Krittika Summer Projects 2025

Imaging a Bright Neutron Star

and a Magnificent Radio Galaxy

Kenil Ajudiya[1,2] Ayushi Chhipa[1,3]

[1]Indian Institute of Science, Bangalore
[2]Indian Institute of Astrophysics, Bangalore

Mentees:

Shivendra: 24B1817, B.Tech Engineering Physics, 2nd Year
Karma Thendup Bhutia: 24B1852, B.Tech Engineering Physics, 2nd Year
Durva D. Kumar: 24B2161, B.Tech Mechanical Engineering, 2nd Year
Ruhaan Singh: 24B2278, B.Tech Mechanical Engineering, 2nd Year

Radio Astronomy

Contents

1 Introduction to Radio Astronomy

1.1 Units

The Jansky (Jy) is the most commonly used unit in radio astronomy. It is a measure of flux density and is defined as:

- $1\,\mathrm{Jy} = 10^{-26}\,\mathrm{W\ m^{-2}Hz^{-1}}$
- $1\,\mathrm{Jy} = 10^{-23}\,\mathrm{erg\ s^{-1}cm^{-2}Hz^{-1}}$

It is the fundamental unit used to describe how much power is received from a source per unit area per unit frequency.

There are three related but distinct concepts:

- **Specific intensity** (I_ν) is the power received per unit area, per unit solid angle, per unit frequency. It is often expressed in Jy/beam or Jy/sr.
- **Flux density** (S_ν) is the integral of specific intensity over a solid angle:

$$S_\nu = \int I_\nu \, d\Omega \quad (\text{in Jy})$$

- **Flux** (S) is the integral of flux density over frequency:

$$S = \int S_\nu \, d\nu \quad (\text{in W/m}^2)$$

1.2 Emission Mechanisms

Radio emission arises from various physical processes, broadly classified into thermal and non-thermal mechanisms.

Bremsstrahlung (Free-Free Emission)

Bremsstrahlung occurs when electrons are accelerated in the Coulomb field of ions, producing continuous radiation. It is a thermal process and is common in ionized gas regions.

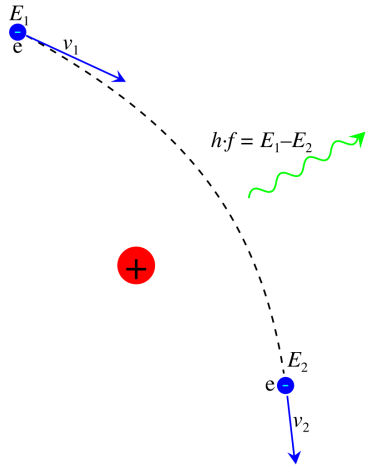


Figure 1

Thermal Dust Emission

Interstellar dust grains emit thermal radiation that can extend into the radio domain. The spectral shape is described by the Planck function:

$$B(\nu, T) = \frac{2h\nu^3}{c^2} \frac{1}{e^{h\nu/kT} - 1}$$

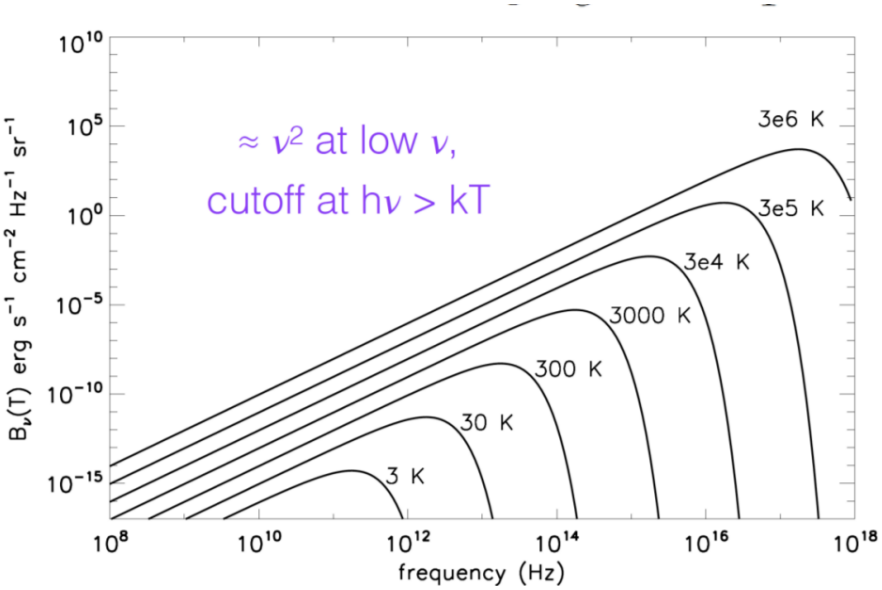


Figure 2: It follows Black body radiation

This emission originates from cold dust in molecular clouds and star-forming regions.

Synchrotron Emission

Synchrotron radiation is a non-thermal process caused by relativistic electrons spiraling around magnetic field lines. It produces broadband, polarized emission and is commonly associated with sources like active galactic nuclei (AGN), pulsars, and supernova remnants. The emissivity follows a power-law:

$$S_\nu \propto \nu^{-\alpha}$$

where α is the spectral index. This emission can exhibit relativistic beaming effects and is highly polarized.

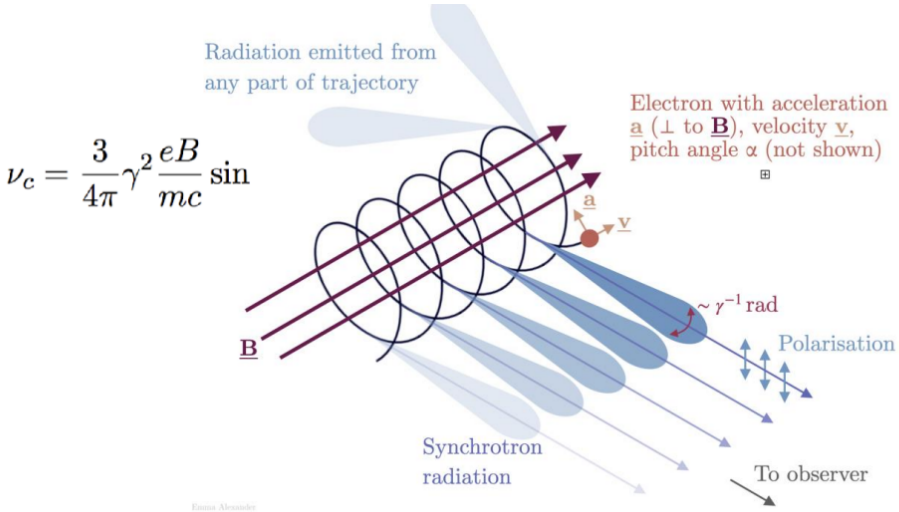


Figure 3

Spectral Line Emission from Molecular Gas

Molecules such as CO, HCN, and NH_3 emit spectral lines due to rotational transitions, often observed in the radio regime. These lines trace cold, dense regions of molecular gas. Molecules can form both in the gas phase and on dust grain surfaces.

