EVN Primary Beam Model

Jack Radcliffe

Last updated: September 11, 2017

1 Introduction

This is the documentation for the primary beam model for the EVN developed by Jack Radcliffe as part of the Hubble Deep Field-North (HDF-N) wide-field VLBI project. Due to the lack of accurate primary beam models for many EVN telescopes, the primary beam can only be approximated as a Gaussian which is then applied directly to the upmost CL table in AIPS. To run the primary beam correction, you must have the packages described in Section 2.

2 Requirements

1. Python version 2.7 (3+ is not supported) with packages:
   - numpy
   - astropy (version 1.3.3 is recommended, 2+ does not work at the moment, it is a known bug)
   - scipy
   - matplotlib
2. AIPS (31DEC16 or newer)
3. ParselTongue
4. AIPS

3 Theory

Each phase centre is primary beam corrected by following similar steps to Cao+2014. Due to the lack of accurate primary beam models for many EVN telescopes, the primary beam power response of each telescope were approximated by using a normalized, symmetric, 2D Gaussian of the form,

$$P(\theta,\phi) \approx \exp\left(-\frac{(\theta-\theta_0)^2+(\phi-\phi_0)^2}{2\sigma^2}\right), \tag{1}$$

where $P(\theta,\phi)$ is the relative power response. θ and ϕ are the respective azimuthal and polar angular distances from the antennas' pointing centers. The azimuthal and polar coordinates of the telescope's pointing centers are defined by θ_0 and ϕ_0 respectively. The standard deviation, σ, can be related to the FWHM of the primary beam, $\theta_{1/2}$, through the expression,

$$\sigma^2 = \frac{\theta_{1/2}^2}{8\ln 2}, \tag{2}$$

where the FWHM of the primary beam is defined as,

$$\theta_{1/2} = K\frac{\lambda_{\mathrm{c}}}{D_{\mathrm{eff}}}. \tag{3}$$

Primary beam correction uses 2D Gaussian model to estimate the beam shapes. `pbcor_model.py` will take the inputs specified in this file and and uses `astropy.modeling` 2D Gaussians to model the primary beam shape. The astropy model used is of the form:

$$f(x,y) = A\exp(-a(x-x_0)^2 - b(x-x_0)(y-y_0) - c(y-y_0)^2) \tag{4}$$

where:

$$a = \left(\frac{\cos^2(\theta)}{2\sigma_x^2} + \frac{\sin^2(\theta)}{2\sigma_y^2}\right) \tag{5}$$

$$b = \left(\frac{\sin(2\theta)}{2\sigma_x^2} + \frac{\sin(2\theta)}{2\sigma_y^2}\right) \tag{6}$$

$$c = \left(\frac{\sin^2(\theta)}{2\sigma_x^2} + \frac{\cos^2(\theta)}{2\sigma_y^2}\right) \tag{7}$$

where θ is the rotation angle in radians. We assume that the primary beam model is circularly symmetric (a zeroth order approximation in the absence of primary beam models), therefore $\theta \to 0$, $\sin^2(\theta) \to 0$, $\cos^2(\theta) \to 1$ and $b \to 0$. Therefore Eqn 4 reduces to:

$$f(x,y) = A\exp\left(-\frac{(x-x_0)^2}{2\sigma_x^2} - \frac{(y-y_0)^2}{2\sigma_y^2}\right) \tag{8}$$

If we assume that the x direction corresponds to that of Right Ascension and y direction to Declination, we need to take into account that we require the angular offset from the pointing centre therefore:

$$\sigma_{\mathrm{RA}} = \sigma_x(1/\cos(\mathrm{Dec})) \tag{9}$$

$$\sigma_{\mathrm{Dec}} = \sigma_y \tag{10}$$

If we then combine Eqns. 3, 2, 9 and 10, we end up with expressions for the standard deviation as:

$$\sigma_{\mathrm{Dec}} = (1/\cos(\mathrm{Dec}))\frac{\theta_{1/2}}{(2\sqrt{2\log(2)})} \tag{11}$$

$$\sigma_{\mathrm{RA}} = \frac{\theta_{1/2}}{(2\sqrt{2\log(2)})} \tag{12}$$

The effective aperture diameters in use are shown in Table ??. A small correction factor, $K = 1.05$, is used to take into account any aperture blockages (Wrigley et al., in prep.). $P(\theta,\phi)$ was derived for each telescope at the centre of every phase centre. These correction factors are then applied using AIPS task `CLCOR`, where the gain amplitudes for each antenna were multiplied by the correction factors, $P(\theta,\phi)^{-1/2}$, of the two telescopes that form the baseline.

