Data Driven Radio Astronomy in the SKA era
SKA-low mini project

Max Talberg

March 28, 2024
Word count: 1381

Contents

0.1 Characterising the Gain Calibration Equations

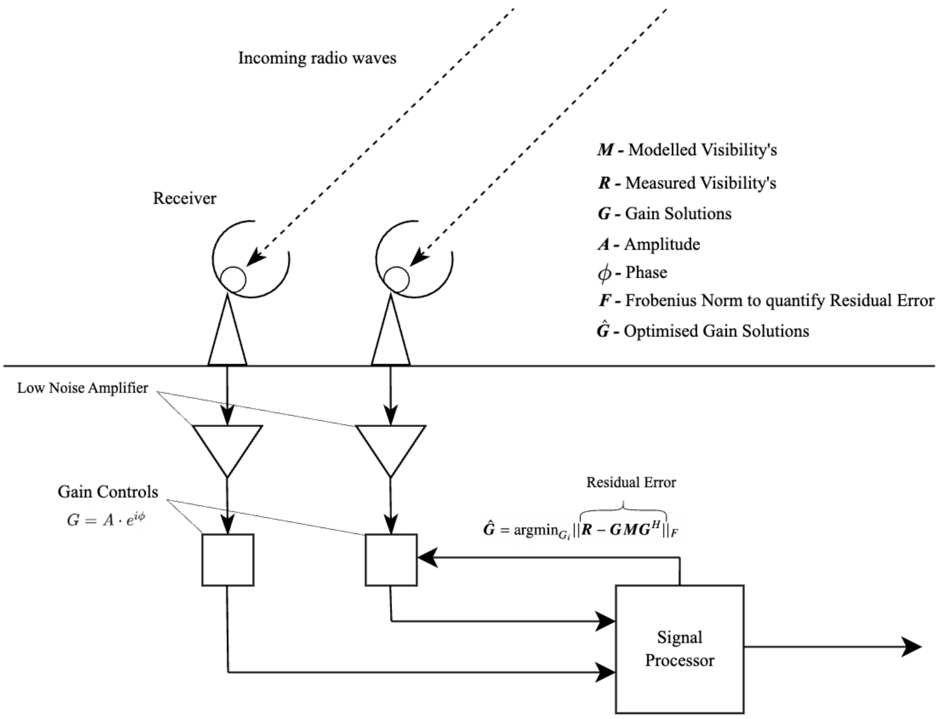

Incoming radio waves
Receiver
M - Modelled Visibility's
R - Measured Visibility's
G - Gain Solutions
A - Amplitude
ϕ - Phase
F - Frobenius Norm to quantify Residual Error
Ĝ - Optimised Gain Solutions
Low Noise Amplifier
Gain Controls
$G = A \cdot e^{i\phi}$
Residual Error
$\hat{G} = \mathrm{argmin}_{G_i} \underbrace{||R - GMG^H||_F}$
Signal Processor

Figure 1: Signal processing flow in radio telescope array calibration. This diagram depicts the sequence from capturing incoming radio waves at the receiver, to amplification by the Low Noise Amplifier, followed by gain adjustments for amplitude and phase, and the combination of signals at the Signal Processor. The calibration equation minimises the Frobenius norm of the residual error, optimising the gain solutions ($\hat{G}$). The residual error quantifies the difference between measured visibility's (R) and the product of the gain solutions (G), the modeled visibility's (M) and the Hermitian transpose of gain solutions (G^H).

In radio astronomy an array of antennas collects radio waves, these signals are calibrated and combined to generate a high-resolution image of the sky. When the radio waves reach an antenna they induce a voltage, the elementary electric pattern (EEP) describes how the antenna responds to the incoming waves as a function of direction. The received signals need to be calibrated to adjust for differences such as response characteristics and environmental factors. The goal of calibration is to correct these differences, ensuring that the combined signal accurately represents the astronomical source.

