Tutorial 2: Calibration

E. Mahony
Name speaker

Start from preprocessed data (flagged and averaged)

- Log onto CEP3 as described in Tutorial 1:

```
> ssh -Y portal.lofar.eu
> ssh -Y lhd002
> use Slurm
> srun -A <accountname> --reservation=<reservationname> -t 600
-u bash -i
*In a new terminal*
> ssh -Y portal.lofar.org
> ssh -Y lhd002
> ssh -Y lof0xx
cd to your working directory
> use Lofar
```

Start from preprocessed data (flagged and averaged)

- We need 1 subband of the calibrator and 1 subband of the target
  - They need to be at the same frequency and averaged the same way!!
- If you didn't get up to this step in yesterday's tutorial, copy from here (on your working node):

```
/data/dataschool2014/imaging/t1/L114220_SAP000_SB031_uv_averaged.MS/
/data/dataschool2014/imaging/t1/L114221_SAP000_SB031_uv_averaged.MS/
```

- For this tutorial I've renamed the filenames for simplicity (also good to create a backup):

```
> cp -rf L114220_SAP000_SB031_uv_averaged.MS/ cal_averaged.MS
> cp -rf L114221_SAP000_SB031_uv_averaged.MS/ target_averaged.MS
```

Check calibrator and field data

- We need to calculate and transfer the solutions from the calibrator to the target field
  - This only works if the frequencies are **identical**

> msoverview in=cal_averaged.MS

> msoverview in=target_averaged.MS

```
mahony@lof013:/data/scratch/mahony/tutorial_t2$ msoverview in=cal_averaged.MS/
msoverview: Version 20110407GvD
================================================================================
           MeasurementSet Name:  /data/scratch/mahony/tutorial_t2/cal_averaged.MS      MS Version 2
================================================================================
   Observer: unknown     Project: MSSS_HBA_2013
Observation: LOFAR
Antenna-set: HBA_DUAL_INNER

Data records: 10620       Total integration time = 60.0834 seconds
   Observed from   29-Mar-2013/13:59:48.0   to   29-Mar-2013/14:00:48.1 (UTC)


Fields: 1
  ID   Code Name                RA                Decl             Epoch        nRows
  0         BEAM_0              01:37:41.299440 +33.09.35.13240 J2000        10620

Spectral Windows:  (1 unique spectral windows and 1 unique polarization setups)
  SpwID  Name   #Chans   FrameCh1(MHz)     ChanWid(kHz)  TotBW(kHz) CtrFreq(MHz)  Corrs
  0      SB-31      4   TOPO      134.300        48.828       195.3   134.3735   XX  XY  YX  YY
```

- Check if central frequencies are the same
- To do the calibration we need a parset file and a skymodel

Finding a skymodel for the calibrator

ASTRON

- Running msoverview also gives us the position of the pointing centre (i.e. the calibrator source)

```
mahony@lof013:/data/scratch/mahony/tutorial_t2$ msoverview in=cal_averaged.MS/
msoverview: Version 20110407GvD
================================================================================
           MeasurementSet Name:  /data/scratch/mahony/tutorial_t2/cal_averaged.MS      MS Version 2
================================================================================
   Observer: unknown     Project: MSSS_HBA_2013
Observation: LOFAR
Antenna-set: HBA_DUAL_INNER

Data records: 10620       Total integration time = 60.0834 seconds
   Observed from   29-Mar-2013/13:59:48.0   to   29-Mar-2013/14:00:48.1 (UTC)


Fields: 1
  ID   Code Name                RA               Decl           Epoch        nRows
  0         BEAM_0              01:37:41.299440 +33.09.35.13240 J2000        10620

Spectral Windows:  (1 unique spectral windows and 1 unique polarization setups)
  SpwID  Name   #Chans   FrameCh1(MHz)     ChanWid(kHz)  TotBW(kHz) CtrFreq(MHz)  Corrs
  0      SB-31       4   TOPO     134.300         48.828       195.3    134.3735   XX  XY  YX  YY
```

- A quick look in NED tells us that this is 3C48, a well known calibrator.

Finding a skymodel for the calibrator

ASTRON

- Check to see if 3C48 skymodel exists in the Models database on CEP3:

```
> cd /globaldata/COOKBOOK/Models
```

- Have a look at which models are available and copy the skymodel you prefer

```
(from your working directory)
> cp /globaldata/COOKBOOK/Models/3C48.skymodel .
> more 3C48.skymodel
```

```
mahony@lof013:/data/scratch/mahony/tutorial_t2$ more 3C48.skymodel
# (Name, Type, Ra, Dec, I, ReferenceFrequency='150.e6', SpectralIndex) = format

3c48, POINT, 01:37:41.299431, 33.09.35.132990, 70.399325, , [-0.396150,-0.650172,0.335733,-0.059050]
```

- Note: if your calibrator source isn’t there you can make your own skymodel – we’ll come to this later

