Models used for MWA dipole and tile beams.

2016 Full Embedded Element Beam Model (Tim Colegate / Marcin Sokolowski)

WARNING : it slowly becomes obsolate as there the MWA beam model will not be available in github under MWATelescope repository

However, some parts of this document are still relevant (perhaps not the links) - please start using github version instead

It is uses a "full embedded element model", where corner, edge, and interior dipoles have different responses on the sky. The embedded element patterns are save in a .h5 file which is a table of spherical harmonic coefficients which characterise the dipole responses for each coarse channel. You can apply delays to the different coefficients (extra phase = exp(i*2*pi*nu*delay)) to create a beam for any beamformer pointing. The model is described in this paper https://arxiv.org/abs/1710.07478. This model has been implemented in :

- **MWA_Tools in python (Marcin Sokolowski / Tim Colegate).:**
  - **Documentation / manual :** 2016_beam_model.pdf
  - **The main scripts are :**
    - make_beam.py
    - beam_correct_image.py
    - mwa_sensitivity.py
- **Standalone python version (Marcin Sokolowski / Tim Colegate). This version was implemented mainly for external users without access to MWA_Tools using publicly available MWA data.**
  - **Documentation / manual :** MWA_beam_standalone_software.pdf
  - **Download (temporary location):**
    - Full standalone version (~150 MB) can be downloaded from : https://www.dropbox.com/s/cuzo57fcegnwldq/standalone.tar.gz?dl=0 (this link will be updated to something more permanent)
    - Only the python code without H5 file and examples : https://www.dropbox.com/s/ct6l967407ovyow/standalone_code.tar.gz?dl=0
  - **The main scripts are :**
    - make_beam_test.py
    - beam_correct_image.py
    - mwa_sensitivity.py
- **RTS in C (Bart Pindor)**
- **Andre's tools (anoko) in calibrate/pbcorrect/beam and other programs (C++ version by Marcin Sokolowski / Andre Offringa)**
- **An on-line version of MWA sensitivity calculator based on mwa_sensitivity.py script is being developed (Andrew Williams / Marcin Sokolowski). Currently there is a prototype available for CIRA/IVEC network users (http://bighorns.pawsey.org.au/bighorns/mwa/calculators/sensitivity/mwa_sensitivity.php or http://202.8.37.217/bighorns/mwa/sensitivity/sensitivity/mwa_sensitivity.php) .**
- **showspec software integrating antenna pattern in FEKO output format (or other text formats with at least 3 columns for phi, theta and power) with Global Sky Model (GSM). It also enables integration of the newest MWA 2016 beam model with GSM sky model:**
  - Showspec documentation showspec_standalone.pdf

- showspec_standalone.tar.gz - Showspec software and documentation (wihtout GSM source code and data files)
- gsm.tar.gz - Local version of Global Sky Model source code (modified) and data files / GSM software - original version / GSM model webpage

Average Embedded Element (2014 Sutinjo) Beam Model

Adrian Sutinjo performed FEKO simulations of the dipoles. He used an average of the sky sensitivity of each dipole: an "average embedded element model". They have 3-degree resolution in azimuth and elevation and simulate the dipole pattern.

Originally he made a simulation for every 7.68 MHz. This was quickly modified to include 1.28-MHz resolution simulations across 120--200MHz, a commonly-used frequency range within the MWA team.

In make_beam.py (MWA_Tools), there is no frequency interpolation across these channels. The working files are "Jmatrix.fits" and "Zmatrix.fits": Jmatrix.fits gives you the Jones matrix for the average antenna pattern. You change the tile beam pattern by changing the array factor (Equation 12 in Sutinjo et al 2015.) Zmatrix.fits is the impedance matrix (Equation 13 of the same paper).

In Andre Offringa's beam (anoko/mwa-reduce), there is frequency interpolation across these channels.

2013 Analytic Beam

Prior to 2014, the majority of MWA software used the "analytic" beam model, details of which are below.

- See AntennaDevelopment for information on measured dipole and tile beams.
- See AntennaMeullerMatrices for some sims by Randall.

Dipole Beams

MAPS and the RTS currently use the following direction-dependent Jones matrix to describe the conversion at a dipole from sky polarization coordinates *x,y* to instrument polarization coordinates *p,q* (alligned North/South and East/West respectively):

$$\begin{aligned} J &= \sin(2\pi h\cos(za)) \begin{bmatrix} J_{px} & J_{py} \\ J_{qx} & J_{qy} \end{bmatrix} \\ &= \sin(2\pi h\cos(za)) \begin{bmatrix} \cos(lat)\cos(\delta)+\sin(lat)\sin(\delta)\cos(HA) & -\sin(lat)\sin(HA) \\ \sin(\delta)\sin(HA) & \cos(HA) \end{bmatrix}, \end{aligned}$$

where the scalar describes the effect of the ground plane; *h* being the height of the dipole above the ground plane (in wavelengths) and *za* the zenith angle. There is also a direction independent complex gain for each dipole (taking into account dipole lengths, cable lengths, etc.). The power gain for each instrument signal path associated with the direction-dependent Jones matrices is given by the diagonal elements of

$$J.J^{\dagger},$$

and for an unpolarized signal, these terms simplify to

$$\begin{aligned} G_{pp} &= \left[\sin(2\pi h\cos(za))\sqrt{1-(\sin(za)\cos(az))^2}\right]^2 \quad \text{and} \\ G_{qq} &= \left[\sin(2\pi h\cos(za))\sqrt{1-(\sin(za)\sin(az))^2}\right]^2, \end{aligned}$$

where *az* is the azimuth. This is the far-field expression for a short dipole (which can be derived from Eqn. 4-10 in Krauss 1st Ed. 1950).

Tile Beams

For tile beams, 16 dipole beams are multiplied by geometric phase terms, beamformer phase terms, and complex gains, then added together.

Typical Parameters

- ED tiles:
- X1, X2, X3 tiles:
- Short/Wides:

Factorizing the Jones matrix

The Jones matrix above contains both a rotation component and a foreshortening component. This can be split into unpolarized gain terms and polarization rotation terms:

$$\begin{bmatrix} J_{px} & J_{py} \\ J_{qx} & J_{qy} \end{bmatrix} = \begin{bmatrix} g_p & 0 \\ 0 & g_q \end{bmatrix} \begin{bmatrix} \cos(\psi_p) & \sin(\psi_p) \\ \cos(\psi_q) & \sin(\psi_q) \end{bmatrix}$$

where the angles made by the two receptors on the sky axes can be taken from the full Jones matrix

$$\psi_p = \tan^{-1}\left(\frac{-\sin(lat)\sin(HA)}{\cos(lat)\cos(\delta)+\sin(lat)\sin(\delta)\cos(HA)}\right) \text{ and}$$

$$\psi_q = \tan^{-1}\left(\frac{\cos(HA)}{\sin(\delta)\sin(HA)}\right).$$

Issues

- No mutual coupling.
- Need to work in dipole rotations for imperfectly placed dipoles.

-- DanielMitchell - 22 Jul 2008

- MWA_beam_standalone_software.pdf: MWA_beam_standalone_software.pdf
- 2016_beam_model.pdf: 2016_beam_model.pdf
- showspec_standalone.pdf: Showspec software documentation (integration of GSM model with antenna pattern)
- showspec_standalone.tar.gz: Showspec software and documentation (wihtout GSM source code and data files)
- gsm.tar.gz: Global Sky Model source code (modified) and data files

This topic: Main > WebHome > PrimaryBeamModels
History: r46 - 29 May 2018 - 03:29:46 - MarcinSokolowski ?