Ionized Gas and Radio Recombination Lines

Ionized regions emit radio recombination lines analogous to the Balmer and Lyman series, but for very high principal quantum numbers ($n20$). As electrons cascade down energy levels, they emit radiation at distinct frequencies. Multiple transitions (e.g., H64α + H63α) can be combined to improve the signal-to-noise ratio.

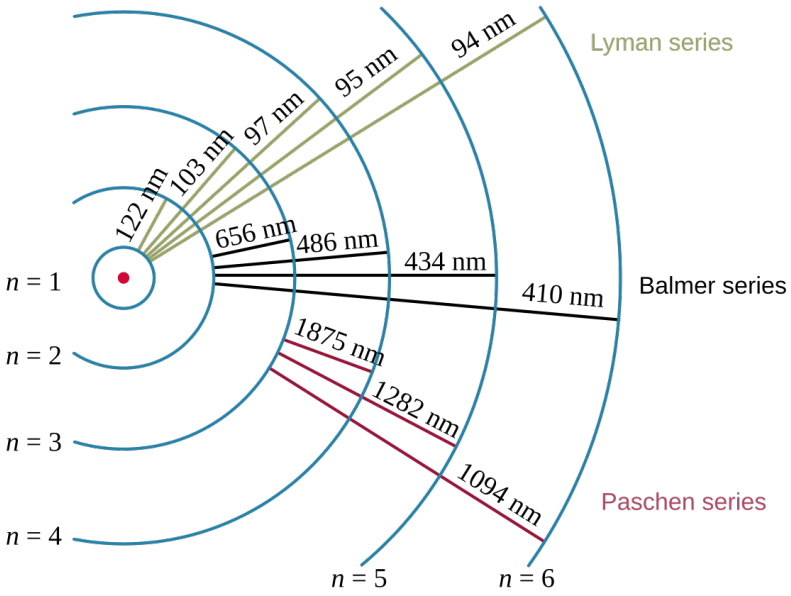


Figure 4: Spectral Lines Series

Neutral Hydrogen Emission

The 21 cm line arises from the hyperfine spin-flip transition of neutral hydrogen atoms. It has a rest frequency of 1420 MHz and is widely used in galactic and extragalactic studies. This line is detectable in both emission and absorption, making it a powerful tracer of neutral gas.

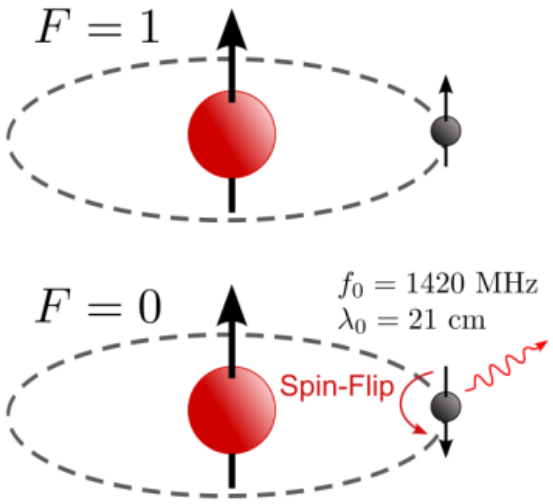


Figure 5: Anti parallel magnetic moment decreases potential energy which is emitted as a photon

Flux Density vs. Frequency

A plot of flux density (S_ν) versus frequency reveals the dominant emission mechanisms across the radio spectrum. At low frequencies (tens to hundreds of MHz), **synchrotron emission** dominates, typically showing a power-law decline with increasing frequency due to the non-thermal nature of the radiation.

Around the GHz range, contributions from **thermal Bremsstrahlung** become significant, producing a relatively flat or slowly declining spectrum.

At higher frequencies (tens to hundreds of GHz), **thermal dust emission** begins to rise steeply following the shape of the Planck function.

Additionally, narrow **spectral lines** from molecular gas and atomic transitions appear as spikes superimposed on the continuum, corresponding to specific rotational or spin-flip transitions. This multi-component nature of the radio SED helps astronomers disentangle physical processes occurring in various astrophysical environments.

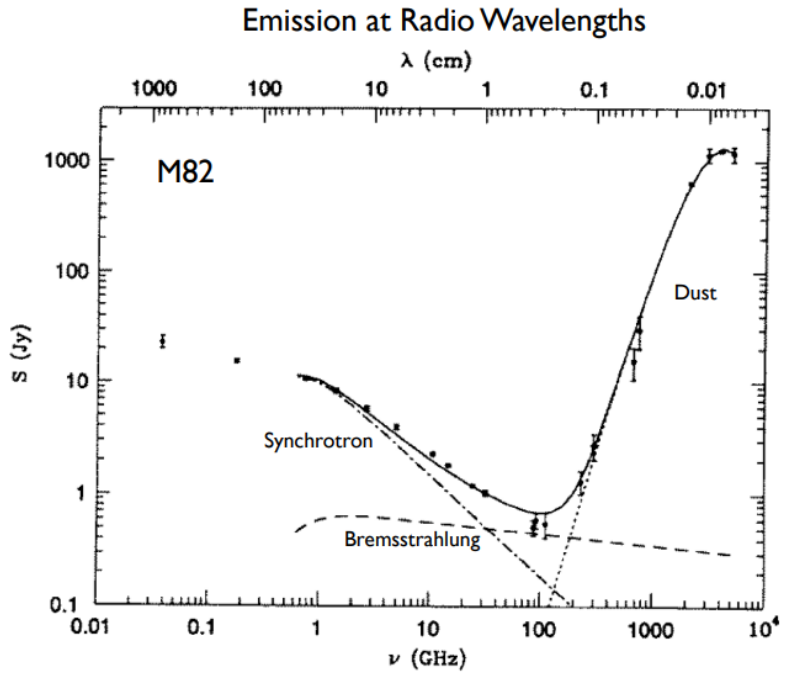


Figure 6: Flux vs Frequency

1.3 Types of Radio Telescopes

Single Dish Telescopes

Single-dish telescopes consist of a single large antenna that collects incoming radio waves. Examples include the Green Bank Telescope (GBT), ALMA in total power mode, Parkes, and APEX. The resolution of such telescopes is limited by the dish size, and the beam size θ can be approximated by:

$$\theta \approx 1.2\frac{\lambda}{D}$$

where λ is the observing wavelength and D is the dish diameter.

