# A report on

# Characterization of Broadband Feeds for GMRT

By,

(Aniruddha Sonde)

BITS Pilani, K. K. Birla Goa Campus
Zuarinagar, Goa, 403726.

Under the supervision of

B. Hanumanth Rao
Engineer – C,
Feeds and Front End Group,
GMRT, NCRA-TIFR

NCRA • TIFR

**Gaint Meterwave Radio Telescope**
**National Centre for Radio Astrophysics**
**Tata Institute of Fundamental Research**
**Narayangoan**

Abstract

GMRT in its up gradation plans aims to operate seamlessly over the entire frequency range of 150 to 1500 MHz. Characterizing the prototype feeds is quintessential to keep up gradation plans for GMRT moving forward. The prototype Feed antennas will be mounted at the parabolic antenna's prime focus and will be tested for its performance using on and off Power levels, HPBW (Half Power Beam Width), etc. The sources used are Astronomical calibrators CYGNUS-A, CASA, VIRGO, and CRAB. The data collected is then analysed and the inferences are followed up with recommendations to optimize, improve the feed performances. The testing and characterization of the GMRT feed antennas needs to be conducted to guarantee the smooth working of this world class instrument. This report entails Scan test analysis using MATLAB to get the deflection measurement results and HPBW for the experimented Feed antenna.

# Contents

# List of Figures and Tables

# 1. Introduction

Since the Second World War, fuelled by military advancements, Microwave Engineering has matured from the fundamentals of electromagnetic theory to its ubiquitous use in communications, radar systems, remote sensing, medical systems and radio-astronomy. In the pursuit of extracting more information from our environment and the universe there is an increased focus on developing low noise, wide-band systems. Radio-astronomical applications have been critical in the development of these systems. Radio-astronomical signals are very much weak in nature. So in order to capture these signals, the receivers in radio telescopes have to be highly sensitive. The sensitivity of a radio telescope is indispensably factored by the choice "feed" antenna. This report entails the testing and characterization of broadband feeds for the GMRT up gradation. Characterizing the prototype feeds is quintessential to keep the GMRT up gradation plans moving forward.

## 1.1Giant Meterwave Radio Telescope (GMRT)

The National Centre for Radio Astrophysics (NCRA) has set up a unique facility for radio astronomical research using the meter-wavelengths range of the radio spectrum, known as the Giant Meterwave Radio Telescope (GMRT), GMRT consists of 30 fully steerable gigantic parabolic dishes of 45m diameter each spread over distances of 25 km. The antennas have been constructed using a novel technique (nicknamed SMART) and their reflecting surface consists of panels of wire mesh. This panel is attached to rope trusses, and by appropriate tensioning of the wires used for attachment the desired parabolic shape is achieved. The meter wavelength part of the radio spectrum has been particularly chosen for study with GMRT because man-made radio interference is considerably lower in this part of the spectrum in India. And there are many outstanding astrophysics problems which are best studied at meter wavelengths.

Of the many configurations of a radio telescope antenna, the most popular is a reflector antenna fed with a feed antenna. A wide variety of feed antennas is available ranging from dipoles to corrugated horns. The GMRT has prime-focus feeds at four faces of a rotating turret, covering frequency bands at 150, 233, 327, 610 & 1420 MHz (one turret face houses a dual frequency feed operating at 233 & 610 MHz), with bandwidths of a few tens of MHz, except for the 1420 feed that covers 1000 to 1450 MHz. The GMRT is going through an upgrade to provide seamless frequency coverage from 50 to 1500 MHz, with a maximum instantaneous bandwidth of 400 MHz for increased sensitivity. All these feeds provide dual polarization outputs. Front End elements are crucial to the sensitivity of a radio telescope.

# 2. Feed Testing and Characterization

The Feed Antenna and Front End system tests include measurement of return loss, testing of the Low Noise Amplifiers (LNAs) for their linearity, sensitivity test, deflection measurement at antenna base and measurement of HPBW of the feed with reflector system.

## 2.1 Deflection Measurement Tests

The deflection test is used to check the health or performance of the GMRT antenna which finds out whether the antenna gives the desired amount of deflection against a strong source.
The tests are done at strong sources like Cas-A, VIRGO, Cyg-A & CRAB. The dominant sources seen in the radio sky depend on the observing frequency. Typically the Sun, supernova remnants, radio galaxies, the Milky Way are some of the brightest sources at metre and cm wavelengths. The quiet Sun has a typical flux density of 10E5 Jy[1] while the next strongest sources are the radio galaxy Cygnus-A (Cyg-A) and the supernova remnant Cassiopeia-A (Cas-A), both of which have flux densities of $10^4$ Jy. (A source of 1 Jy produces a signal of only ~ 7E-17 Watts)

Table 1.1 Source Deflections

| Source and deflection (dB) | | | | |
|---|---|---|---|---|
| Frequency (MHz) | Virgo | Cyg-A | Casa-A | CRAB |
| 150 | 2.1 | 8.3 | 8.2 | 3.4 |
| 235 | 3.3 | 10.4 | 10.6 | 4.8 |
| 325 | 4.5 | 12.4 | 12.7 | 6.6 |
| 610 | 3.4 | 10.7 | 11.1 | 6.5 |
| 1060 | 2.5 | 8.7 | 9.5 | 6.1 |
| 1170 | 2.3 | 8.3 | 9.2 | 6.0 |
| 1280 | 2.2 | 7.9 | 8.9 | 5.9 |
| 1390 | 1.9 | 7.3 | 8.3 | 5.5 |

[1] The unit of Flux density is the Jansky (Jy): 1 Jy = $10^{-26}$ Watts $m^{-2}$ $Hz^{-1}$. Typical units include milliJansky (mJy) and microJy (μJy).

Deflection test on the C-10 Antenna's Cone-Dipole Feed 550-900 MHz (20/01/2015)

For this deflection test the *cable length* between the feed and the LNA and the Quadrature Hybrid was reduced to 0.3 mts. The desired deflection for the source Cyg-A at 610 MHz is 12 dB.

Table 2.1 Source Deflections at Cyg-A for 550-900 Mhz feed at C10 Antenna

| Source: Cyg- A | Channel 1 | | | Channel 2 | | |
|---|---|---|---|---|---|---|
| Frequency(MHz) | On Source (dBm) | Off Source(dBm) | Deflection (dB) | On Source (dBm) | Off Source (dBm) | Deflection (dB) |
| 610 | -67 | -78 | 11 | -68 | -80 | 12 |
| 700 | -71 | -82 | 11 | -72 | -83 | 11 |
| 750 | -74 | -84 | 10 | -75 | -85 | 10 |
| 8000 | -77 | -85 | 8 | -76 | -86 | 8 |

## 2.2 Antenna Test Range

Cone Dipole Feed for the 550-900 MHz band was tested at the NCRA Antenna Test Range and the subsequent E-H patterns at 610, 700, 750, 800 MHz were analyzed. Identical Vertical and Horizontal polarization patterns are desired.