4 Algorithm

1. `pbcor_model.py` will take the inputs specified in this file and and uses `astropy.modeling` 2D Gaussians to model the primary beam shape. The standard deviation is entered as:

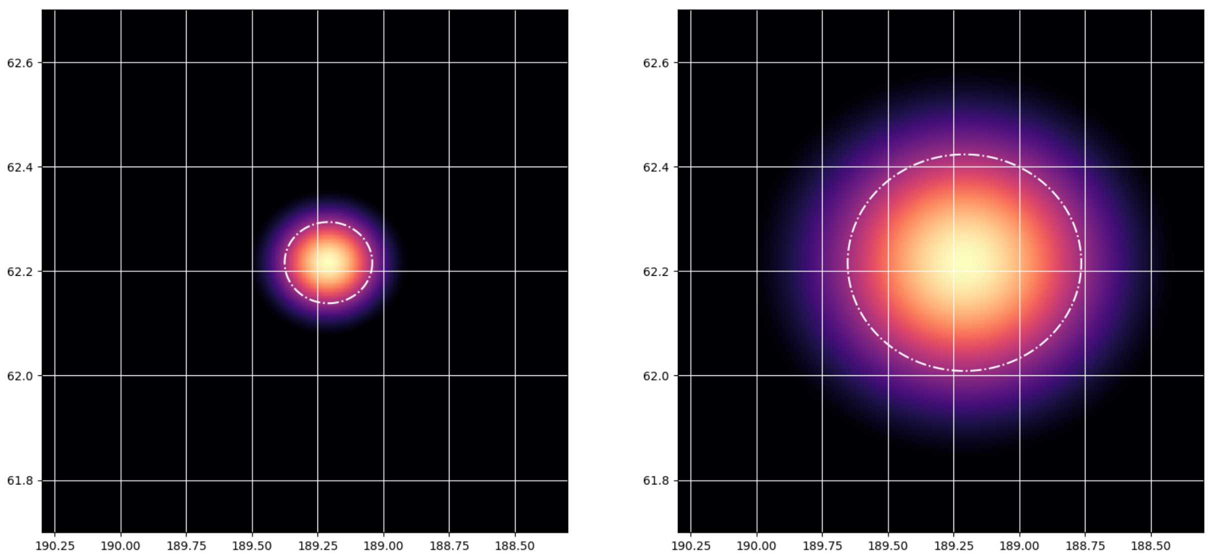
$$\sigma_{\mathrm{Dec}} = (1/\cos(\mathrm{Dec}))\frac{\theta_{1/2}}{(2\sqrt{2\log(2)})} \tag{13}$$

$$\sigma_{\mathrm{RA}} = \frac{\theta_{1/2}}{(2\sqrt{2\log(2)})} \tag{14}$$

   and the full block of Python code implementing the model is:

Telescopes	Country	Diameter (Effective)
Ef	Germany	100 (78)
Wb	Netherlands	25
On	Sweden	25
Nt	Italy	32
Tr	Poland	32
Sv	Russia	32
Bd	Russia	32
Zc	Russia	32
Sh	China	25 (22.5)
Jb1	United Kingdom	76 (67)

Table 1: Sub-sample of EVN telescopes. Effective diameters are derived from fitting to beam models (A. Keimpema priv. comm.)



(a) Robledo 100m single power beam

(b) 32m Medicina power beam

Figure 1

```
def station_HPBW(station,frequency):
        HPBW = ((constants.c/frequency)/station)*(180/np.pi)
        return HPBW

xstd= station_HPBW(stations[telescope],frequency)/(2*np.sqrt(2*np.log(2)))*u.degree*(1/np.cos((phase_centers[i].dec.radian)))
ystd =station_HPBW(stations[telescope],frequency)/(2*np.sqrt(2*np.log(2)))*u.degree
models.Gaussian2D(amplitude=1, \
x_mean=phase_centers[i].ra.degree,\
y_mean=phase_centers[i].dec.degree,\
x_stddev=xstd,\
y_stddev=ystd, theta=0)
```

Examples of the Effelsberg (100m, effective 78m) and Medicina (32m) beams are shown in Figure **??**

2. This code will then grid the model to the observational FoV (for either single or multiple pointings) and then use a user-defined number of FITS images to extract filenames and RA and Declinations.
3. These are evaluated per pointing and per telescope, where the square root is taken so that the value represents the **VOLTAGE** beam. This creates a `pickle` file which contains the corrections for each phase center and each telescope.
4. `apply_clcor_Parseltongue.py` is then used to apply this to all the calibrated data sets. The data set (with only one phase centre) is moved to the current directory and the filename is checked against the filename of the fitsfiles (not pretty I know). The correction values for each telescope and pointing centre are parsed. The correction is applied to each telescope using AIPS Task `CLCOR` with `OPCODE='GAIN'`. This is applied to the upmost CL table.