Writing the parset file

ASTRON

solvecal.parset

Strategy.ChunkSize = 0	read entire MS into memory (By default BBS reads in the DATA column)
Strategy.Steps = [solve,correct]	
Step.solve.Operation = SOLVE	
Step.solve.Model.Sources = [3c48]	The source you want to solve for
Step.solve.Model.Gain.Enable = T	
Step.solve.Model.Beam.Enable = T	Correct for the beam
Step.solve.Solve.Parms = ["Gain:0:0:*","Gain:1:1:*"]	Solve for xx,yy (not xy,yx), * means amplitude and phase
Step.solve.Solve.CellSize.Freq = 0	Number of channels to solve for (0=all)
Step.solve.Solve.CellSize.Time = 1	Number of timeslots to solve for (1=solve for every timestamp)
Step.solve.Solve.CellChunkSize = 10	Number of solution cells to simultaneously process
Step.solve.Solve.Options.MaxIter = 50 Step.solve.Solve.Options.EpsValue = 1e-9 Step.solve.Solve.Options.EpsDerivative = 1e-9 Step.solve.Solve.Options.ColFactor = 1e-9 Step.solve.Solve.Options.LMFactor = 1.0 Step.solve.Solve.Options.BalancedEqs = F Step.solve.Solve.Options.UseSVD = T	These are the default values that determine the stop criteria – see wiki for details
Step.correct.Operation = CORRECT	
Step.correct.Model.Gain.Enable = T	
Step.correct.Model.Beam.Enable = T	Apply beam
Step.correct.Model.Sources = [3c48]	
Step.correct.Output.Column = CORRECTED_DATA	

- Can copy parset from /data/dataschool2014/lof013_t2/parsets/solvecal.parset

Run BBS

- Calibrate the data by running BBS. First look at the help file:

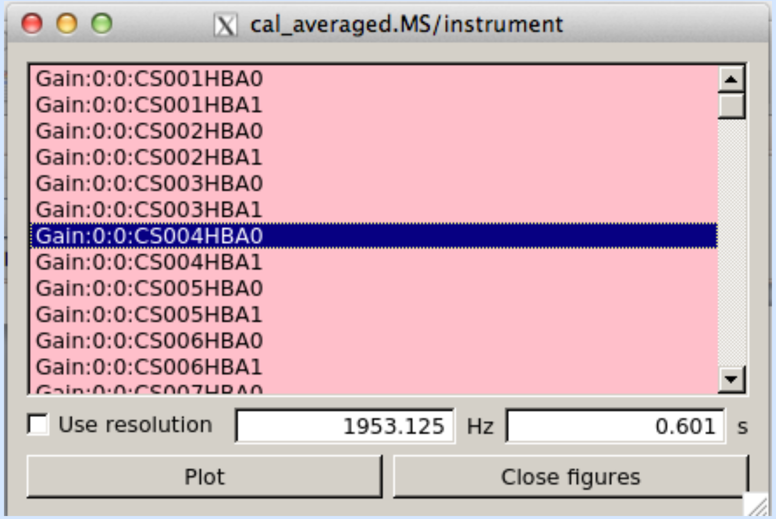
```
> calibrate-stand-alone -h
```

- The run the calibration command with the relevant parset and skymodel:

```
> calibrate-stand-alone -f cal_averaged.MS/ solvecal.parset 3C48.skymodel > solvecal.log &
```

- Check the log output to see if BBS finished successfully (or crashed)
- The solutions are stored in the instrument table – check the solutions using parmdbplot:

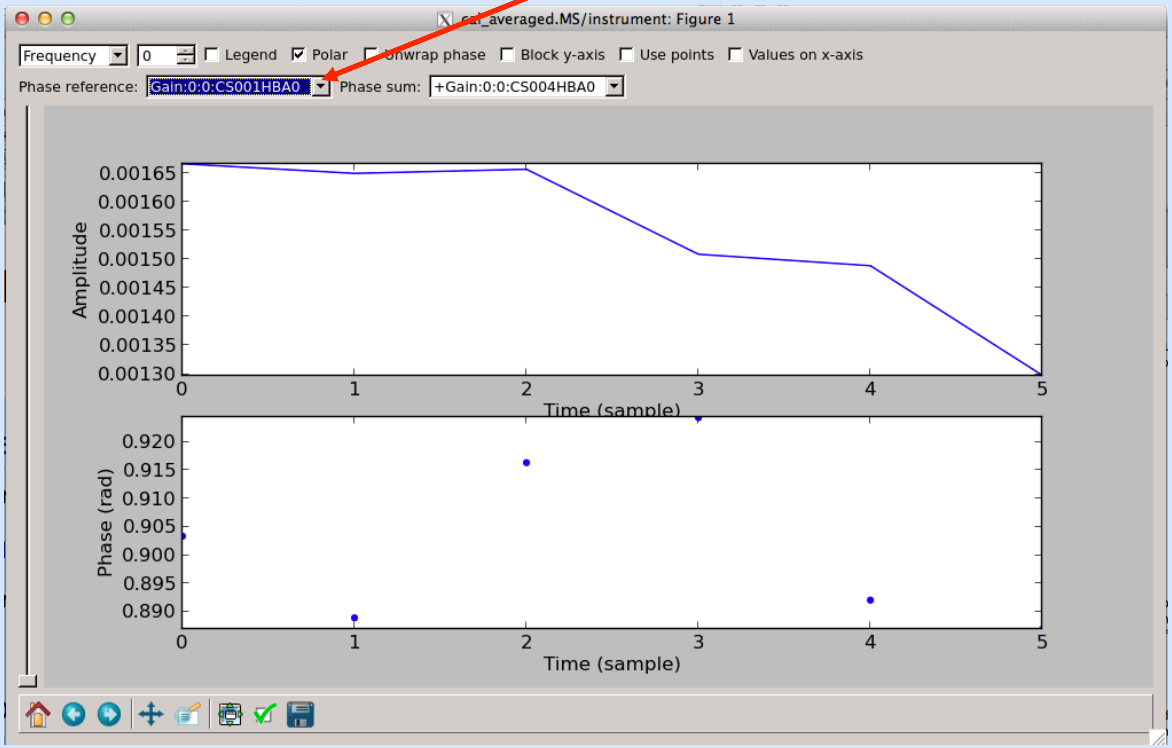
```
> parmdbplot.py cal_averaged.MS/instrument/ &
```