Interferometers

Interferometers use multiple antennas working together to simulate a much larger telescope. This configuration significantly improves angular resolution. Prominent examples include the Very Large Array (VLA), ALMA, NOEMA, ATCA, and the VLBA.

In interferometry, the **primary beam** is the field of view determined by the size of each individual dish, while the **synthesized beam** represents the effective resolution and depends on the maximum separation between antennas (i.e., the baseline). Because the baseline projection varies with source position and time, it is the *projected baseline* that contributes to the effective resolution and image quality.

In an interferometric array, signals from different antennas must be combined in a way that preserves phase information. This is done by the **correlator**, a specialized digital system that multiplies and averages the signals from all antenna pairs. The correlator outputs quantities known as *visibilities*, which encode information about the spatial structure of the radio source. Accurate time synchronization and signal digitization are essential for effective correlation.

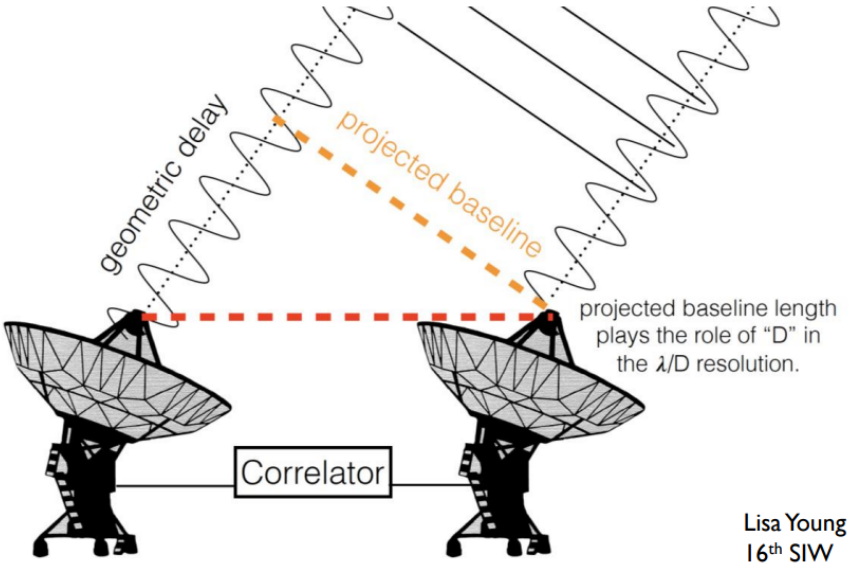


Figure 7

Types of Antennas

Radio telescopes utilize different antenna designs based on their intended frequency range and observational goals.

The most widely used is the **parabolic dish antenna**, which consists of a curved metallic reflector shaped like a paraboloid. Incoming parallel radio waves are focused onto a receiver placed at the focal point. These antennas offer high directionality and are used in most single-dish and interferometric arrays.

Figure 8

Dipole antennas are among the simplest, typically made of two conductive elements aligned along a straight axis. They are resonant structures, commonly used in low-frequency observations and deployed in phased arrays.

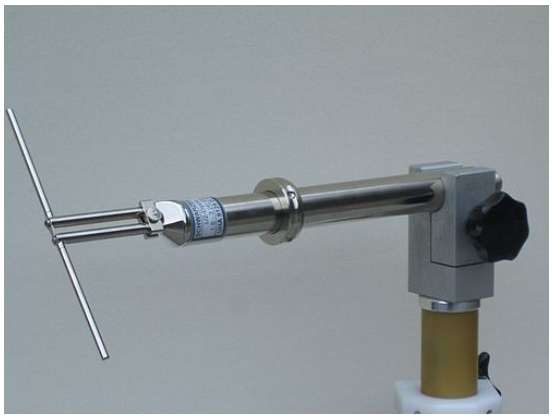


Figure 9

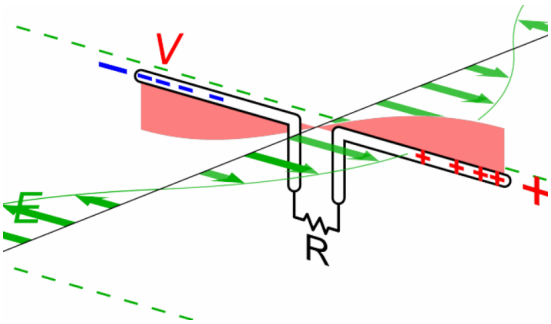


Figure 10: How recieved signal produces oscillating potential

Wire reflector antennas, such as the cylindrical or Yagi-Uda designs, use stretched wires or mesh reflectors to guide and focus radio waves. These are lightweight, cost-effective, and suitable for large-area coverage at long wavelengths.