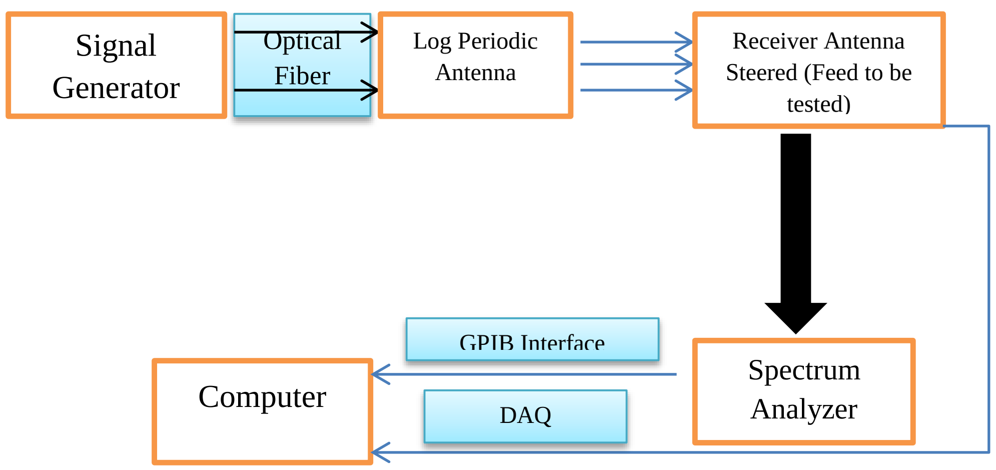
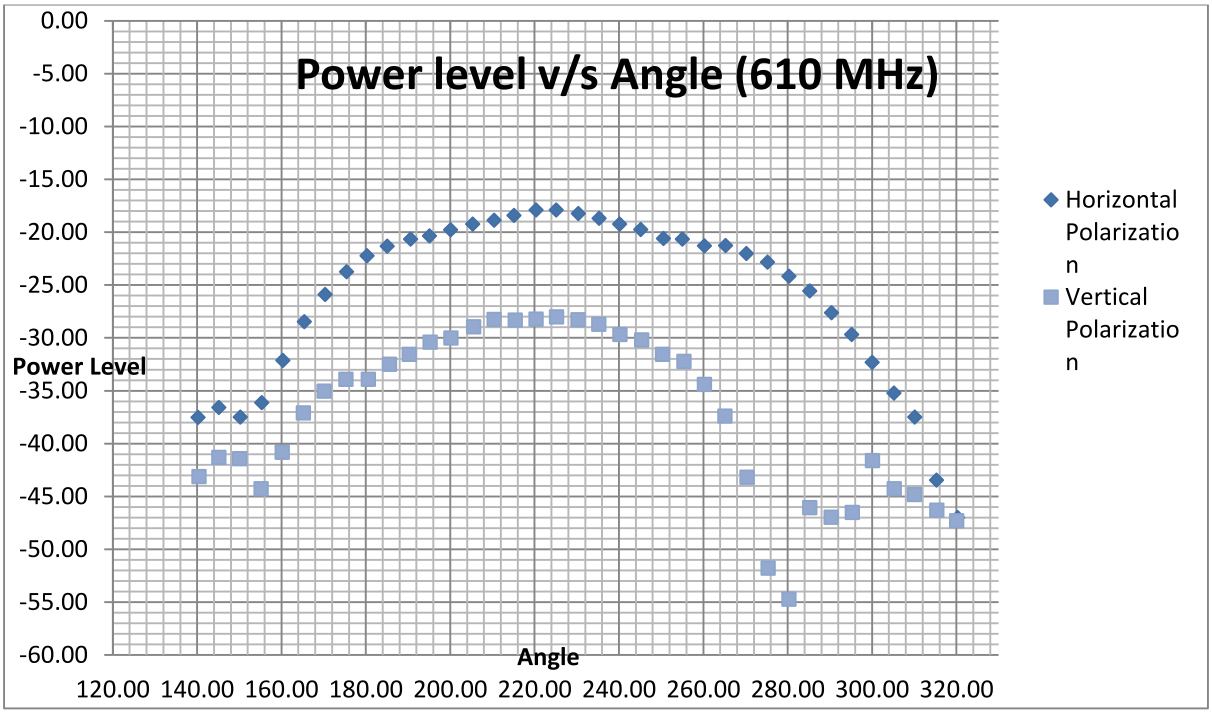
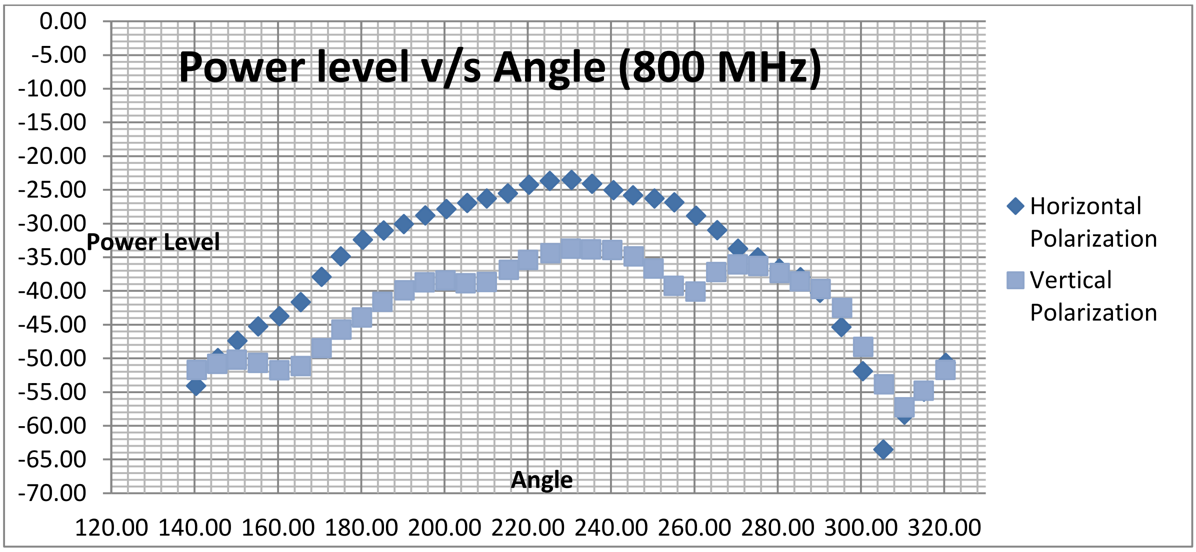



Fig. I. Block Diagram of Antenna Test Range

$T_x$ : Log Periodic Antenna
$R_x$ : Cone-Dipole Feed
Line of Sight (LOS) Angle: 230.14 degrees (approx.)
Range of sweep: -90 to +90 degrees (140 - 320 degrees)
Cone Dipole Feed rotated through 90 degrees for Horizontal Polarization w.r.t intial set-up

Fig. 1.1 Power levels v/s Angle, E,H Patterns, Antenna Test Range



## 2.3 Scan Test

In a scan test, the antenna is steered in the elevation angle and/or in the azimuth direction from OFF source to ON source, to OFF source. The default scan test settings include a scan cycle of 16 minutes; the antenna is rotating at 50 arc-min / min (Slew Rate). The scan test helps in the calculation of the HPBW which is an important characteristic of an antenna. This beam-width is used to describe the resolution capabilities of the antenna to distinguish between two adjacent radiating sources or radar targets.

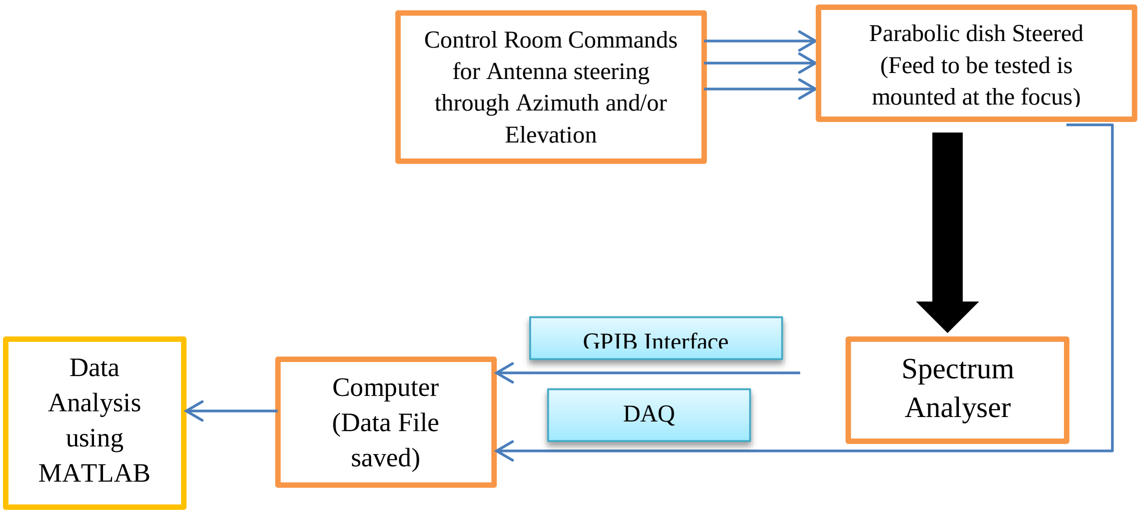
- The data is acquired through a LabView DAQ device and a Data Acquisition Code. Once the data has been logged into a '.dat' or a '.txt' file it is imported in MATLAB for further data analysis.

- The file imported is separated into numeric and non-numeric data to remove the headers involved in the data logging. One of the merits of the corresponding MATLAB code is its independence from the type of Spectrum Analyzer used for collecting the data.

- The numeric data which is a matrix of power levels at different frequencies over the scan cycle is the processed to get deflection measurement results and HPBWs at various frequencies.

- The data at a particular frequency over the scan cycle time is Gaussian fitted to get a Power level v/s angle[2] plot. The HPBW (3dB beam width) is then calculated for the data.