Inspect solutions

- Gain:0:0:CS004HBA0

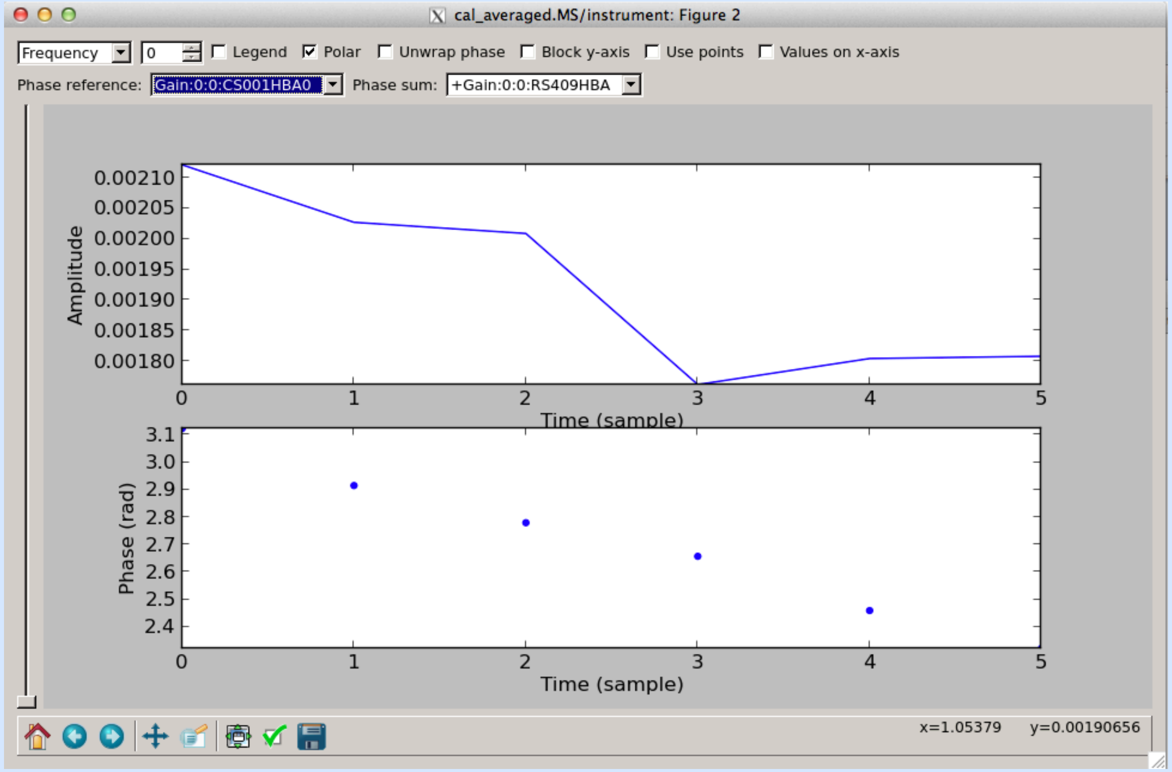
Remember to set a phase reference!



- We only 60s of data, so there are not many solutions. Check that there are no bad solutions

Inspect solutions

- Gain:0:0:RS409HBA



- Note: the phases will usually change more rapidly on the longer baselines

Transfer solns to target field

- We need to make the solutions time-independent to transfer the gains to the target field.

```
> parmexportcal in=cal_averaged.MS/instrument/ out=3c48solns
```

- Apply gain solutions to target field by doing a correct step in BBS:

```
> calibrate-stand-alone --parmdb 3c48_solns target_averaged.MS/ transfersolns.parset > applycal.log &
```

transfersolns.parset:

```
Strategy.ChunkSize = 0
Strategy.Steps = [correct]

Step.correct.Operation = CORRECT
Step.correct.Model.Sources = []
Step.correct.Model.Gain.Enable = T
Step.correct.Model.Beam.Enable = F
Step.correct.Output.Column = CORRECTED_AMP
```

NOTE: do NOT apply the beam in this correct step. We only want to apply the beam at the last correct step before imaging!