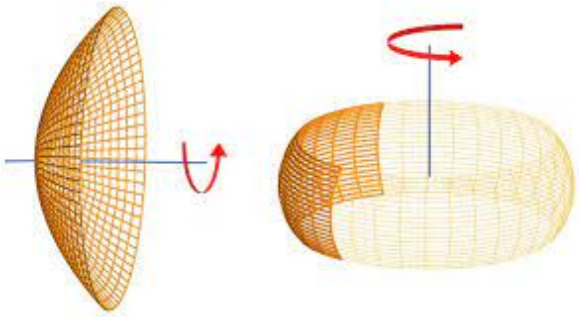


Figure 11

2 Antennas & Receivers

2.1 Electronic Diagram and Effect of Antenna Properties on Data

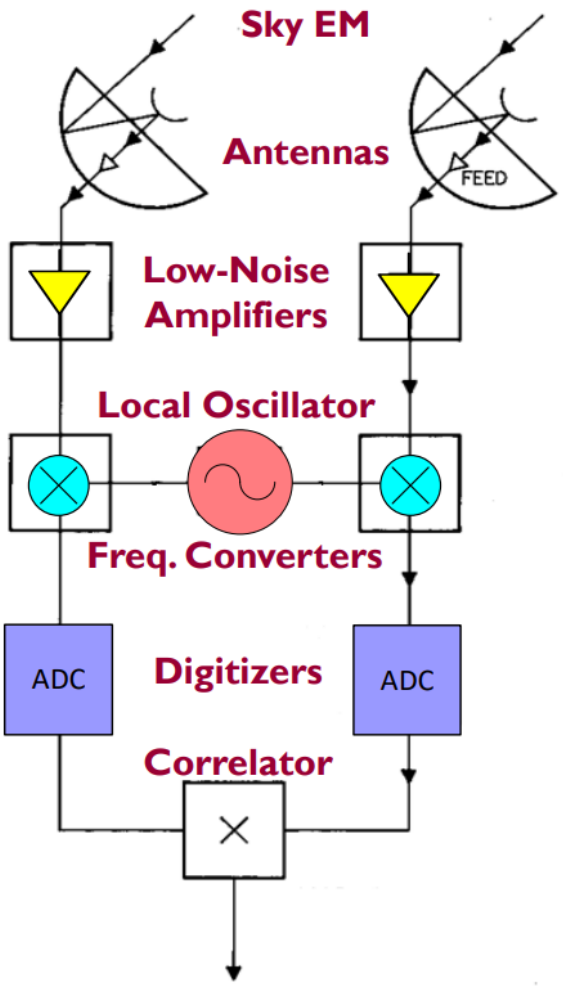


Figure 12: Block Diagram

- The antenna's **amplitude pattern** causes signal strength to vary across the source.
- The **phase pattern** introduces spatial variations in the observed phase.
- **Polarization properties** of the antenna can distort the source's intrinsic polarization.
- **Pointing errors** lead to time-dependent amplitude and phase fluctuations.
- **Noise pickup from the ground** can introduce variable amplitude errors over time.

- **Antenna surface deformations**, especially at short wavelengths, can cause significant amplitude and phase errors.

2.2 Terminology and Definitions

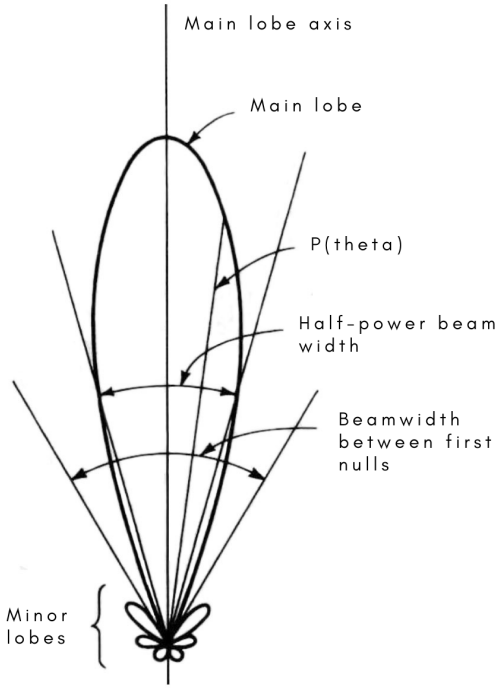


Figure 13: Beam pattern for antenna

- The **effective collecting area**, denoted $A(n, q, f)$ in m^2, represents the antenna's ability to gather energy as a function of direction and frequency.
- The **on-axis response** is defined as $A_0 = hA$, where h is the aperture efficiency and A is the geometric area.
- The **normalized antenna pattern** (also called the primary beam) is:

$$\frac{A(n, q, f)}{A_0}$$

- The **beam solid angle** (or field of view), Ω_A, is given by:

$$\Omega_A = \iint A(n, q, f)\, d\Omega$$

- A fundamental relation between effective area and beam solid angle is:

$$A_0 \Omega_A = \lambda^2$$

  This implies a trade-off: an antenna can have either high gain (large effective area) or a wide field of view (large solid angle), but not both simultaneously.

- The **main lobe** is the central, most sensitive part of the antenna beam that captures the target source.

- **Sidelobes** and **backlobes** are off-axis regions of sensitivity. They:
  - Increase the system temperature by picking up ground noise.
  - Make the antenna more susceptible to radio frequency interference (RFI).
  - Reduce image dynamic range by detecting strong background sources outside the main beam.

2.3 Beam Patterns

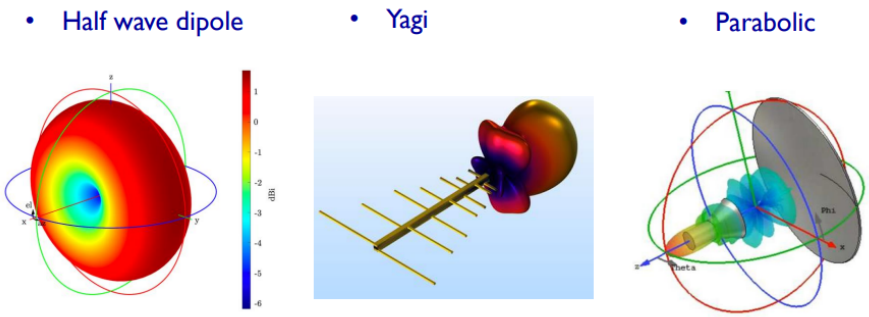


Figure 14: Beam patterns for different types of antennas

- The **beam pattern** of an antenna describes how its sensitivity varies with direction in the sky.

- It is determined by the geometry and illumination of the antenna aperture — including shape, feed design, and surface accuracy.

- For interferometers, the synthesized beam pattern arises from the Fourier transform of the antenna sampling in the (u, v) plane.

- The beam pattern is essentially a rotated version of the **fringe pattern**, which represents the interference pattern produced by a baseline as a function of sky position. I mean if you draw fringe pattern on a x axis , now u fold the x axis from central finge you get this, which creates lobes.