Characterising the system of equations: Gain calibration is pivotal in this process, the 'gain' of an antenna is its ability to convert incoming radio waves into electrical signals. Here the gain behaves as a corrective measure by variations in amplitude and phase. Equation 1 formalises this optimisation problem:

$$\hat{G} = \arg\min_{G_i} ||R - GMG^H||_F \tag{1}$$

where $\hat{G}$ is an estimator of the complex gain solutions for each antenna, R represents the measured visibility's (the actual signals received by the array), M the modeled visibility's (the expected signal given an astronomical source), G denotes the gain matrix to be optimised, and the superscript H represents the Hermitian transpose. The goal is to minimise the Frobenius norm ($|| \cdot ||_F$), which quantifies the residual error between the measured and modeled signals. Finding the gain matrix that minimised this Frobenius norm value allows for proper calibration of the incoming signals so an accurate visualisation of the sky is obtained.

Properties of matrix M: M is represented as a matrix of the modelled visibility's. These theoretical predictions are based on skies brightness and the EEPs of the antennas. Which are calculated by integrating the product of the sky brightness and the EEPs over the celestial sphere, taking into account the geometric arrangement of the antennas:

$$M_{i,j,p} = \iint \left(F_{i,p}(\theta,\phi) \cdot F^*_{j,p}(\theta,\phi)\right) T_b(\theta,\phi) e^{-jk(\mathbf{r}_i - \mathbf{r}_j)} \sin\theta d\theta d\phi \tag{2}$$

where $*$ means complex conjugate, T_b is the sky brightness temperature, $\mathbf{r}_i$ is the position of antenna i, and $\mathbf{k} = k\sin\theta\cos\phi\hat{\mathbf{x}} + k\sin\theta\sin\phi\hat{\mathbf{y}} + k\cos\theta\hat{\mathbf{z}}$ is the wavevector with the wavenumber k.

The model matrix $M_{i,j}$ correspond to the expected correlation between the i-th and jth antennas. As such the matrix will contain complex terms to account for the amplitude and phase of the received signal.

$$M = \begin{bmatrix} M_{1,1} & M_{1,2} & M_{1,3} \\ M_{2,1} & M_{2,2} & M_{2,3} \\ M_{3,1} & M_{3,2} & M_{3,3} \end{bmatrix} \tag{3}$$

Multiplying gains by phase factor: Multiplying all the gains by a common phase factor $e^{j\phi}$ is equivalent to rotating the phase of every measured signal by the same amount. Since the visibility metric is phase invariant and is a relative measure adding the same phase to all gains has no effect.

Mathematically if the calibration equation $R = GMG^H$, has its gain multiplied by a factor of $\exp(j\phi)$. The new gain matrix, $G' = Ge^{j\phi}$. As such the Hermitian matrix $G'^H = G^* e^{-j\phi}$. Since $e^{j\phi}e^{-j\phi} = 1$, the calibration equation simplifies back to $R = GMG^H$, therefore the theoretical error, $||R - GMG^H||_F$ should remain unchanged.

0.2 Computing the Power of the EEPs and AEP

The EEPs and AEP: The embedding element pattern (EEP), $\vec{F}_i(\theta,\phi)$, is the reception pattern an individual antenna receives by considering only the antenna "i" as active.