Phase calibration on the field

- Get a skymodel for the target field
  - Run msoverview to get the co-ordinates of the pointing centre (RA=01:02:21.73, Dec=+31:27:36.0)
- Get the GSM skymodel for this field using gsm.py

```
> gsm.py –h
> gsm gsm.py targetfield.skymodel 15.59 31.46 3 1
Sky model stored in source table: targetfield.skymodel

> more targetfield.skymodel
```

```
mahony@lof013:/data/scratch/mahony/tutorial_t2$ more targetfield.skymodel
FORMAT = Name, Type, Ra, Dec, I, Q, U, V, ReferenceFrequency='60e6', SpectralIndex='[0.0]', MajorAxis, MinorAxis, Orientation

# the next lines define the sources
0049.0+3220, POINT, 00:49:01.94880000, +32.20.23.20800000, 2.8587, , , , , [-0.5724, -0.1103]
0050.2+3229, POINT, 00:50:17.52960000, +32.29.14.38800000, 2.7945, , , , , [-0.657, -0.1036]
0050.9+3050, POINT, 00:50:56.46000000, +30.50.03.58800000, 1.2646, , , , , [-0.7373, -0.1638]
0053.7+2925, GAUSSIAN, 00:53:44.60880000, +29.25.10.88400000, 5.0451, , , , , [-0.7525, 0.0181], 49.7, 35.0, 165.7
0053.8+3114, GAUSSIAN, 00:53:49.51920000, +31.14.48.91200000, 8.7384, , , , , [-0.8641], 43.4, 40.3, 32.8
0054.1+3203, POINT, 00:54:09.52080000, +32.03.43.99200000, 1.3, , , , , [-0.7]
0054.1+3101, POINT, 00:54:09.95040000, +31.01.59.41200000, 1.2422, , , , , [-0.5142, -0.2463]
0054.2+3201, POINT, 00:54:17.53920000, +32.01.06.88800000, 3.4466, , , , , [-0.6143]
0054.3+3353, POINT, 00:54:22.03920000, +33.53.36.09600000, 2.0607, , , , , [-0.4791, -0.1043]
0054.6+3219, POINT, 00:54:41.88000000, +32.19.04.58400000, 1.1784, , , , , [-0.6299, -0.1301]
0057.7+3021, GAUSSIAN, 00:57:46.60080000, +30.21.34.59600000, 4.25, , , , , [-0.7], 105.0, 52.7, 136.7
0058.0+3121, POINT, 00:58:05.69040000, +31.21.13.60800000, 2.8847, , , , , [-0.0435, -0.3402]
```

Phase calibration on the field

solve_phaseonly.parset

```
Strategy.InputColumn = CORRECTED_AMP # define input column
Strategy.ChunkSize = 500
Strategy.Steps = [solve, correct]

Step.solve.Operation = SOLVE
Step.solve.Model.Sources = []  #solves for all sources in skymodel
Step.solve.Model.Cache.Enable = T
Step.solve.Model.Phasors.Enable = T
Step.solve.Model.Gain.Enable = T
Step.solve.Model.Beam.Enable = T
Step.solve.Model.Beam.UseChannelFreq = F
Step.solve.Solve.Mode = COMPLEX #use COMPLEX not PHASE
Step.solve.Solve.Parms = ["Gain:0:0:Phase:*", "Gain:1:1:Phase:*"]
Step.solve.Solve.CellSize.Freq = 0
Step.solve.Solve.CellSize.Time = 1
Step.solve.Solve.CellChunkSize = 40
Step.solve.Solve.PropagateSolutions = F   #don't use previous
Step.solve.Solve.Options.MaxIter = 50      solution as starting guess
Step.solve.Solve.Options.EpsValue = 1e-9
Step.solve.Solve.Options.EpsDerivative = 1e-9
Step.solve.Solve.Options.ColFactor = 1e-9
Step.solve.Solve.Options.LMFactor = 1.0
Step.solve.Solve.Options.BalancedEqs = F
Step.solve.Solve.Options.UseSVD = T
```

```
Step.correct.Operation = CORRECT
Step.correct.Model.Sources = []
Step.correct.Model.Phasors.Enable  = T
Step.correct.Model.Gain.Enable  = T
Step.correct.Model.Beam.Enable  = T
Step.correct.Model.Beam.UseChannelFreq = F
Step.correct.Output.Column = CORRECTED_DATA
```

UseChannelFreq – this option needs to be set to True when using datasets where multiple subbands have been combined. (this corrects for how the beam changes with frequency).