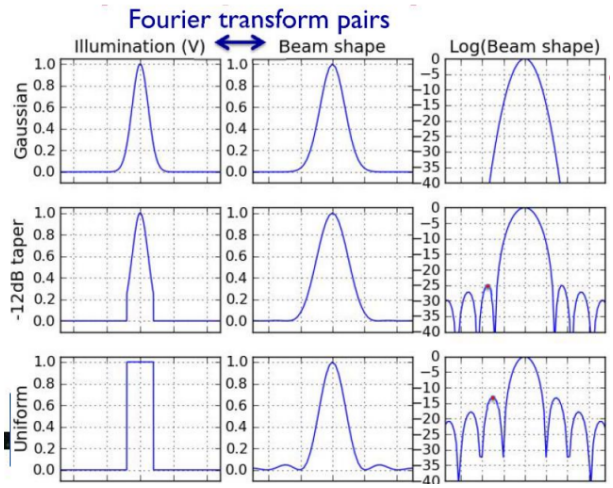


Figure 15: Antenna's far-field radiation pattern (beam) is related to the Fourier transform of its aperture distribution (illumination pattern)

2.4 Antenna Mounts

The mounting of a radio antenna determines how it is physically oriented and moved to track sources across the sky. It affects not only the mechanical stability of the structure but also the complexity of tracking celestial objects and the behavior of the beam on the sky.

Altitude-Azimuth (Alt-Az) Mount

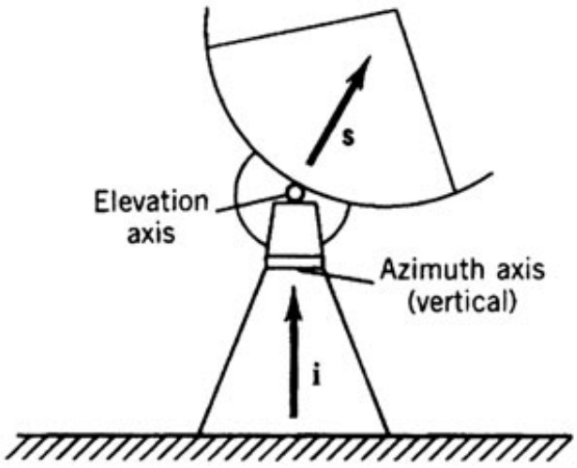


Figure 16

The altitude-azimuth mount consists of two rotational axes: one horizontal (azimuth) and one vertical (altitude). The antenna rotates around the vertical axis to sweep in azimuth

and tilts up or down to change altitude. This design is structurally simple, mechanically robust, and widely used in modern large radio telescopes such as the Green Bank Telescope.

However, it has two main disadvantages. First, there is a **zone of avoidance** directly overhead (at the zenith), where both axes must rotate rapidly, making tracking unstable. Second, as the telescope tracks a source, the beam rotates relative to the sky, requiring software correction to maintain proper orientation — an effect known as **field rotation**.

Equatorial Mount

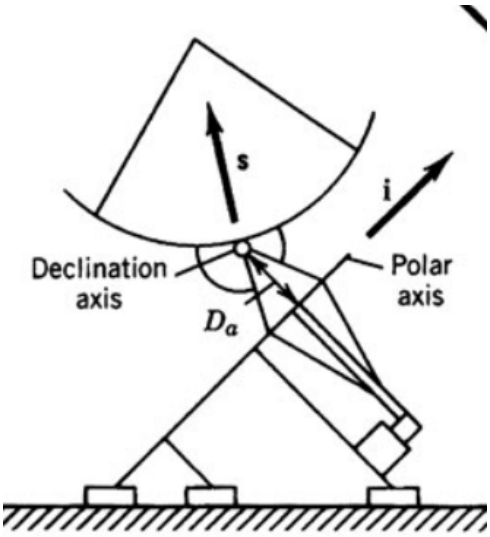


Figure 17

An equatorial mount has one axis aligned with Earth's rotation axis (right ascension) and the other perpendicular to it (declination). This allows the antenna to track stars by rotating at a constant rate around just one axis, simplifying tracking. Structurally, equatorial mounts are more complex and harder to balance, especially for large dishes.

A major limitation is near the celestial poles — for observers at mid-latitudes, sources near the horizon at the pole are difficult to access due to extreme tilts required in the structure. As a result, equatorial mounts are less favored for large-scale arrays but still useful in smaller radio telescopes and optical systems.

2.5 Antenna Optical Configuration

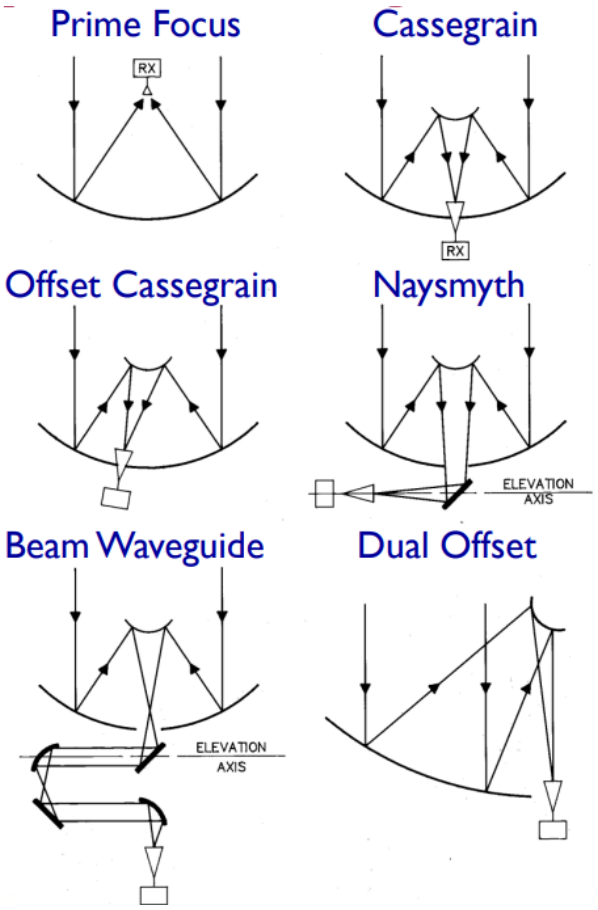


Figure 18

Prime Focus

- Can be used across the entire frequency range of the reflector.
- Subject to over-illumination, where radiation spills over the edge of the dish and picks up ground noise, increasing system temperature.
- Limited number of receivers can be placed at the focus; access for maintenance or switching is difficult.

Multiple Reflector (Cassegrain) Systems

- Allows more space and easier access for multiple receivers.
- Reduces ground spillover — any excess illumination typically lands on cold sky, lowering system noise.

- May limit low-frequency performance due to the large size of required feed horns.
- Over-illumination by large feed horns can reduce gain by exciting sidelobes in the primary reflector.
- Nearby strong sources (even a few degrees off-axis) can enter through sidelobes, limiting image dynamic range.