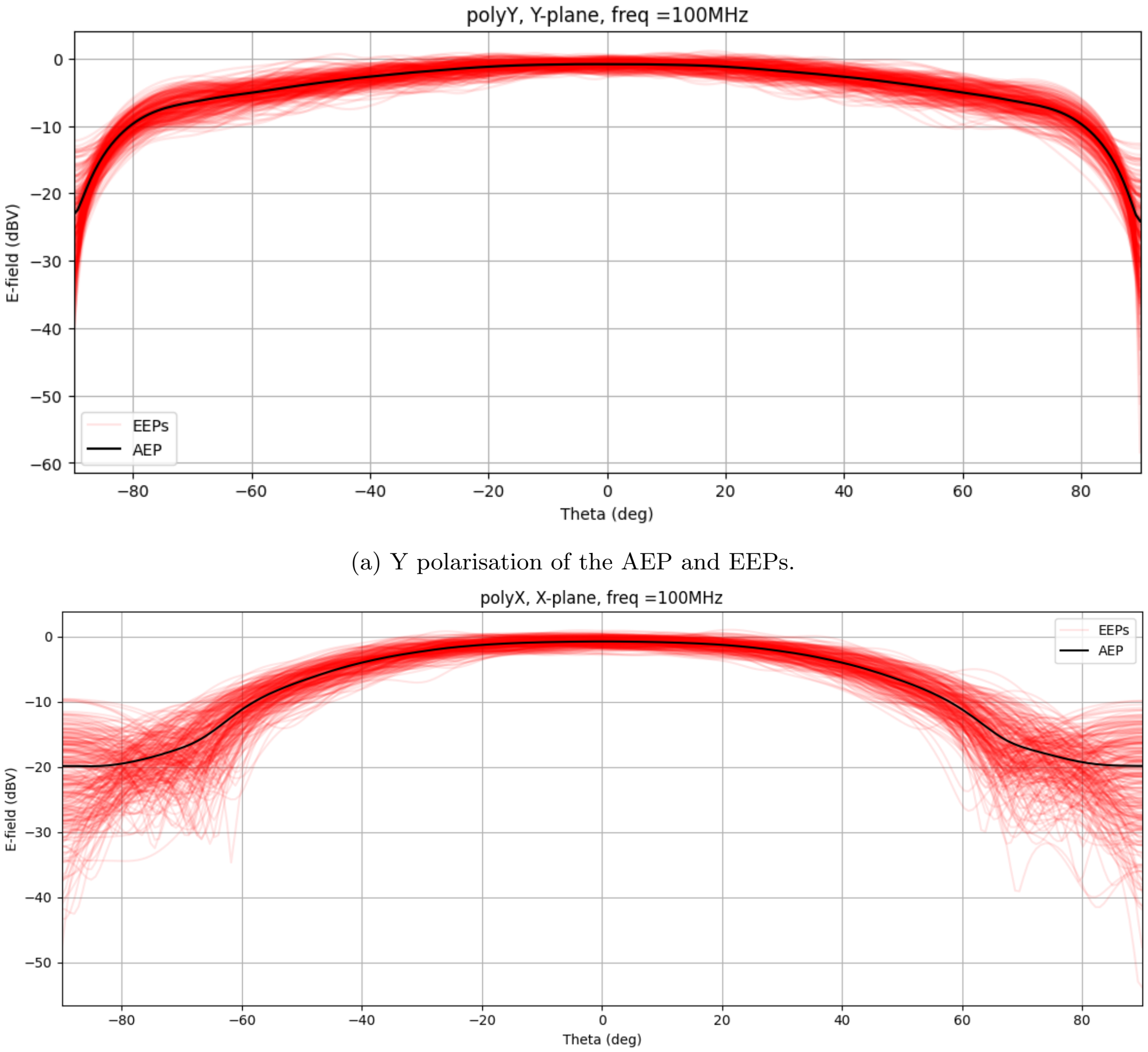
The average element pattern (AEP), $\text{AEP}(\theta,\phi) = \frac{1}{N}\sum_{i=1}^{N} |\vec{F}_i(\theta,\phi)|$, is used to describe the average radiation pattern of individual elements within the array. The AEP is an average of the EEPs as there are multiple EEP values for each θ and ϕ value.

Calculating the EEPs and AEP: In the code the function `compute_EEPs` provides θ and ϕ vector components of the electric field for X and Y polarisation. These vector components are complex valued arrays indicating both the magnitude and phase of the electric field in those directions. The absolute of both the *theta* and *phi* vectors for each polarisation are taken. The magnitude for each absolute electric field is calculated:

$$||\mathbf{E}|| = \sqrt{|E_x|^2 + |E_y|^2 + |E_z|^2} \tag{4}$$

where E_x is the radial component, which in the far field is assumed to be zero (Balanis (2016)). These magnitudes are converted into dBV units for each polarisation. The AEP is calculated as the mean of these magnitudes. These values are plotted against the θ values initially used to generate the vector components.

Variability: The plot exhibits a degree of variability in the individual EEPS, depicted by the red lines. The variability suggests differences in how each antenna is receiving signal. The smooth AEP lines show the array as a whole has predictable average behaviour, despite the individual variability.



(a) Y polarisation of the AEP and EEPs.

(b) X polarisation of the AEP and EEPs.