Fig. II. Block Diagram of Scan Test on GMRT Antenna




[2] Antenna Rotation Speed = 50 arcmin/min; Total time approx. 16 mins. ; Total scan angle = Antenna Rotation Speed * time

## 2.4 Data Analysis

The columns of the data imported correspond to the power level at various frequencies throughout the range and rows correspond to the time over the scan cycle. The power levels at each frequency in the range follow a profile which closely represents a Gaussian. The data measures the power in the unit of dB, and is converted to linear units before using Gaussian fitting algorithm. The curve to be fitted is:

$$y = (yp)*exp( -(xdata - mu).\hat{}2 / (2*sigma\hat{}2)) + ymin$$

yp, mu, sigma and ymin are the parameters that form this non-linear equation. The data is then fitted using non-linear regression techniques after initial guesses for these parameters are given to the program. The HPBW for a Gaussian is related to its sigma according to,

$$HPBW = 2.36*sigma$$

The figure also plots the expected HPBW. The function we use for plotting expected HPBW for GMRT dishes of diameter D=45m is (x in MHz),

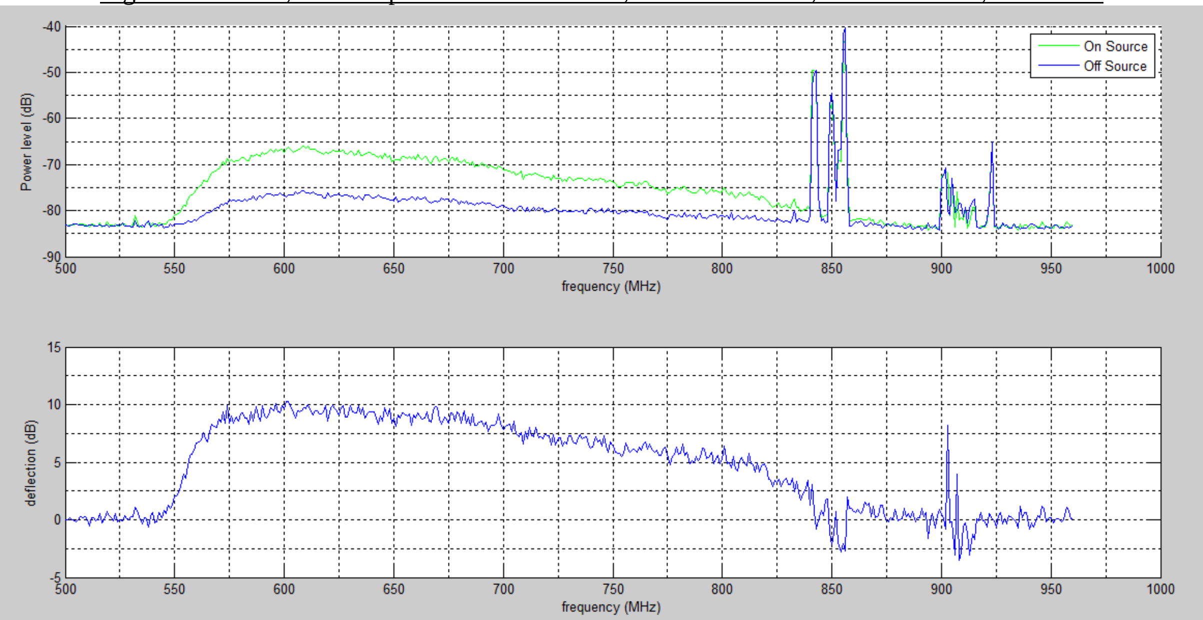
$$f(x) = (1.22 * (3E + 8)/(x * (1E + 6) * D)) * 180 * (60/3.1415)$$

It also shows the error (95 % confidence intervals) on the beam-widths according to the quality of the fit.

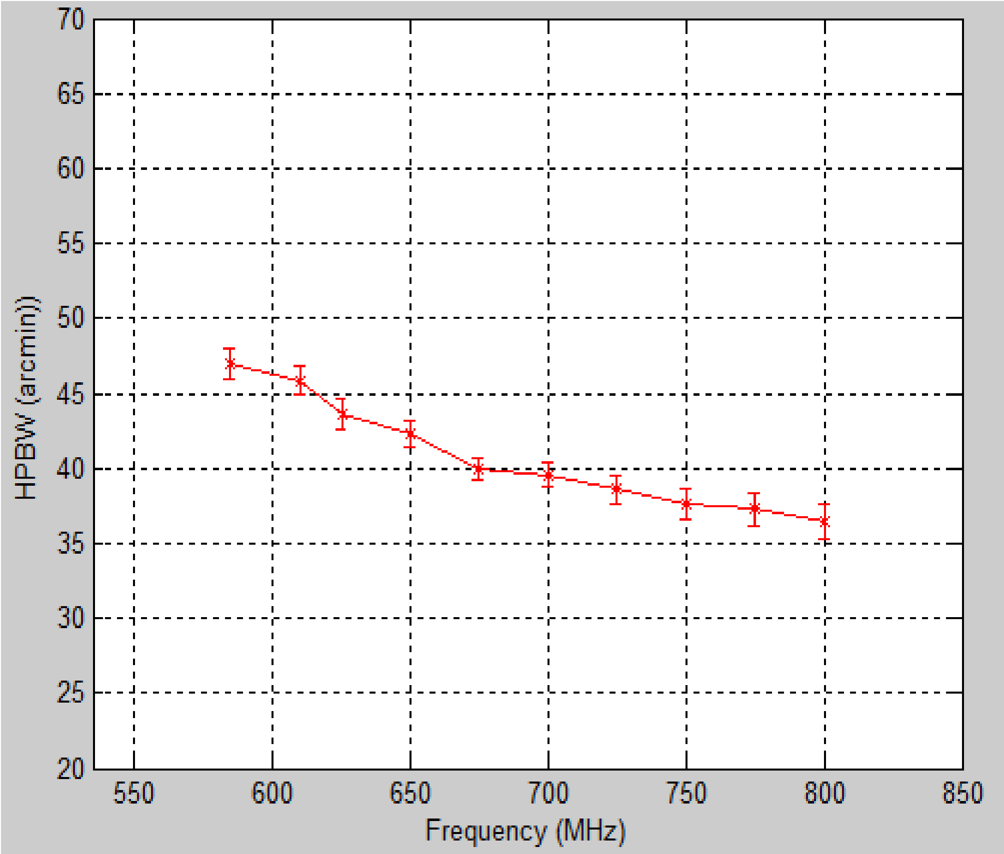
# 3. Results

The following sets of results are for the scan test conducted on 6th May, 2015, carried out on the GMRT's C10 Antenna's *Cone Dipole Feed 550-900 MHz with choke*. The supernova remnant Cassiopeia-A (Cas-A),[3] was used as the source. This Antenna was steered through its elevation angle. We can observe from the graphs that the desired Source deflection is achieved.

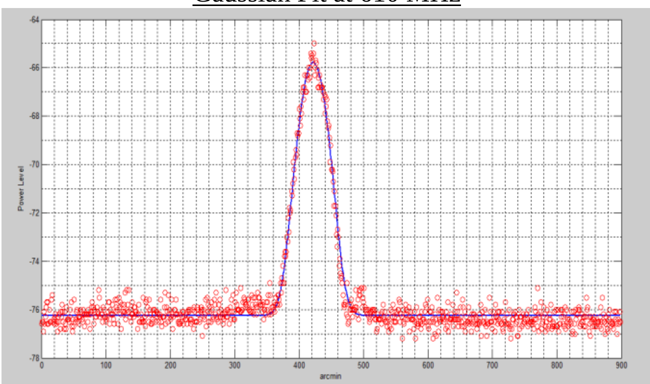
Fig. 3.1. Scantest; Cone Dipole Feed with Choke, focus: 610 MHz; Elevation scan; Channel 1