Offset Optics (Gregorian or Cassegrain)

- Unblocked aperture improves aperture efficiency and reduces sidelobes and scattered radiation.
- Results in lower system temperature and higher imaging fidelity.
- Better suited for practical low-frequency feed designs.
- More complex and expensive mechanical structure due to asymmetry and lack of rotational symmetry.
- Requires custom tooling for each unique panel, increasing construction cost.

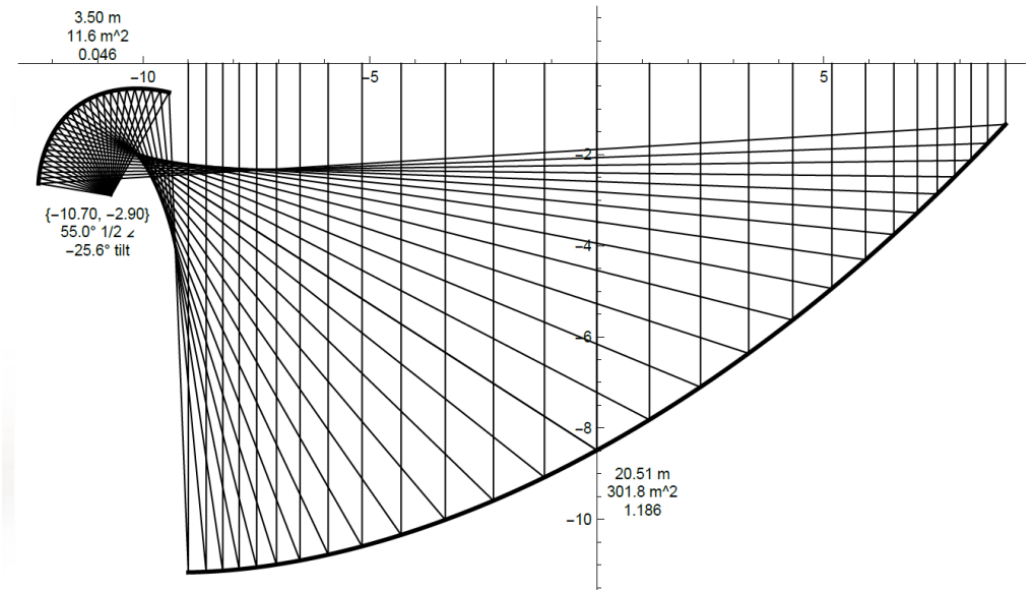


Figure 19: Ray Diagram for Offset Optics

Aperture Efficiency

Aperture efficiency (η) quantifies how effectively the physical area of an antenna contributes to signal collection. It is defined as the ratio of the effective collecting area A_e to the

physical (geometric) area A of the antenna:

$$A_e = \eta A$$

The value of η typically ranges from 0.5 to 0.8 and depends on several factors: illumination efficiency (how uniformly the aperture is filled), spillover losses (energy spilling outside the dish), surface accuracy (especially critical at high frequencies), blockage from support structures, and feed alignment. A high aperture efficiency leads to better sensitivity and improved signal-to-noise ratio in observations.

2.6 Surface Errors

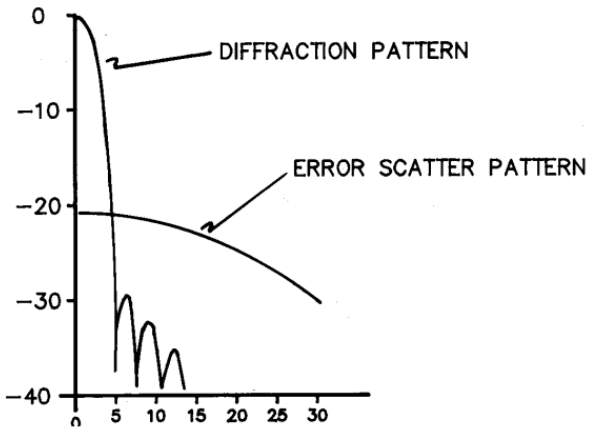


Figure 20

- **Correlated surface errors** occur when deformations across the antenna surface are not random but follow a pattern — for example, due to panel misalignments.
- These errors generate an **error scatter pattern**, a secondary radiation pattern that can interfere with imaging.
- The width of this pattern is determined by the physical size scale of the correlated structure, such as the dimensions of individual panels.
- In some cases, the error pattern level can exceed the sidelobe level, introducing significant contamination into radio images.

Antenna Gain and Elevation Effects

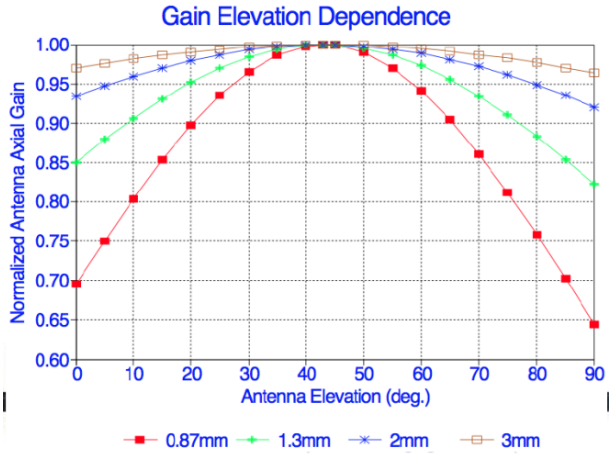


Figure 21: Gain vs Declination

Antenna efficiency is generally highest at mid-elevation angles, typically around 45° to 60°. At these angles, the structure experiences a balanced distribution of gravitational forces, minimizing distortions in the surface shape.

At very low elevations, increased ground spillover and mechanical stress reduce efficiency, while at high elevations (close to zenith), pointing challenges and structural compression can also degrade performance. Active surface systems aim to flatten these variations and maintain consistent efficiency across all elevations.

- The **on-axis antenna gain** (maximum sensitivity) changes with elevation angle due to gravitational forces.
- As the antenna tilts during tracking, its **backup structure** (which supports the reflective surface) deforms slightly under its own weight.
- These deformations redistribute the surface shape and lead to small changes in focus, gain, and beam shape, especially at high frequencies.
- To counteract these effects, some modern radio telescopes use an **active surface** — a system of actuators placed behind the reflective panels.
- The active surface adjusts the shape of the antenna in real time based on elevation angle and structural models, preserving the optimal parabolic shape.
- This technology helps maintain high aperture efficiency and beam quality across a wide range of elevations.

