

# Radio Astronomy Tutorial 2: Self-Calibration February 23, 2017

Joshua Albert

albert@strw.leidenuniv.nl

Jit Mandal

mandal@strw.leidenuniv.nl

## 1. Introduction

In the previous session you learned the basic concepts of synthesis imaging. This session will build on that with self-calibration. The dataset for this session was taken with the VLBA, which allows us to image the extragalactic source BL Lacertae on size scales of milli- arcseconds. The data has been calibrated and fringe-fitted to correct for the additional delays introduced by the fact that your antennas are widely distributed, and on independent clocks.

Create a sub-directory in your own directory on the local disk. The data can be found at `/net/pczaal2/data1/albert/RA2017/PRACTICAL2/ra2017tut2.tar`. Copy the tar file to your local disk and untar with the command `tar xvf ra2017tut2.tar`.

## 2. Data Inspection

First, familiarize yourself with the data. Don't take too long here, more importantly you want to get to imaging. There are three files:

- `bllac.2cm.MS`: 15 GHz dataset
- `bllac.4cm.cal.MS`: 7.5 GHz dataset, already self-calibrated
- `bllac.4cm.mod.MS`: 7.5 GHz dataset, but with an error introduced.

Start with the 7.5 GHz, already self-calibrated, dataset. You can use `listobs` to gather general information on the observation. Use `plotms` to look at amplitude vs. time, and amplitude vs. uv-distance (UVdist).

1. What type of object do you think this is? Why?
2. Can you tell if it is resolved by looking at amp vs. uvwave?
3. Also look at the uv coverage by plotting u vs. v. What is the maximum angular resolution?

4. You can also use `plotants` to plot the antenna locations, and see if you can match them to the antenna names: <http://www.vlba.nrao.edu/>.

Repeat the same for the 15 GHz dataset, and then the 7.5 GHz dataset with the error introduced.

## 3. Imaging

Now you can play around with imaging options. Use the `bllac.4cm.cal.MS` dataset, and try out uniform, natural, and robust (Briggs, 1995) weighting schemes<sup>1</sup>. Use `viewer` to inspect not only the images, but the psf and the residuals. Do you see differences in them for different weightings?

The `uvtaper` parameter in `clean` can be used to apply a gaussian taper to the visibility data, effectively down-weighting the longest baselines. Do you understand what tapering does? This can help bring out extended features. Try setting `outertaper=['0.0001arcsec']`.

- What happens to the resulting image?
- What if you use `outertaper=['0.0002arcsec']`?

## 4. Error Recognition and self-calibration

Now image the `bllac.4cm.mod.MS` dataset and look at the resultant image. What sort of error do you think is present? And why? We can try to self-calibrate the target to remove the error.

Self-calibration uses the model that you created during imaging, otherwise known as the clean component model. This is stored in the model column of your measurement set when `clean` is run, and is over-written every time you run `clean`. The task `gaincal` is used to calibrate the data against the model column. Then you can use `plotcal` to look at the solutions, and `applycal` to apply them. This

---

<sup>1</sup>Remember when you use robust weighting in a report that you properly cite Briggs 1995. Here we have omitted the proper citation so you can go find it. Robust weighting comes from the late Dan Briggs.

process is iterated until the self-calibration converges, i.e., the solutions don't change. The solutions are always applied to the original data.

Recall self-calibration consists of these general steps.

1. Use `clean` to create a source model with current data.
2. Perform `gaincal` on current data against the model from step 1.
3. Apply the solutions from step 2.
4. If stopping condition not met then go to step 1.

Each time you loop from steps 1 to 3 you have done one iteration of self-calibration. In practice usually performs at least five rounds of self-cal. Each iteration slowly (hopefully) converges to a global solution. Each step requires understanding. For instance, in step 1 when creating a model with `clean`, it doesn't make sense to clean *too* deeply, because we are not actually sure about the source distribution (since the data is not fully calibrated yet). `gaincal` will try to calibrate the data to the model you give so you don't want to start off too confident about the model by cleaning too deep. Also in step 2 there are different types of calibration one can perform. What are these? In practice we usually start with the type of calibration that *moves flux around, and doesn't change the amount of flux*. After several rounds of this type of self-cal, one can then select the type of calibration that *moves flux around, and also changes the amount of flux*.

It is instructive to first run one self-calibration loop on `b1lac.4cm.cal.ms`, to see what the phase solutions should look like. We've already imaged (step1), so we can start out with step2, `gaincal`. Name the caltable something like `mycaltable.gcal0`, use `refant='ANT5'`, and set the solution interval to 300 sec (5 minutes). The `gaintype` should be 'G' and the `calmode` should be set to 'p' for a phase-only calibration. Once you have run the calibration, use `plotcal` to inspect the solutions. You can save a plot file to compare with later on.

We will now self-calibrate `b1lac.4cm.mod.ms` fully. Assuming we don't already have a good model to self-calibrate with, we can start with a point source model (we are not a priori confident, right?). Do this by running `clean` with only one iteration, which will give you a clean component model with only one point. You can use `viewer` to inspect the image and model. Now use `gaincal` with the same parameters as before, and then `plotcal` to look at the solutions. Can you see the error that was introduced in the data?

Apply the solutions with `applycal` (with `gaintable=<yourcalibrationtable>`,

`gainfield='0'`). Then go to step 1. Iterate until you are happy with the solutions and image. After several rounds of `calmode = 'p'` you can switch to 'ap' mode which does amplitude and phase together. How does including amplitude corrections change the image?

It's useful to keep track of the iteration number you are on for the calibration tables and images by appending an iteration number to the end (e.g., `mycaltable.gcal0`, `mycaltable.gcal1`, `mycaltable.gcal2`, etc.).

## 5. Extra for the black-belts

Self calibrate the 15 GHz dataset, and then make a spectral index map between the two frequencies using `immath`, with `mode='spix'`. Spectral index means finding the  $\alpha$  so flux follows this relation  $S(\nu) = \frac{\nu}{\nu_0}^{-\alpha}$ .