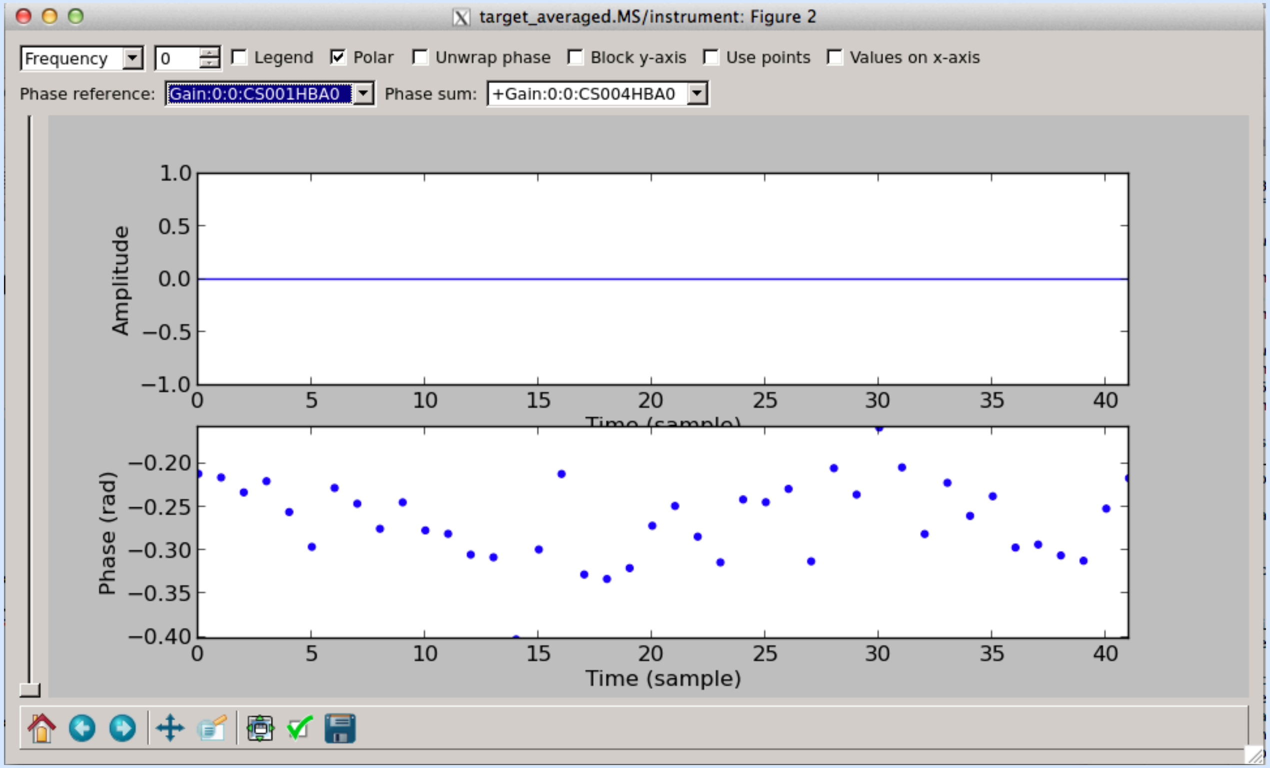
Phase calibration on the field

- Run BBS:

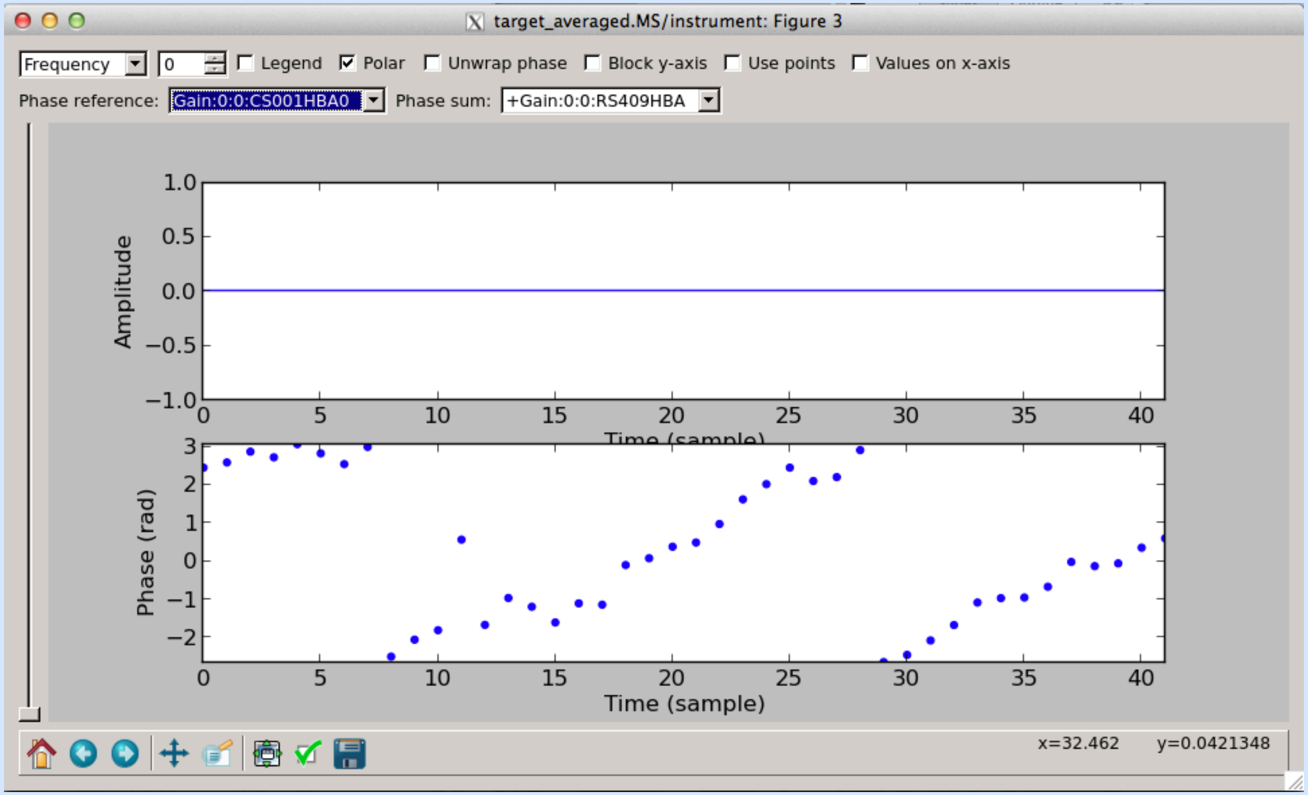
```
> calibrate-stand-alone -f target_averaged.MS/ solve_phaseonly.parset
targetfield.skymodel > phasecal.log &
```