Figure 2: Analysis of the EEPs and the AEP in the $\phi = 0$ plane at 100 MHz and 256 samples. The plot illustrates the variability in the EEPs for each individual antenna element (shown in red), and the smoothed average response of the array (shown in black), across a range of theta angles, $\theta = [-90, 90]deg$.

0.3 Implementation of the StEFCal Algorithm

Algorithm 1 Algorithm StEFCal

```
Initiate $G[0]$; $G[0] = I$ is adequate in most cases
for $i = 1, 2, \ldots, i_{\max}$ do
    for $p = 1, 2, \ldots, P$ do
        $z \leftarrow G[i-1] \cdot M_{:,p} \equiv g^{[i-1]} \circ M_{:,p}$
        $g_p \leftarrow (R^H_{:,p} \cdot z)/(z^H \cdot z)$
    end for
    if $\mathrm{mod}(i, 2) = 0$ then
        if $\|g[i] - g[i-1]\|_F / \|g[i]\|_F \leq \tau$ then
            Convergence reached
        end if
    else
        $G[i] \leftarrow (G[i] + G[i-1])/2$
    end if
end for
```

The StEFCal algorithm: The StEFCal algorithm 1 (Salvini & Wijnholds (2018)) aims to solve the system in Equation 1 and estimate the per-antenna complex-valued gain iteratively. This is achieved by initialising an identity matrix for the gain ($G^{[0]}$), then iterate (i) for some maximum iteration number (i_{max}) through each antenna (p). In each antenna, the gain is updated by calculating a temporary variable (z), which involves the element-wise product of the previous gain estimate ($G^{[i-1]}$) and the model visibility matrix ($M_{:,p}$). The gain antenna at index p (g_p) is calculated with the measured visibility ($\hat{R}^H_{:,p}$) and the temporary variable. Convergence is checked for even iterations ($\mathrm{mod}_2(i)$) by comparing the current gain estimate with the previous one. If the relative change ($||g^{[i]} - g^{[i-1]}||_F/||g^{[i]}||_F$) is below a threshold ($\tau$), convergence is assumed. If convergence is not met, the gains are smoothed by averaging the current and previous estimates ($(G^{[i]} + G^{[i-1]})/2$).

Average of current and previous odd iteration: The averaging step occurs every even iteration. The gains are smoothed according to condition $(G^{[i]}+G^{[i-1]})/2$, where $G^{[i-1]}$ is the gain solution of the previous odd iteration. The averaging step at every even iteration aids the convergence efficiency. Without this, the iterative process might converge slowly and bounce between different solutions. By updating the gain solution at each even iteration to be the average of the most recent solution and the previous odd iteration, the algorithm has a smoothing effect and mitigates these convergence problems. This simple process makes the iteration both fast and robust.

Convergence rate: The StEFCal algorithm was implemented in a function called `stefcal`, with a threshold value of $\tau = 10^{-5}$ following the tolerance used by Salvini & Wijnholds (2018). The StEFCal Algorithm 1 converged in 231 and 193 iterations for M_{AEP} and M_{EEPs} respectively. Inspecting the convergence term $\|g[i] - g[i-1]\|_F / \|g[i]\|_F \leq \tau$ and Figure 3, the algorithm initially converged close to this τ value in the first few iterations iterations. A smaller τ term would result in similar answers for less iterations.

Optimising StEFCal Salvini & Wijnholds (2018) suggested an optimised StEFCal algorithm, by employing the current gain matrix $G^{[i]}$ rather than the previous gain matrix $G^{[i-1]}$ when calculating z.

The second algorithm checks for convergence every iteration, although this was omitted in the code to provide a smooth comparison to Algorithm 1 and reduce oscillations in results. Furthermore, this reduces the computational overhead of a calculation at every iteration. The optimisation employed to the StEFCal algorithm ensures the algorithm adapts quicker to changes in gain, this speeds up the iterations at the cost of stability.

With the same tolerance value $\tau = 10^{-5}$ used for Algorithm 1, Algorithm 2 converged in 143 and 121 iterations for M_{AEP} and M_{EEPs} respectively. An improvement when compared to Algorithm 1's 231 and 193 iterations respectively.