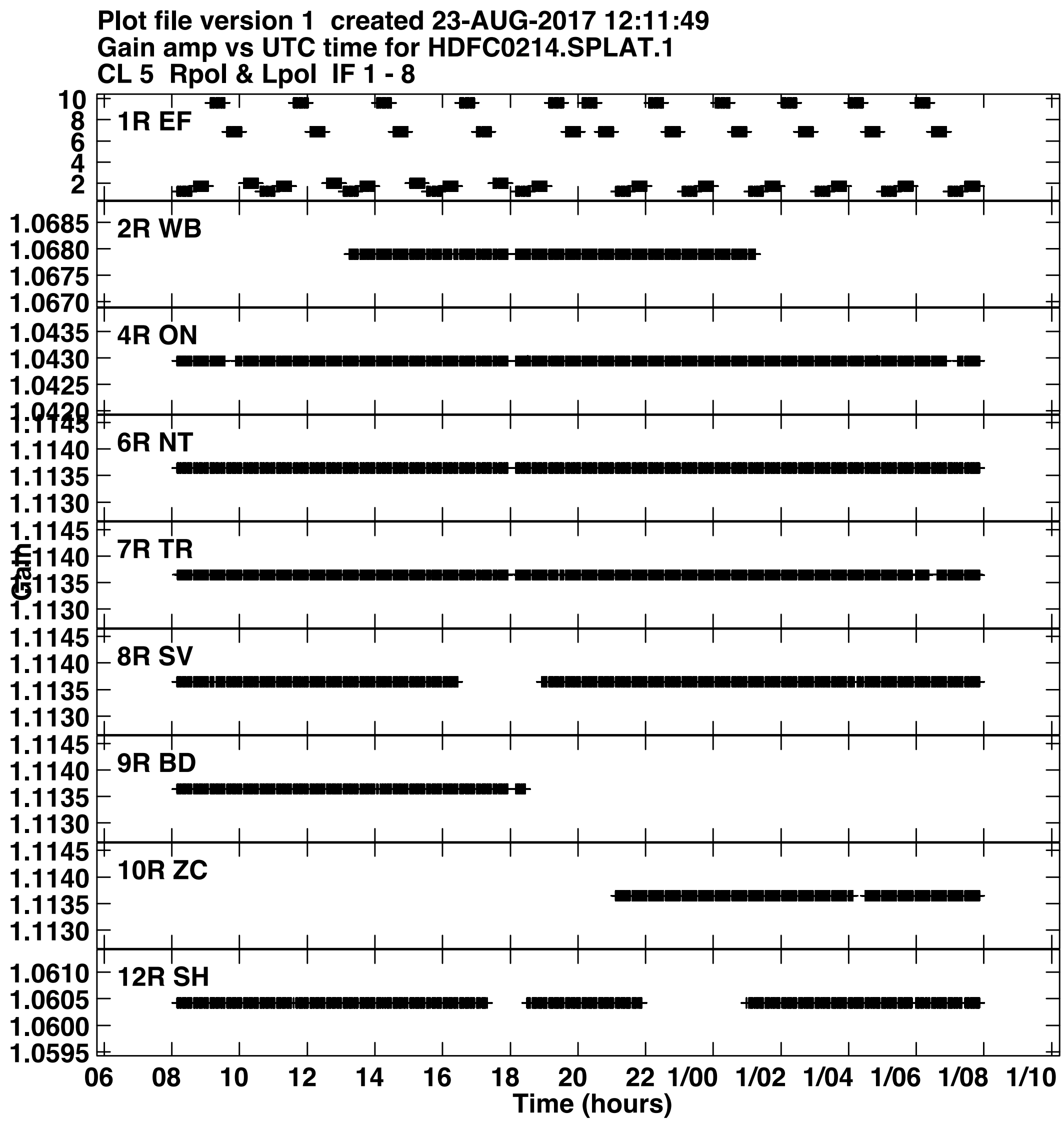


Figure 2: CL table for HDF-N source HDFC0214. The EF telescope nods between 5 different pointings where three are close to the position of the source and 2 are distant ∼20arcmin. These furthest pointings are flagged in the observations subsequently as the primary beam model will diverge rapidly past the HPBW.