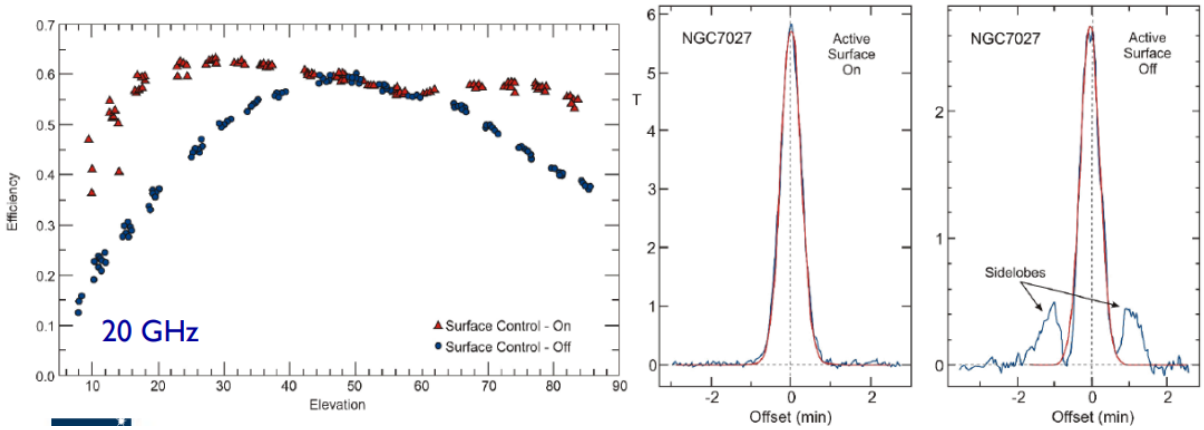


Figure 22: Effects after using Active Surfaces

3 Radio Interferometry

3.1 Why Interferometry?

- In astronomy, we aim to measure the **angular distribution** of emission from distant sources — including variations with frequency, polarization, and time.
- This means resolving the **brightness distribution**, which requires high angular resolution.
- Distant sources are extremely faint and compact — even the strongest radio source, Cygnus A, delivers only 1 watt across the entire Earth in a 1 GHz bandwidth.
- For perspective, 1 arcsecond is about the angular size of a US quarter viewed from 3 miles away.
- Due to the wave nature of light, resolution is limited by **diffraction**:

$$\theta \approx \frac{\lambda}{D}$$

- To achieve 1 arcsecond resolution at 21 cm wavelength, we would need a dish about 42 km wide — not physically feasible.
- The largest single-dish steerable telescopes, like those at Bonn and Green Bank, are only 100 meters in diameter — far too small for such resolution.
- **Interferometry** allows us to overcome this by using multiple antennas spread over large distances to simulate a much larger aperture.

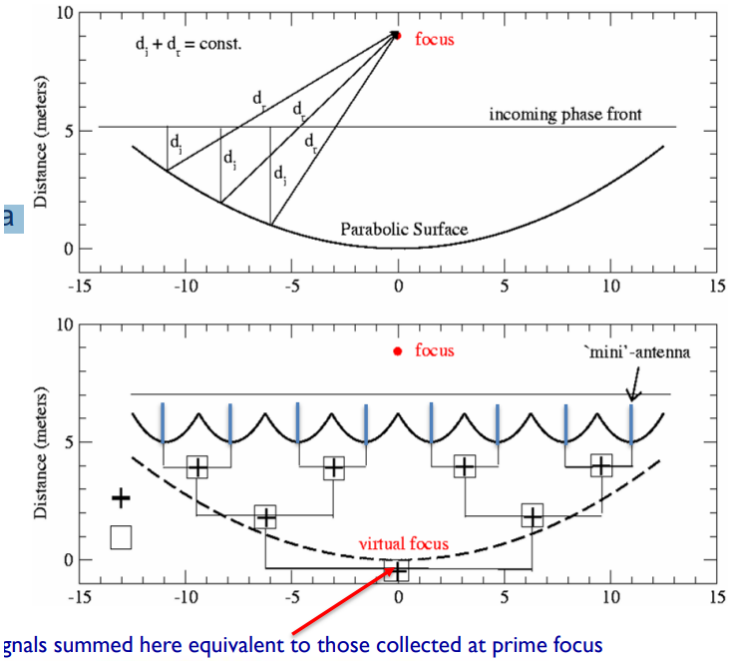


Figure 23

Basic Concept of Interferometry

- A **parabolic dish** works by coherently summing incoming electromagnetic (EM) fields at its focal point.
- The same effect can be achieved by combining the **voltages** from multiple antennas through a wired network.
- These antennas do not need to be physically adjacent — they can be widely separated.
- This forms the core idea of **interferometry**: the coherent summation of signals from multiple antennas to simulate a larger aperture and improve resolution.

3.2 Role of Sensor

- Coherent interferometry relies on the ability to **correlate electric fields** measured at spatially separated antennas.
- This requires converting the electric field $E(\vec{r}, \hat{n}, t)$ at a location $\vec{r}$ into a voltage signal $V(\hat{n}, t)$ that can be transmitted to a central processor **without losing phase information**.
- An **antenna** (or sensor) serves as the interface between incoming EM waves and electronics — converting the local field into a voltage that preserves both amplitude and phase.

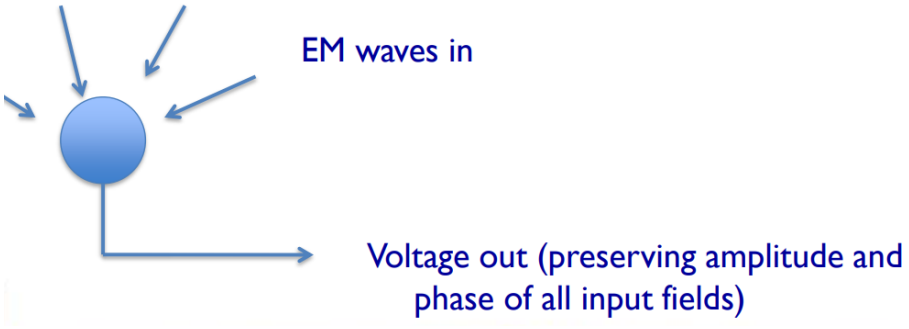


Figure 24

Idealized Two-Element Interferometer Assumptions

- Sensors are **isotropic** and fixed in space — no motion or rotation.
- Incoming radiation is **quasi-monochromatic** (narrowband).
- The source is a **point source in the far-field**, satisfying:

$$D \gg \frac{B^2}{\lambda}, \quad \theta \ll \frac{\lambda}{B}$$

- No frequency conversions — this is a direct RF (radio frequency) interferometer.
- Only a **single polarization** is considered.
- Assumes **ideal propagation** — no ionospheric or atmospheric distortions.
- All electronics are ideal: perfectly linear, frequency- and direction-independent, identical for both antennas, and noiseless.