Algorithm 2 Algorithm StEFCal2

```
Initiate $G^{[0]}$, $G^{[0]} = I$ is adequate in most cases
for $i = 1, 2, \dots, i_{\max}$ do
    $G^{[i]} = G^{[i-1]}$
    for $p = 1, 2, \dots, P$ do
        $Z \leftarrow G^{[i]} \cdot M_{:,p}$, $g^{[i]}_{:,p} \leftarrow g^{[i]} \circ M_{:,p}$
        $g^{[i]}_p \leftarrow \frac{(R^H_{:,p} \cdot Z)}{(Z^H \cdot Z)}$
    end for
    if $\left\|g^{[i]} - g^{[i-1]}\right\|_F / \left\|g^{[i]}\right\|_F \leq \tau$ then
        Convergence reached
    end if
end for
```

0.4 Absolute Error in the Calculated Gain

Implementing the two model matrices M_{AEP} and M_{EEPs} in the StEFCal algorithm produces estimates for the respective gains G_AEP and G_EEPs. The accuracy of these results is assessed by calculating the absolute error of the gain solutions and the amplitude and phase as a function of iteration number.

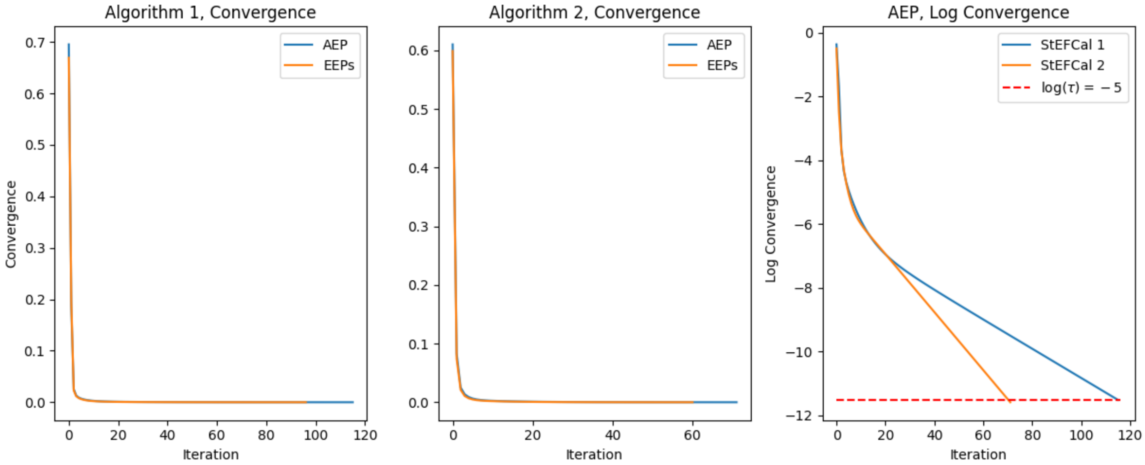
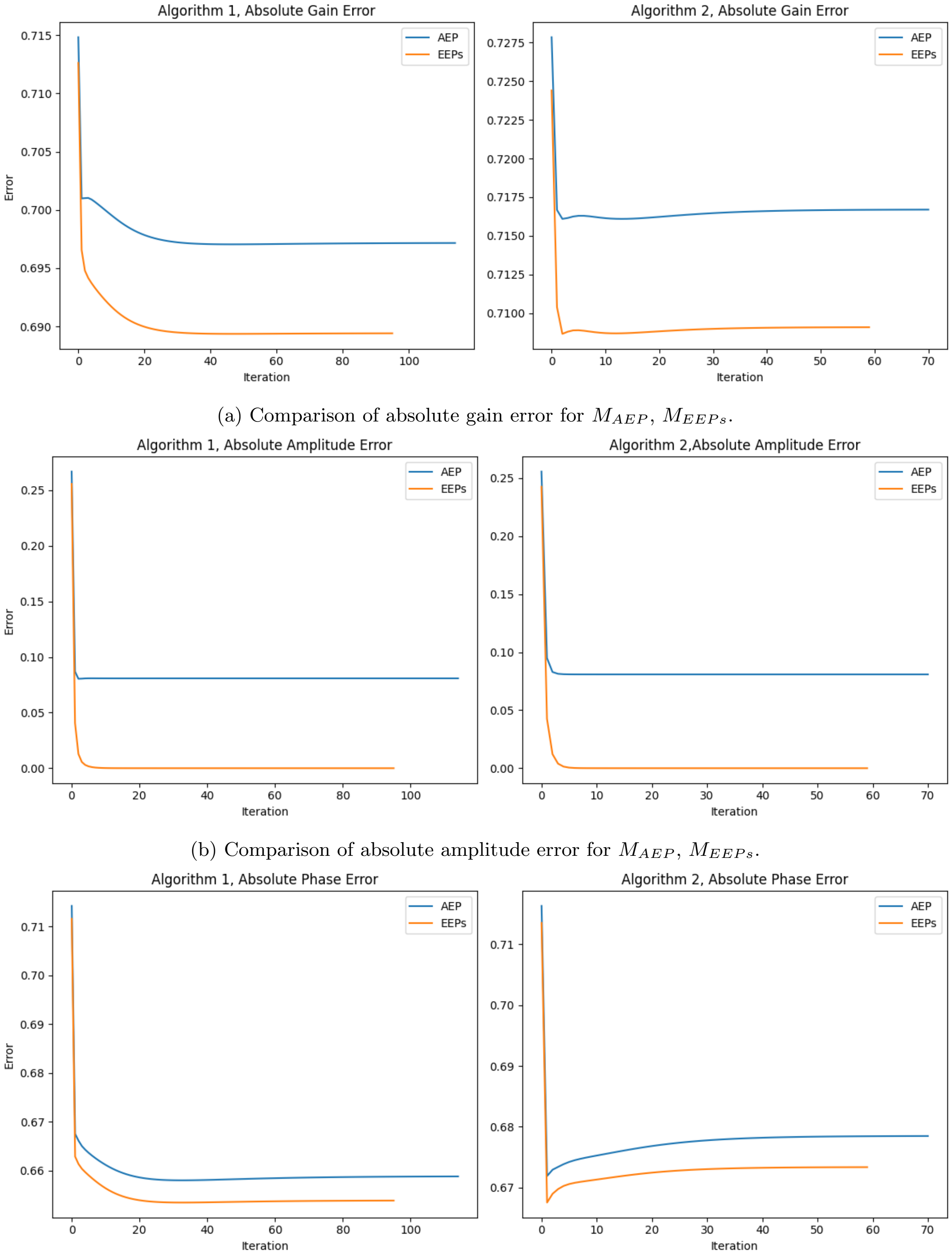