- Inspect solutions:

```
> parmdbplot.py target_averaged.MS/instrument/ &
```



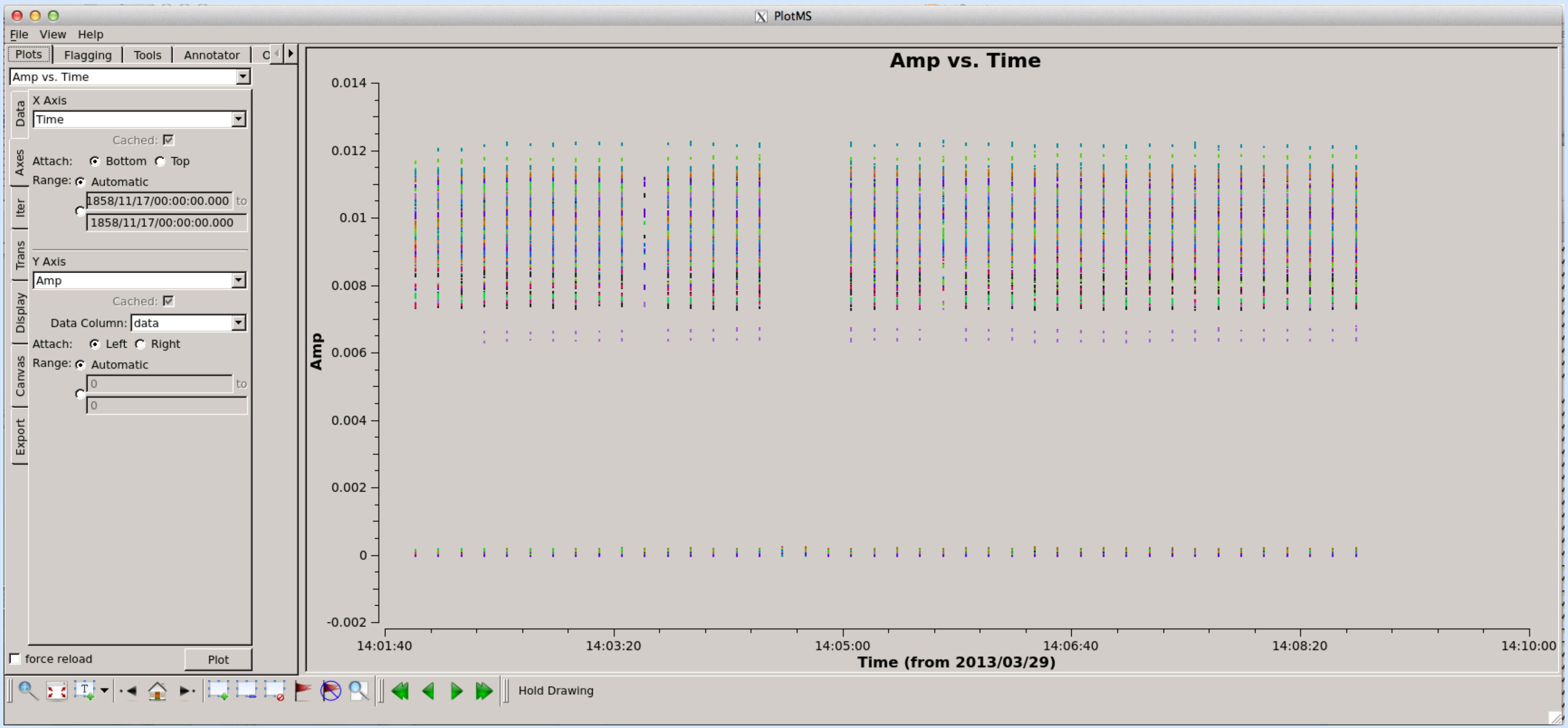
Inspect solutions



- And you're done! almost…
    - Inspect the data in plotms again to check the data has been properly calibrated