[3] Deflection Levels at Cas-A at 610 MHz are calculated to be 11.1dB




Gaussian Fit at 610 MHz



Gaussian Fit at 700 MHz

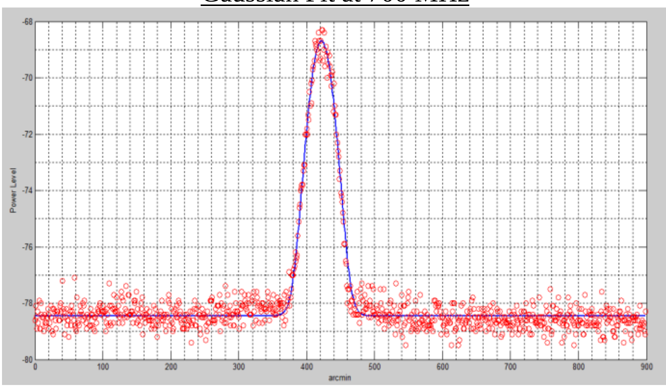
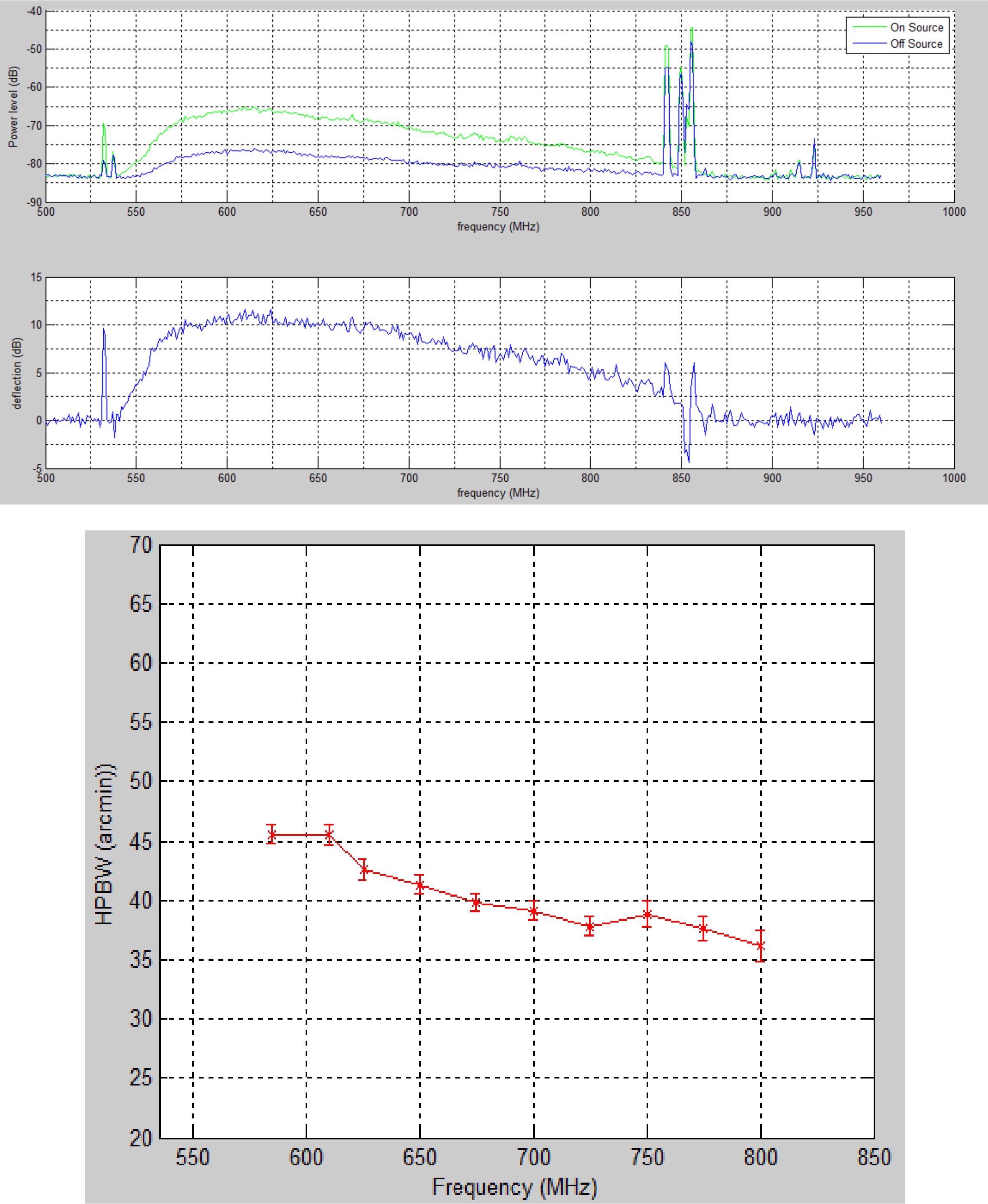
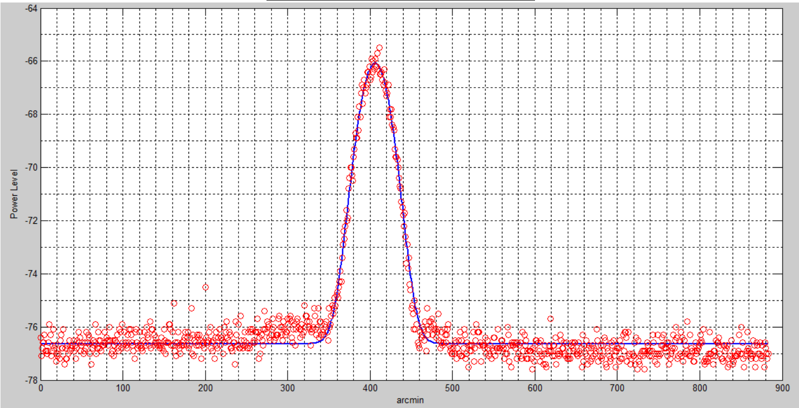


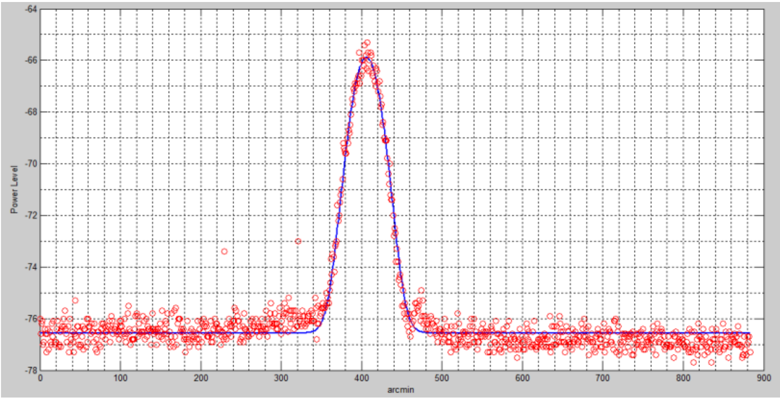
Fig. 3.2. Scantest; Cone Dipole Feed with Choke, focus: 610 MHz; Elevation scan; Channel 2