Figure 3: Comparison of algorithm convergence for M_{AEP}, M_{EEPs}.



(a) Comparison of absolute gain error for M_{AEP}, M_{EEPs}.

(b) Comparison of absolute amplitude error for M_{AEP}, M_{EEPs}.

(c) Comparison of absolute phase error for M_{AEP}, M_{EEPs}.

Figure 4: Comparison of absolute errors over M_{AEP} and M_{EEPs}, including the Absolute Gain, Amplitude, and Phase Errors using the different StEFCal Algorithms plotted against iterations.

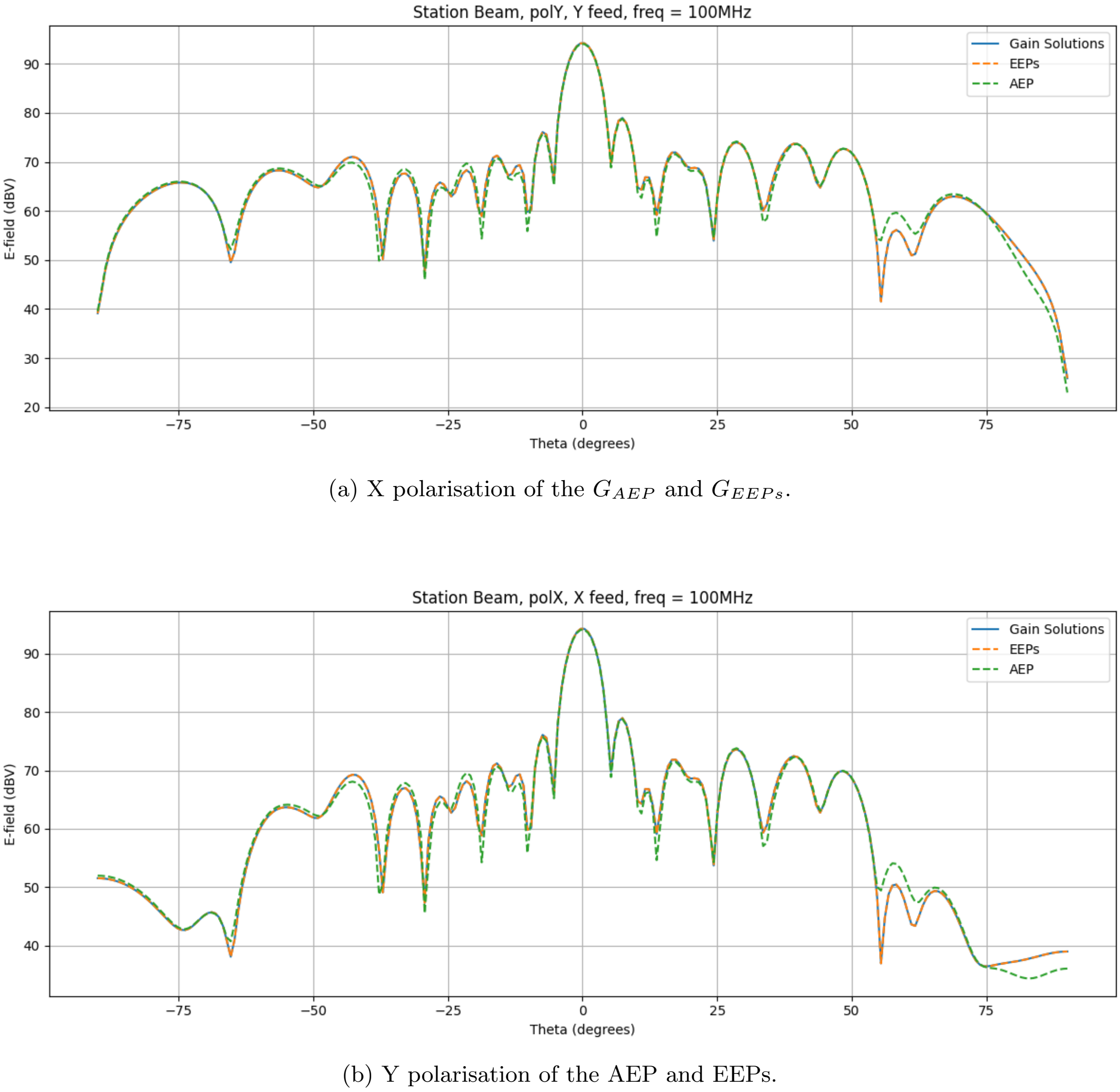
Following Salvini & Wijnholds (2018) a threshold value of 10^{-5} was used to asses convergence. All the absolute errors rapidly decreased in the first 25 iterations and then converged to the desired threshold, with the AEP values taking slightly longer to converge. It is evident that Algorithm 2 converged faster than Algorithm

1. The converged absolute error for the gain, amplitude and phase were all slightly higher for Algorithm 2. The gain and phase exhibited behavior of dropping to a minimum error quite quickly and then bouncing up. Perhaps this represents the instability of updating the algorithm with the current gain value.

The minimum error achieved by M_{AEP} and M_{EEPs} is that from the absolute amplitude error.

0.5 Array Patterns from the Gain Solutions

Beamforming all 256 voltages to zenith and plotting the power (in dBV) of the beamformed voltages in the plane $\phi = 0$ generated the array pattern below. The most accurate gain solutions from the StEFCal algorithm was used, these were the gain values generated from Algorithm 1.



(a) X polarisation of the G_{AEP} and G_{EEPs}.

(b) Y polarisation of the AEP and EEPs.