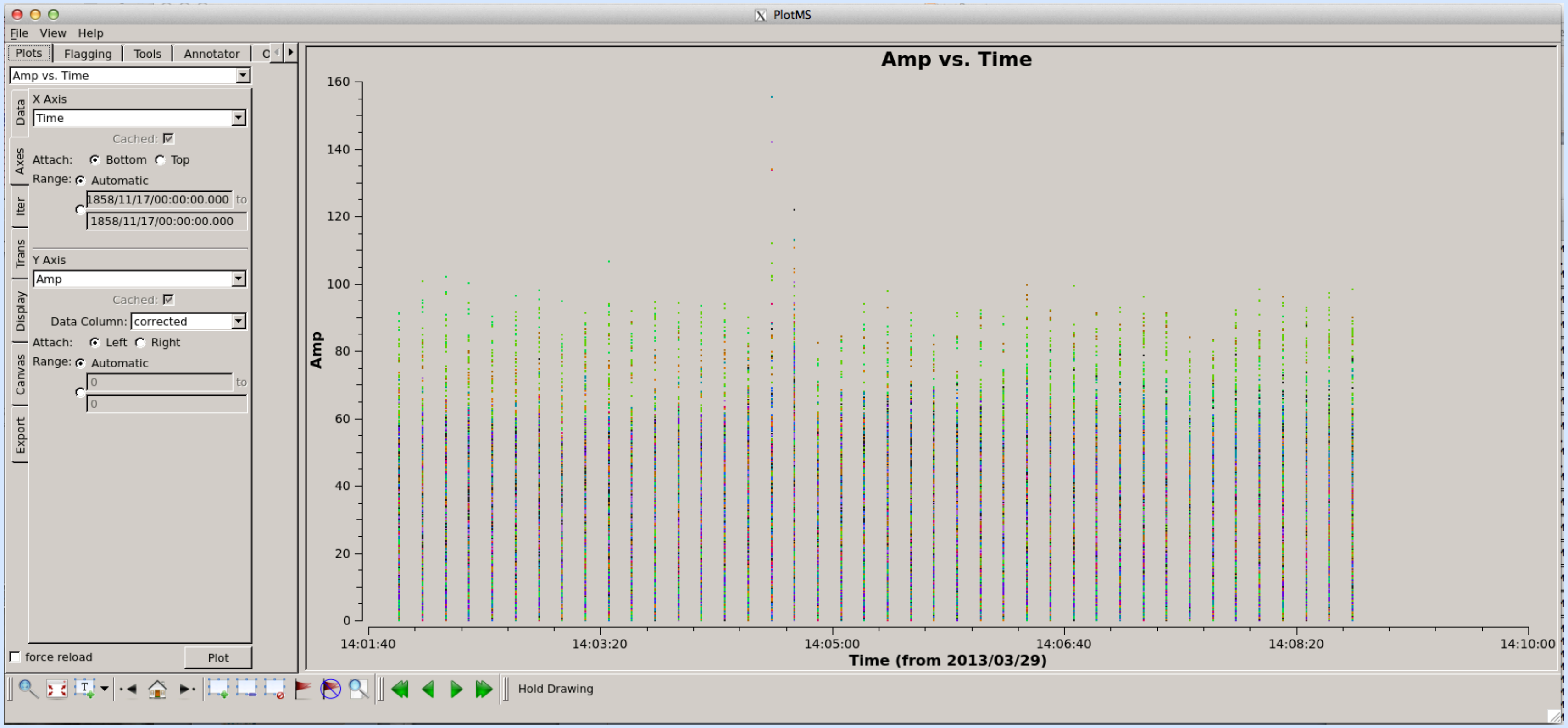
Check the calibrated data

Time vs. amp – pre-calibration



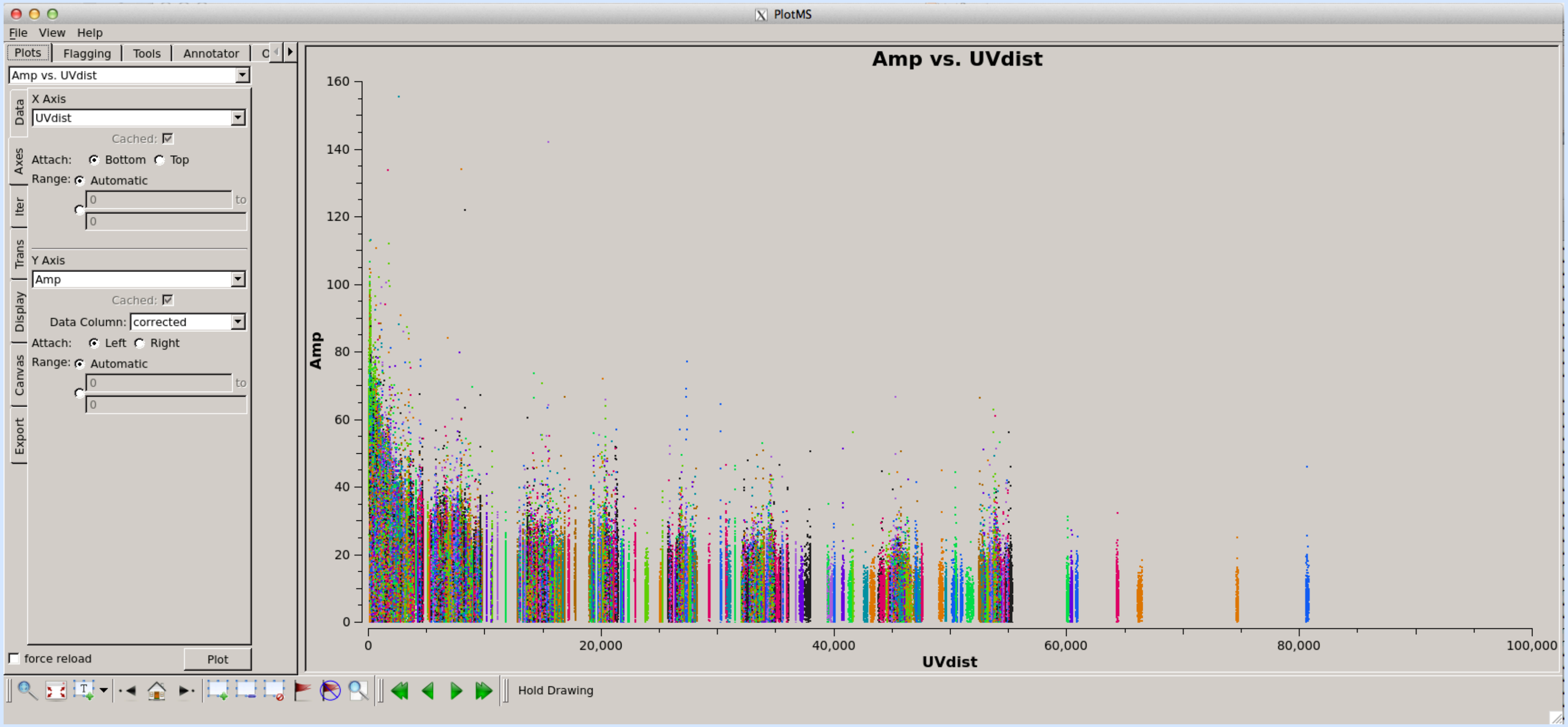
Check the calibrated data

Time vs. amp – post-calibration (check the amplitude scale has changed)



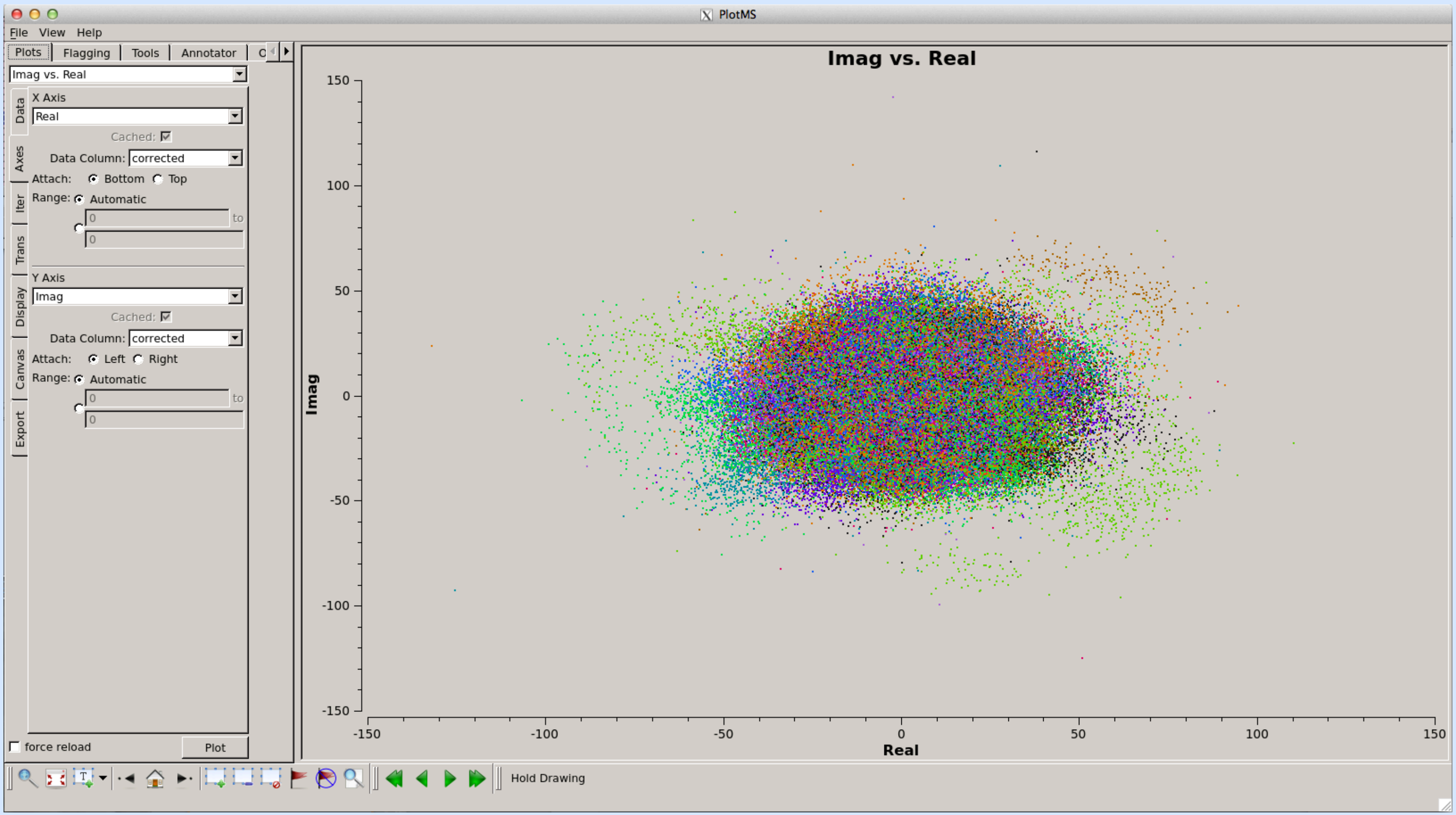
Check the calibrated data

- Amp vs. uvdist



Check the calibrated data

- Real vs. imag



Things to try next…

- Expand your script from tutorial 1 to do the calibration for a number of subbands
- Calibrate more subbands to use for the imaging tutorial (tomorrow)
- Try combined subbands prior to the phase calibration to see if that improves the quality of the solutions (more S/N is often needed for the phase calibration, particularly in fields with no dominate bright sources)