MODEL_DATA column
    3. Run “gaincal” in CASA to solve for gains
    4. Inspect and flag gain solutions using “plotcal” in CASA
    5. Run “applycal” in CASA to apply gains

Initial Steps

1. "clearcal" - adds needed columns to MS:

```
CASA <5>: inp clearcal
--------> inp(clearcal)
#  clearcal :: Re-initializes the calibration for a visibility data set
vis                 = 'SB169.MS.dppp'   #  Name of input visibility file (MS)
field               =         ''        #  Select field using field id(s) or field name(s)
spw                 =         ''        #  Select spectral window/channel.
async               =      False        #  If true the taskname must be started using clearcal(...)
```

2. Simulate in BBS – adds model to MS:

```
Strategy.Stations = []
Strategy.InputColumn = DATA
Strategy.TimeWindow = []
Strategy.ChunkSize = 0
Strategy.UseSolver = F
Strategy.Correlation.Selection = CROSS
Strategy.Correlation.Type = []
Strategy.Steps = [predict]

Step.predict.Baselines.Station1 = []
Step.predict.Baselines.Station2 = []
Step.predict.Model.Sources      = []
Step.predict.Model.Components = []
Step.predict.Correlation.Selection = CROSS
Step.predict.Correlation.Type = []
Step.predict.Operation      = PREDICT
Step.predict.Output.Column   = MODEL_DATA
```

Calibration

- "gaincal" in CASA – solves for gains:

```
#  gaincal :: Determine temporal gains from calibrator observations
vis                 = 'SB169.MS.dppp'   #  Name of input visibility file
caltable            = 'SB169.gcal'      #  Name of output gain calibration table
field               =         ''        #  Select field using field id(s) or field name(s)
spw                 =         ''        #  Select spectral window/channels
selectdata          =      False        #  Other data selection parameters
solint              =       '3s'        #  Solution interval: egs. 'inf', '60s' (see help)
combine             =         ''        #  Data axes which to combine for solve (scan, spw, and/or field)
preavg              =       -1.0        #  Pre-averaging interval (sec) (rarely needed)
refant              =         ''        #  Reference antenna name.  ' '= '0'
minblperant         =          4        #  Minimum baselines _per antenna_ required for solve
minsnr              =        0.0        #  Reject solutions below this SNR
solnorm             =      False        #  Normalize average solution amplitudes to 1.0 (G, T only)
gaintype            =        'G'        #  Type of gain solution (G, T, or GSPLINE)
calmode             =       'ap'        #  Type of solution" ('ap', 'p', 'a')
append              =      False        #  Append solutions to the (existing) table
gaintable           =       ['']        #  Gain calibration table(s) to apply on the fly
gainfield           =       ['']        #  Select a subset of calibrators from gaintable(s)
interp              =       ['']        #  Temporal interpolation for each gaintable (=linear)
spwmap              =         []        #  Spectral windows combinations to form for gaintables(s)
gaincurve           =      False        #  Apply internal VLA antenna gain curve correction
opacity             =        0.0        #  Opacity correction to apply on the fly (nepers)
parang              =      False        #  Apply parallactic angle correction on the fly
async               =      False        #  If true the taskname must be started using gaincal(...)
```