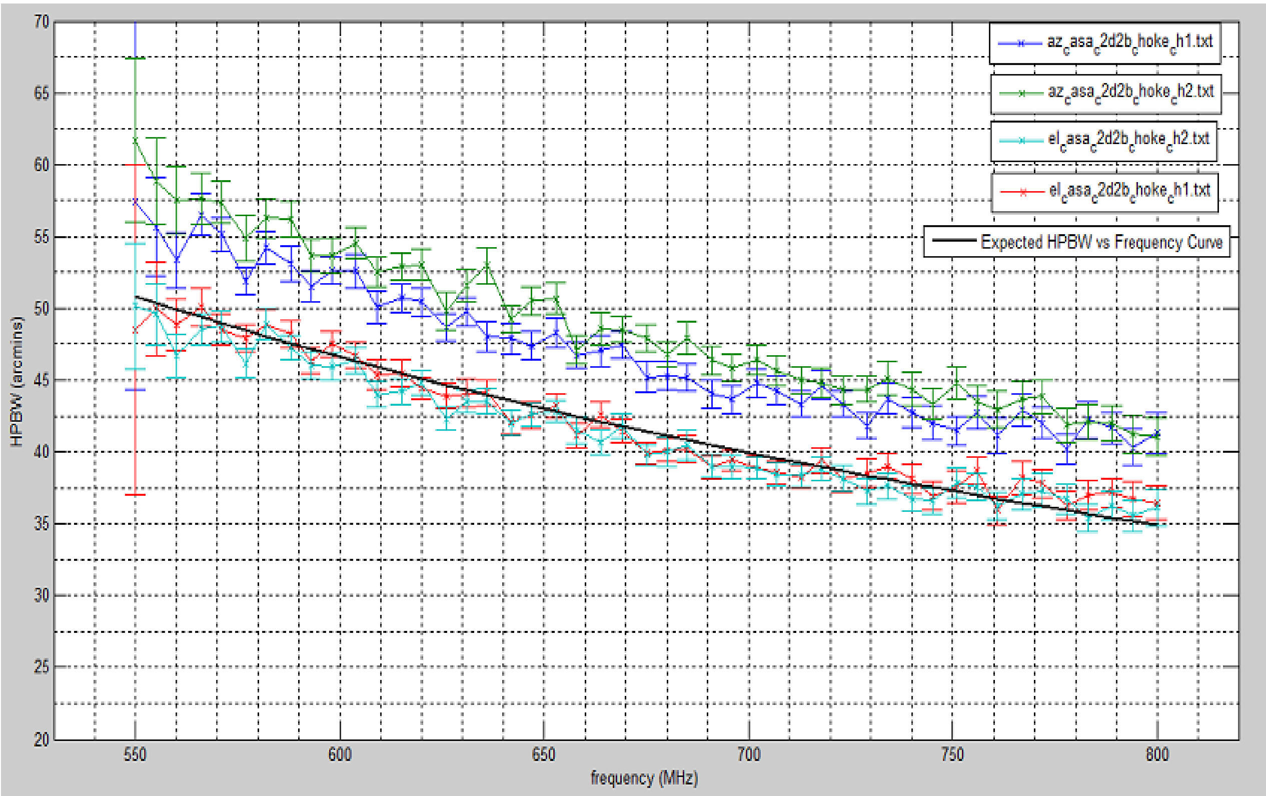
Gaussian Fit at 610 MHz




Gaussian Fit at 700 MHz

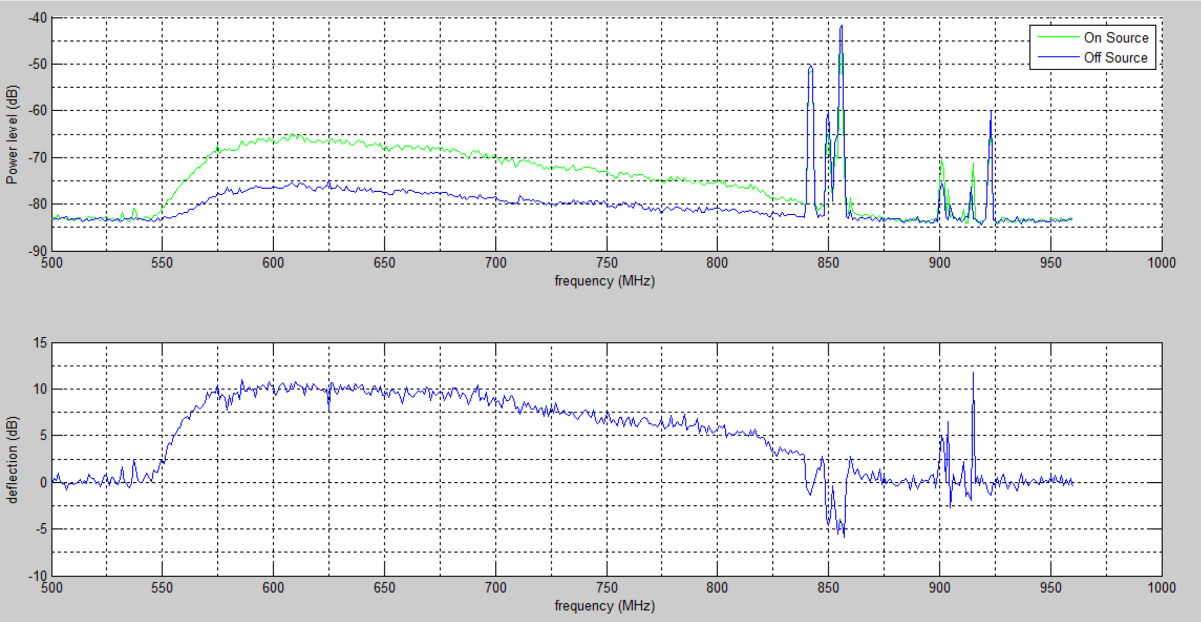


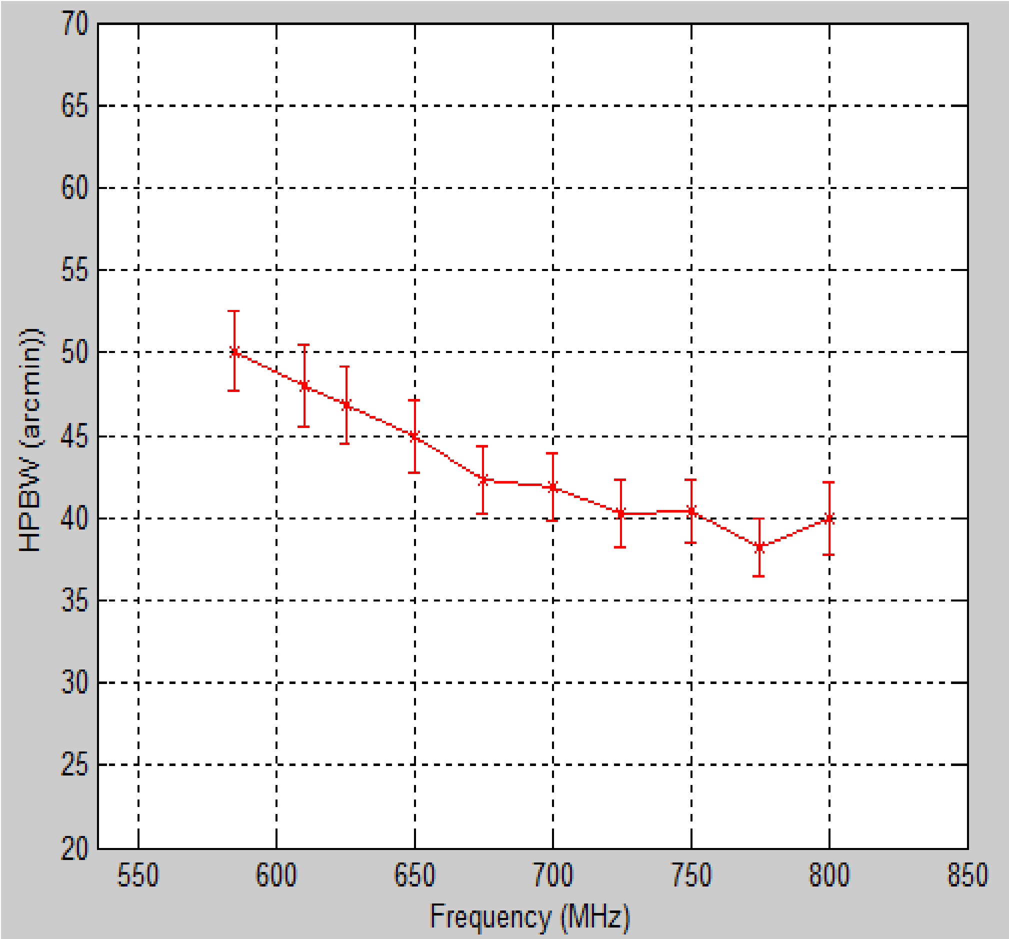

HPBW vs Frequency; 20/05/2015; with choke;Azimuth ch1,2; Elevation ch 1,2



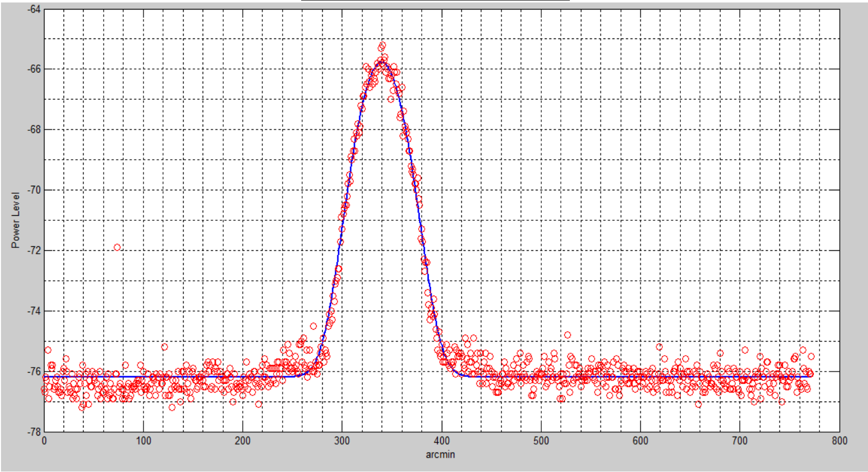
The following sets of results are for the scan test conducted on 6th May, 2015, carried out on the GMRT's C10 Antenna's *Cone Dipole Feed without choke for Cone 2 Dipole 2b* for the 550-900 MHz range. The supernova remnant Cassiopeia-A (Cas-A), was used as the source.