Figure 5: Beamformed array patterns in polarisation Y (top) and polaszization X (bottom) using different gain solutions from Algorithm 1. The solutions include the provided true solution (Gain Solutions), and the estimations obtained from the matrices M_{AEP} and M_{EEPs} . Each curve represents the power in dBV as a function of θ in the zenith plane $\phi = 0$. The gain solutions derived from M_{AEP} and M_{EEPs} matrices show varying levels of reconstruction error compared to the true solution.

The array patterns were produced using the following equation:

$$\vec{P}(\theta,\phi) = \sum_{i=1}^{N_{\text{ant}}} \hat{w}_i c_i \vec{F}_i(\theta,\phi) e^{-jk(\sin\theta\cos\phi x_i + \sin\theta\sin\phi y_i)}, \tag{5}$$

where x_i, y_i are the position of antenna i and $\hat{w}_t$ is $\exp jk(\sin\theta_0 \cos\phi_0 x_i + \sin\theta_0 \sin\phi_0 y_i)$, where θ_0 and ϕ_0 are set to 0. The calculated EEPs gain exhibits no obvious reconstruction error and closely follows the gain solution. The calculated AEP gain depicts a small amount of reconstruction error, particularly just after 50 deg, although for the most part follows the solution closely.

0.6 Array Pattern of Calibrated Station Beam

The most accurate station beam is the calculated EEPs gain, this is plotted below for both the X and Y feed with a steer direction of $(\theta, \phi) = (40, 80)$.

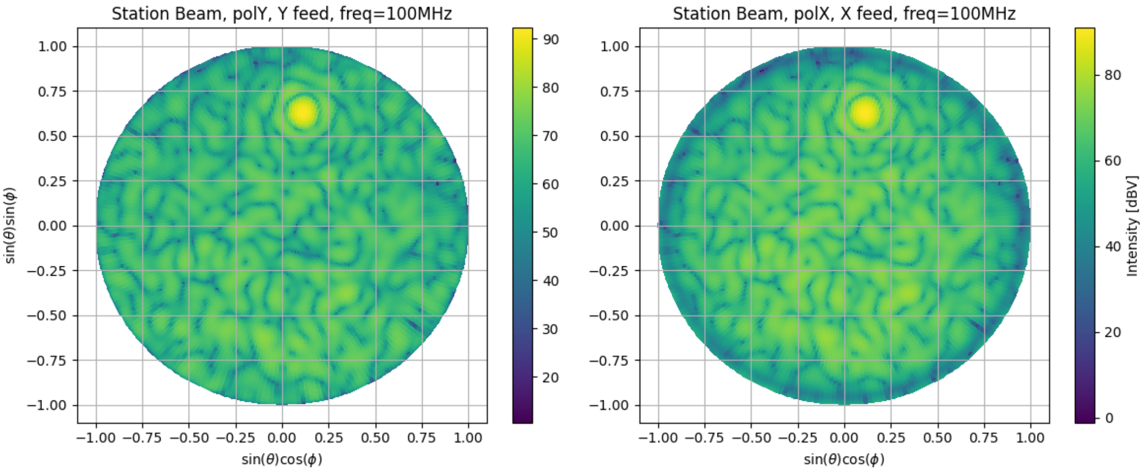


Figure 6: Station beam patterns for both the X and Y feed. The beam is steered in the direction of (θ, ϕ) $= (40, 80)$. This plot depicts the intensity distribution in dBV within the sine-cosine projection space, where the x-axis represents $sin(\theta)cos(\phi)$ and the y-axis represents $sin(\theta)sin(\phi)$. The pattern exhibits the complex structure of the beam, with the main lobe reflecting the beam's directional response at the specified steering direction.

0.7 Appendix

0.7.1 Auto-generative tools

This project has utilised auto-generative tools in the development of documentation that is compatible with auto-documentation tools, latex formatting and the development of plotting functions.

Example prompts used for this project:

- Generate doc-strings in NumPy format for this function.
- Latex code for a subplot.
- Latex code for a 3 by 3 matrix.
- Python code for a 2 by 1 subplot.

Bibliography

Balanis, C. A. 2016, Antenna theory: analysis and design (John wiley & sons)

Salvini, S., & Wijnholds, S. J. 2018, Astronomy & Astrophysics, doi: `1410.2101`