One can do phase-only ('p') or amplitude-only ('a') calibration as well

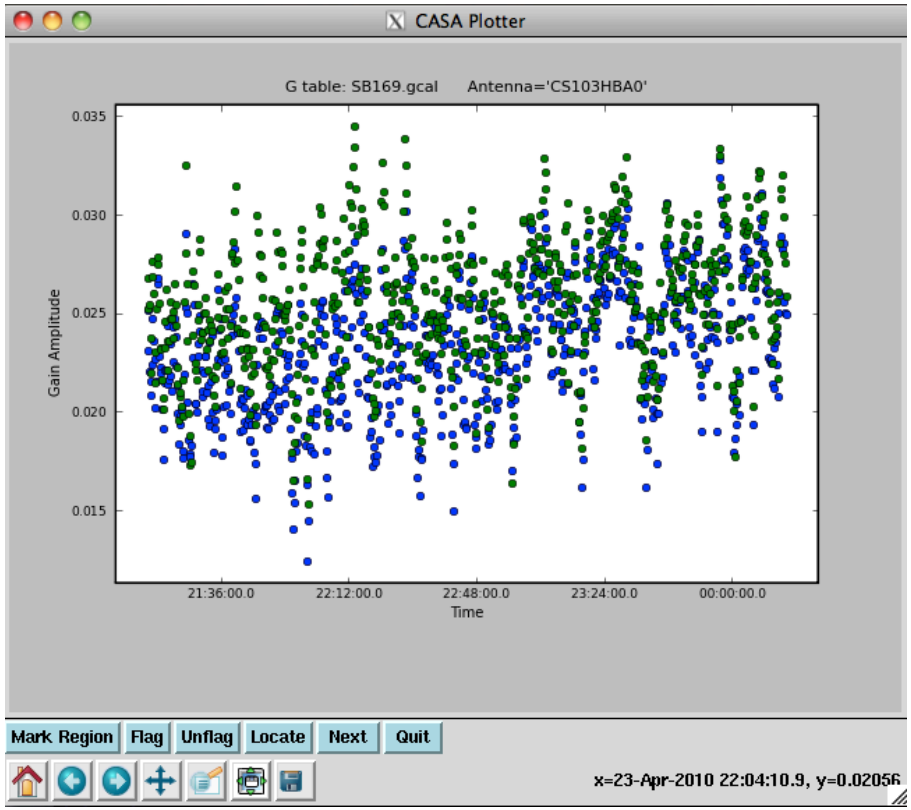
Inspect and Flag Solutions

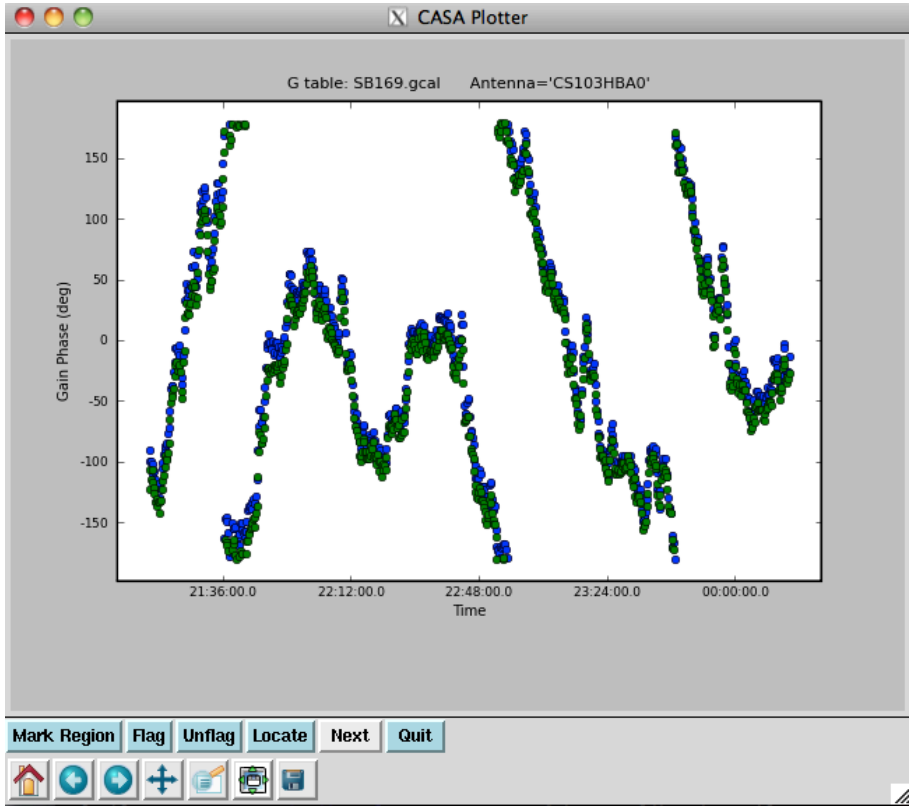
- "plotcal" in CASA:

```
#  plotcal :: An all-purpose plotter for calibration results
caltable            = 'SB169.gcal'      #  Name of input calibration table
xaxis               =         ''        #  Value to plot along x axis (time,chan,freq...see pdoc)
yaxis               =         ''        #  Value to plot along y axis (amp,phase,real,imag,snr,antenna)
poln                =         ''        #  Antenna polarization to plot (RL,R,L,XY,X,Y,/)
field               =         ''        #  field names or index of calibrators: ''==>all
antenna             =         ''        #  antenna/baselines: ''==>all, antenna = '3,VA04'
spw                 =         ''        #  spectral window:channels: ''==>all, spw='1:5~57'
timerange           =         ''        #  time range: ''==>all
subplot             =        111        #  Panel number on display screen (yxn)
overplot            =      False        #  Overplot solutions on existing display
clearpanel          =     'Auto'        #  Specify if old plots are cleared or not (ignore)
iteration           =  'antenna'        #  Iterate plots on antenna,time,spw,field
plotrange           =         []        #  plot axes ranges: [xmin,xmax,ymin,ymax]
showflags           =      False        #  If true, show flagged solutions
plotsymbol          =        'o'        #  pylab plot symbol
plotcolor           =     'blue'        #  initial plotting color
markersize          =        5.0        #  Size of plotted marks
fontsize            =       10.0        #  Font size for labels
showgui             =       True        #  Show plot on gui
figfile             =         ''        #  ''= no plot hardcopy, otherwise supply name
async               =      False        #  If true the taskname must be started using plotcal(...)
```

Setting "iteration" to 'antenna' will plot solutions station-by-station

Inspect and Flag Solutions





Flag bad solutions by selecting “Mark region”, drawing region on plot, then selecting “Flag”

Apply Calibration

- "applycal" in CASA - writes the calibrated data to the CORRECTED_DATA column of the MS:

```
#  applycal :: Apply calibrations solutions(s) to data
vis                 = 'SB169.MS.dppp'   #  Name of input visibility file
field               =         ''        #  Select field using field id(s) or field name(s)
spw                 =         ''        #  Select spectral window/channels
selectdata          =      False        #  Other data selection parameters
gaintable           = 'SB169.gcal'      #  Gain calibration table(s) to apply on the fly
gainfield           =       ['']        #  Select a subset of calibrators from gaintable(s)
interp              =       ['']        #  Temporal Interpolation type.  default=linear
spwmap              =         []        #  Spectral windows combinations to form for gaintables(s)
gaincurve           =      False        #  Apply internal VLA antenna gain curve correction
opacity             =        0.0        #  Opacity correction to apply (nepers)
parang              =      False        #  Apply parallactic angle correction
calwt               =       True        #  Calibrate data weights from all relevant calibrations
async               =      False        #  If true the taskname must be started using applycal(...)
```

Other Useful CASA Tasks

- “split” – used to split off CORRECTED_DATA and to average the data before imaging
- “setjy” – used to place a model (using an existing CASA model image) into the MS

Review of Calibration with CASA

1. Run “clearcal”
2. Run BBS to simulate sky model (or use “setjy” if you have a CASA model image already)
3. Run “gaincal” to solve for gains
4. Inspect and flag gain solutions using “plotcal
5. Run “applycal” to apply gains

For selfcal:

6. Image with “clean” – this step make the image and puts an updated model in the MODEL_DATA column
7. Go to 3