Fig. 3.3. Scantest; Cone Dipole Feed without Choke,Cone 2 Dipole 2b, focus: 610 MHz; Azimuth scan; Channel 1



<u>Gaussian Fit at 610 MHz</u>




<u>Gaussian Fit at 700 MHz</u>

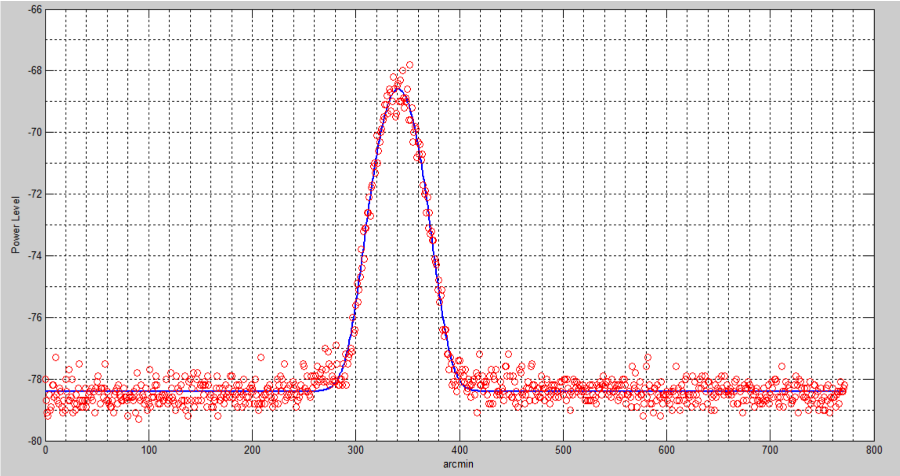
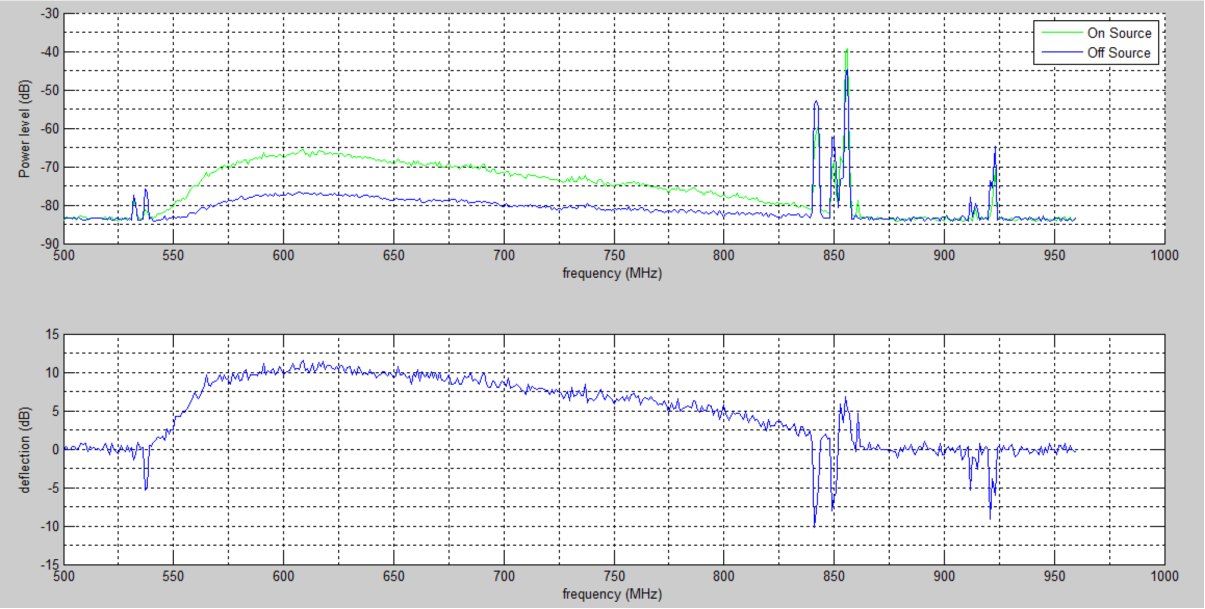
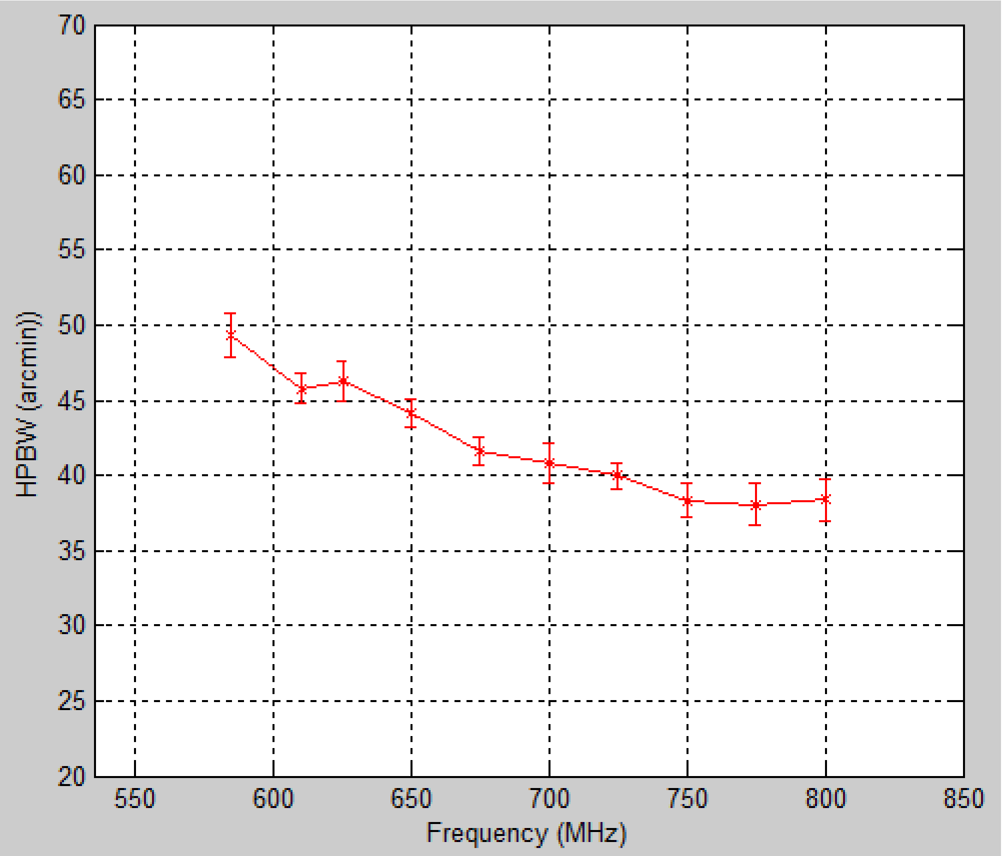
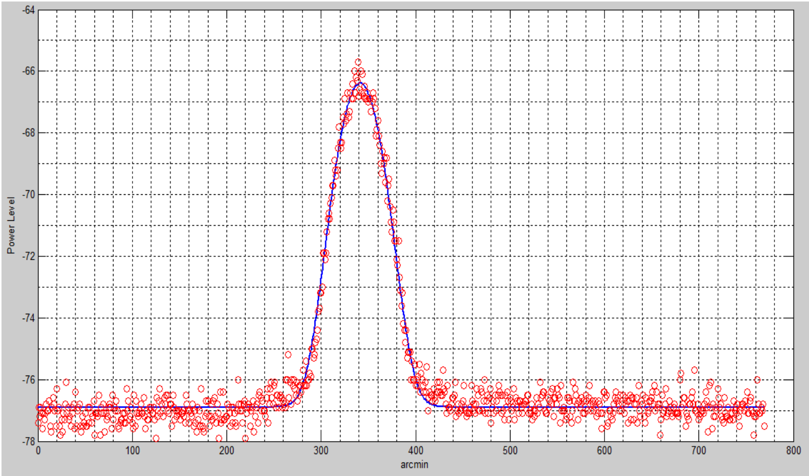


Fig. 3.4. Scantest; Cone Dipole Feed without Choke,Cone 2 Dipole 2b, focus: 610 MHz; Azimuth scan; Channel 2

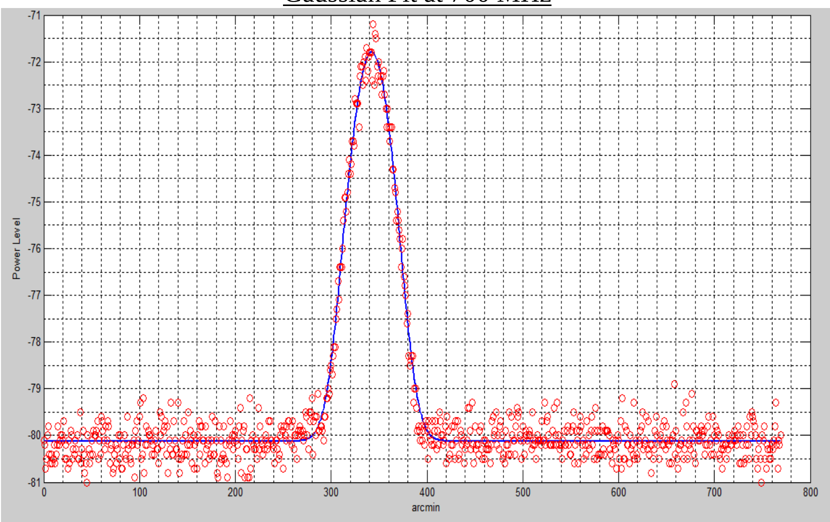


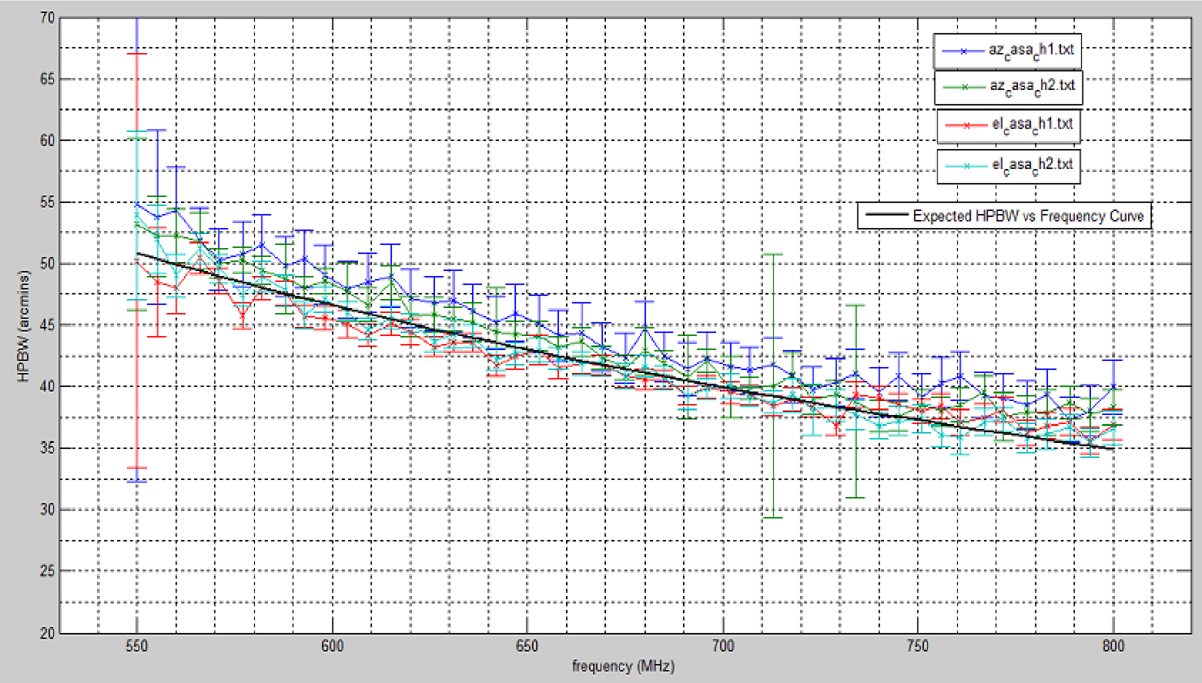
Gaussian Fit at 610 MHz




Gaussian Fit at 700 MHz



HPBW vs Frequency; 06/05/2015;Azimuth ch1,2; Elevation ch 1,2



# 4. Conclusion

Continuing work expects to involve the cosine Elevation angle correction factor to Azimuth scans. (The elevation of the source for the epoch of observation is to be calculated. The precession of the source co-ordinates for the epoch of observation for calculating elevation of the source also need to be formulated.)

Future scope in the complete automation of the Scan test includes adding various other characterization tests to the code to enhance functionality. The MATLAB code offers a far greater functionality to the data analysis. The MATLAB isn't restricted by this or the type of Spectrum Analyser used. The MATLAB code offers multiple analyses which are also important in the characterization such as deflection measurements. One of the major advantages of this package is that only one scan measurement is required to characterize the entire frequency band or the set of frequencies the user chooses to analyse.

Another advantage of MATLAB is its powerful and diverse inbuilt library functions and the online community and resources. Data Analyses and Automation are stand out features of MATLAB. The automated package can be run in tandem with the daq to reduce the time delay and get immediate results.

# Bibliography


*gmrt.ncra.tifr.res.in*

*http://www.gmrt.ncra.tifr.res.in/gmrt_hpage/sub_system/front/front.html*

*http://www.gmrt.ncra.tifr.res.in/gmrt_hpage/Upgrade/Beamwidth-fitting.pdf*

Chengalur J. N. , Y. Gupta and K. S. Dwarkanath, "Low Frequency Radio Astronomy" 3rd Edition NCRA-TIFR Pune, 2007.

Shankar G., "Ch. 19 Low Frequency Radio Astronomy," 3rd Edition NCRA-TIFR Pune, 2007.

Rao B. H., G. Shankar, A. P. Kumar, "Wideband feeds for the upgraded GMRT" IOP Conf. Series: Materials Science and Engineering 44, 2013.

Press W. H., S. A. Teukolsky, W. T. Vetterling, B. P. Flannery, "Numerical Recipes Function in C: The Art of Scientific Computing" 2nd Edition, CAMBRIDGE UNIVERSITY PRESS.

# I. Appendix

---

//read_input_file.m

------

```
function importfile(fileToRead1)
%  IMPORTFILE(FILETOREAD1)
%  Imports data from the specified file
%  FILETOREAD1:  file to read
%  Import the file
newData1 = importdata(fileToRead1);

% Create new variables in the base workspace from those fields.
%Segregating numeric and non-numeric data

vars = fieldnames(newData1);
for i = 1:length(vars)
    assignin('base', vars{i}, newData1.(vars{i}));
end
```

---

// Scan Test Function with deflection measurements and user chosen frequency analysis

scantest.m

-------

```
clear all
prompt = 'Number of files : ';
f_num = input(prompt);

prompt = 'Start Frequncy (MHz): ';
fstart = input(prompt);
prompt = 'End Frequncy (MHz): ';
fend = input(prompt);
prompt = 'Frequency of the Feed focussed (MHz): ';
fin = input(prompt);

for i=1:f_num

    prompt = 'Enter File name : ';
    readfile = input(prompt,'s');
    test_05_05(readfile)

    d = size(data);
    xdata = 0:(d(1)-1);
```

```
    diff = (fend - fstart)/(d(2)-1);

    %%%
    col = floor(((fin - fstart)/ diff) + 1);

    %%

    %%ON SOURCE POWER LEVEL TO BE SORTED
    x = [fstart:(fstart + d(2) - 1)];


    % The row corresponding to the On source power level needs to be
calculated.
    % Since we know it is around the mid way, we search from row 350 to 410
    % 'col' is the column of the Focussed frequency and thus we find in its
    % vicinity the max power level(max(maxval1)), thereby fetching the
corresponding row
    % (row_on)

    maxval = [];
    maxval1 = [,];
    max_in_row_array = [];
    %%ON SOURCE POWER LEVEL TO BE SORTED
     subplot(2,1,1)

    x = [fstart:(fstart + d(2) - 1)];
    for i = 350:410
        for j = (col - 10):(col + 10)
            maxval = [maxval; data(i,j)];
            %%create array, find max val of the array;
        end
        max_in_row = max(maxval);
        max_in_row_array = [;max_in_row];
        maxval1 = [i, max_in_row_array];
      end
    A = max(maxval1,[],1);
    for i = 350:410
        for j = (col - 10):(col + 10)
            if(A(2) == data(i,j))
                row_on = i;
                break
            end
        end
    end
    hold on
    plot(x, data(row_on,:),'g', x, data(1,:),'b')
    legend('On Source','Off Source','Location','NorthEast');
    xlabel( 'frequency (MHz)');
    ylabel( 'Power level (dB)' );
    grid on
    grid minor

    def = (data(row_on,:) - data(1,:));
    subplot(2,1,2)

    plot (x, def)
    hold on
    xlabel( 'frequency (MHz)');
    ylabel( 'deflection (dB)' );
```

```
    grid on
    grid minor

    %%%
    prompt = 'How  many Frequencies to be analysed? ';

        num = input(prompt);
        for i = 1:(num)
            prompt = 'Input Frequncy (MHz): ';
            fin = input(prompt);
            col = ceil(((fin - fstart)/ diff) + 1);
            ymin_dB = min(data(:,col));
            ymin = 10.^(ymin_dB/10);
            xdata = xdata(:);
            A = [xdata.^2,  xdata,  ones(size(xdata))];
            ydB = data(:,col);
            y = 10.^(ydB./10);
            Y = y - ymin;
            nonlinfit(xdata,Y,ydB,ymin, num,i,fin, d);
         end

    grid on
    grid minor
end
```

---

//Non Linear Fit modelling Function for scantest.m

------

```
function nonlinfit(xdata,Y,ydB,ymin,num,i,fin, d)

% Set up fittype and options.
ft = fittype( 'gauss1' );
opts = fitoptions( ft );
opts.Display = 'Off';
% opts.Lower = [1e-08 (d(1)/2-50) 0];
opts.StartPoint = [max(Y) 390 50];
% opts.Upper = [4e-07 (d(1)/2+50) 125];
yp = opts.StartPoint(1);
mu = opts.StartPoint(2);
sigma = opts.StartPoint(3)/sqrt(2);
ymin_ = ymin;

betaguess = [yp mu sigma ymin_];
warning('off','all');

[betahat, R, J, COVB, MSE] = nlinfit(xdata, Y+ymin, @GaussEqn, betaguess);

yfit = GaussEqn(betahat, xdata);
yfin = 10*log10(yfit);
HPBW = 2.36*betahat(3);
ci = nlparci(betahat, R,'covar', COVB);
CI_ = 2.36*ci(3,1)- 2.36*ci(3,2);
```

```
figure
h = plot(xdata, yfin, 'b-', xdata, ydB, 'ro');
set(h(1),'linewidth',2);
ylabel( 'Power Level' );
xlabel( 'arcmin' );
grid on
grid minor
% figure
% plot(ydB, yfin, '*')
% grid on
% grid minor

persistent fin_array;

fin_array = [fin_array;fin];

persistent HPBW_ARRAY;

persistent CI;

CI = [CI;CI_];
% After calculating HPBW for each of the input frequencies, it is appended
% toan array for plotting and displaying collectively.

HPBW_ARRAY = [HPBW_ARRAY;HPBW];
if(i == num)
    figure
    freq_HPBW = [fin_array.';HPBW_ARRAY.'].';
    fprintf('The Frequency vs HPBW (arcmin) table :\n')
    disp(freq_HPBW)
    HPBWdeg = HPBW_ARRAY*0.0167;
    freq_HPBWdeg = [fin_array.';HPBWdeg.'].';
    fprintf('The Frequency vs HPBW (deg) table :\n')
    disp(freq_HPBWdeg)
%     plot(fin_array, HPBW_ARRAY, 'ro-', 'Linewidth', 2)

    errorbar(fin_array, HPBW_ARRAY, CI, 'rx-')
    grid on
    grid minor
    axis([(fin_array(1)-50) (fin_array(num)+50) 20 70]);
    xlabel( 'Frequency (MHz)' );
    ylabel( 'HPBW (arcmin))' );
    clear all

end




end
```

---

//Gaussian Equation Function for scan test

GaussEqn.m

```
function y = GaussEqn(par, xdata)

yp = par(1);
mu = par(2);
sigma = par(3);
ymin_ = par(4);
y = (yp)*exp( -(xdata - mu).^2 / (2*sigma^2)) + ymin_;

end
```

---

// Beamwidth v/s Frequency over the input frequency range

bw_test.m

------

```
clear all
%%%
prompt = 'Number of files : ';
f_num = input(prompt);
prompt = 'Start Frequncy (MHz): ';
fstart = input(prompt);
prompt = 'End Frequncy (MHz): ';
fend = input(prompt);


    prompt = 'Start Frequncy for analysis(MHz): ';
    f_in = input(prompt);
    prompt = 'End Frequncy for analysis(MHz): ';
    f_fin = input(prompt);

for f = 1:f_num

%%%
    prompt = 'Enter File name : ';
    readfile = input(prompt,'s');
    test_05_05(readfile)

    d = size(data);
    xdata = 0:(d(1)-1);

    diff = (fend - fstart)/(d(2)-1);
    col_in = floor(((f_in - fstart)/ diff) + 1);
    col_fin = floor(((f_fin - fstart)/ diff) + 1);
    k = (col_in - col_fin)/5;
    %col = floor(((fin - fstart)/ diff) + 1);
    frange = linspace(550,800,47);;
    frange = (floor(frange))';


%     fstart = 60;
```

```
%       k = 250;
          for i = col_in:5:col_fin
              ymin_dB = min(data(:,i));
              ymin = 10.^(ymin_dB/10);
              xdata = xdata(:);
              A = [xdata.^2,  xdata,  ones(size(xdata))];
              ydB = data(:,i);
              y = 10.^(ydB./10);
              Y = y - ymin;
              gaussfit_HPBW(xdata,Y, ymin, col_fin, i, d, readfile, frange);
          end
end
```

---

```
// Gaussian Fit code for HPBW v/s frequency
gaussfit_HPBW.m

------

function gaussfit_HPBW(xdata,Y, ymin,col_fin,i, d, readfile, frange)
% Set up fittype and options.
ft = fittype( 'gauss1' );
opts = fitoptions( ft );
opts.Display = 'Off';
opts.Lower = [1e-08 (d(1)/2-50) 0];
opts.StartPoint = [max(Y) (d(1)/2) 45];
opts.Upper = [4e-07 (d(1)/2+50) 125];

yp = opts.StartPoint(1);
mu = opts.StartPoint(2);
sigma = opts.StartPoint(3)/sqrt(2);
ymin_ = ymin;

%%
warning('off','all');
%%

betaguess = [yp mu sigma ymin_];

[betahat, R, J, COVB, MSE] = nlinfit(xdata, Y+ymin, @GaussEqn, betaguess);

yfit = GaussEqn(betahat, xdata);
yfin = 10*log10(yfit);
HPBW = 2.36*betahat(3);
ci = nlparci(betahat, R,'covar', COVB);
CI_ = 2.36*ci(3,1)- 2.36*ci(3,2);
grid on
grid minor


persistent HPBW_ARRAY;

persistent CI;
```

```
CI = [CI;CI_];

HPBW_ARRAY = [HPBW_ARRAY;HPBW];

if (i == col_fin)


        [xData, yData] = prepareCurveData( frange, HPBW_ARRAY );

        % Set up fittype and options.
        ft = fittype( '(1.22*(3e8)/(x*45))*k', 'independent', 'x',
'dependent', 'y' );
        opts = fitoptions( ft );
        opts.Display = 'Off';
        opts.Lower = -Inf;
        opts.StartPoint = 0.27692298496089;
        opts.Upper = Inf;

        % Fit model to data.
        [fitresult, gof] = fit( xData, yData, ft, opts );

        y = (1.22 *(3e8)./(frange .* (1e6) * 45))* 180 *(60/3.1415);

        % Label axes
        xlabel( 'frequency (MHz)' );
        ylabel( 'HPBW (arcmins)' );

        axis([(frange(1)-20) (frange(numel(frange))+20) 20 70]);
        h_ = errorbar(frange, HPBW_ARRAY, CI, 'x-');
        leg1=legend(h_(1),readfile,'location','NorthEast');
        grid on
        grid minor
        clear HPBW_ARRAY;
        clear CI;
        clear axis;
        hold all;
        copied_leg1=copyobj(leg1,gcf);
        h = plot(frange, y,'k-');
        leg2=legend(h,'Expected HPBW vs Frequency
Curve','location','North');
        set(h(1),'linewidth',1.5);
%
end
```

---

## GaussEqn.m

```
function y = GaussEqn(par, xdata)

yp = par(1);
mu = par(2);
sigma = par(3);
ymin_ = par(4);
y = (yp)*exp( -(xdata - mu).^2 / (2*sigma^2)) + ymin_;

end
```