3.3 Quasi-Monochromatic Radiation

- Mathematical analysis of wideband noise is very complex and often impractical.
- The analysis becomes simpler if the electric fields are assumed to be **monochromatic** (single frequency).
- However, natural astronomical radiation is never truly monochromatic.
- To make progress, we consider **quasi-monochromatic** radiation — where the bandwidth $\Delta\nu$ is small but non-zero.
- Over a short time interval $\Delta t \approx 1/\Delta\nu$, the electric field behaves like a pure sinusoid with approximately constant amplitude and phase.

- The electric field can then be written as:

$$E(t) = E_0 \cos(2\pi\nu_0 t + \phi)$$

where:

  - E_0 is the amplitude,
  - ν_0 is the central frequency,
  - ϕ is the phase (assumed constant over Δt).

- This approximation forms the basis for visibility analysis in interferometry.

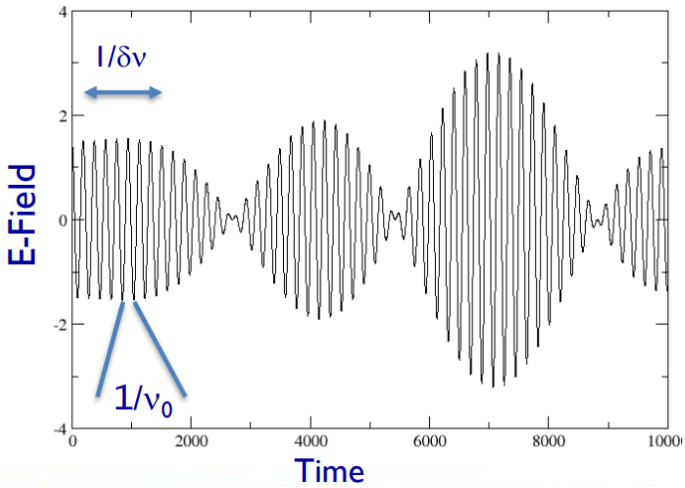


Figure 25: The figure shows an 'oscilloscope' trace of a narrow bandwidth noise signal. The period of the wave is T1=1/n0 , the duration over which the signal is closely sinusoidal is T2 1/dn. There are N T2 /T1 n0 /dn oscillations in a 'wave packet

Stationary Quasi-Monochromatic RF Interferometer

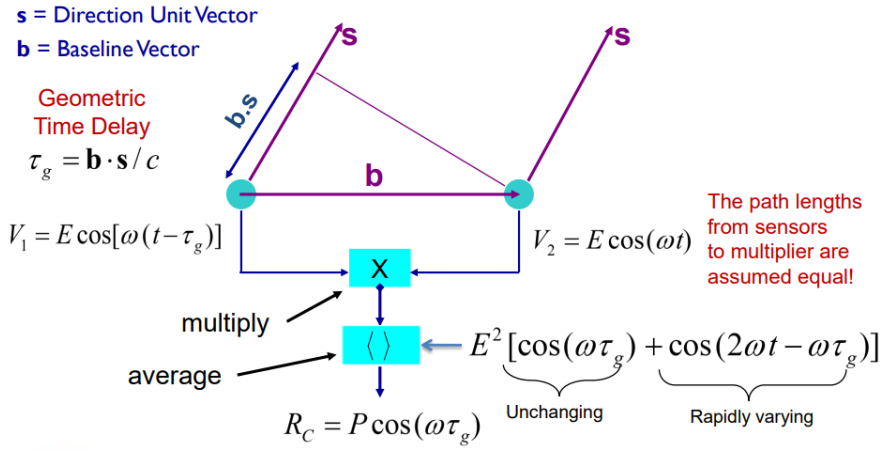


Figure 26

- In a basic interferometer setup with quasi-monochromatic radiation, we analyze the **averaged product** of the voltages received at two antennas.
- This quantity, denoted R_C, depends on the received power and the geometric delay between the two antennas.
- The received power is proportional to $P = \frac{E^2}{2}$, where E is the electric field amplitude.
- The geometric delay τ_g arises from the difference in path length due to the baseline orientation and the source direction.
- The correlation output is:

$$R_C = P\cos(2\pi\nu\tau_g)$$

- In terms of the baseline vector $\vec{B}$ and source direction $\hat{s}$, the geometric delay is:

$$\tau_g = \frac{\vec{B}\cdot\hat{s}}{c} \quad \Rightarrow \quad R_C = P\cos\left(\frac{2\pi\nu}{c}\,\vec{B}\cdot\hat{s}\right)$$

Key Properties

- R_C is independent of the **time of observation**, as long as the source is not intrinsically variable.
- It does not depend on the **absolute location of the antennas**, provided the source lies in the far-field.
- It is also independent of the **absolute phase** or distance to the source — again, assuming a far-field approximation.

3.4 Angular Perspective

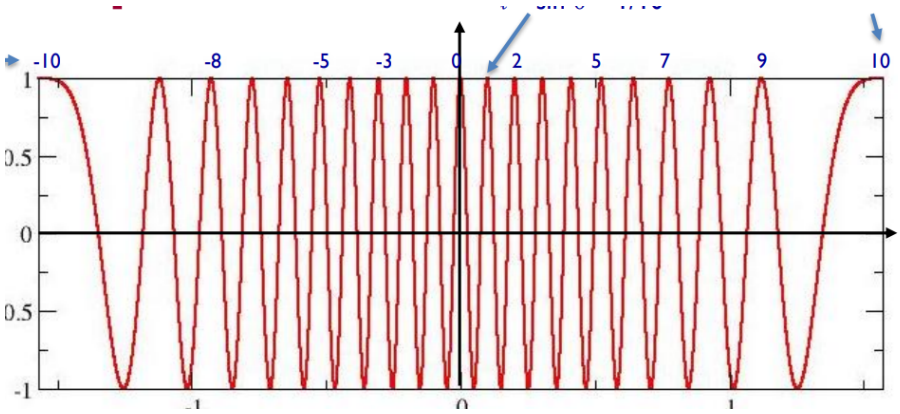


Figure 27

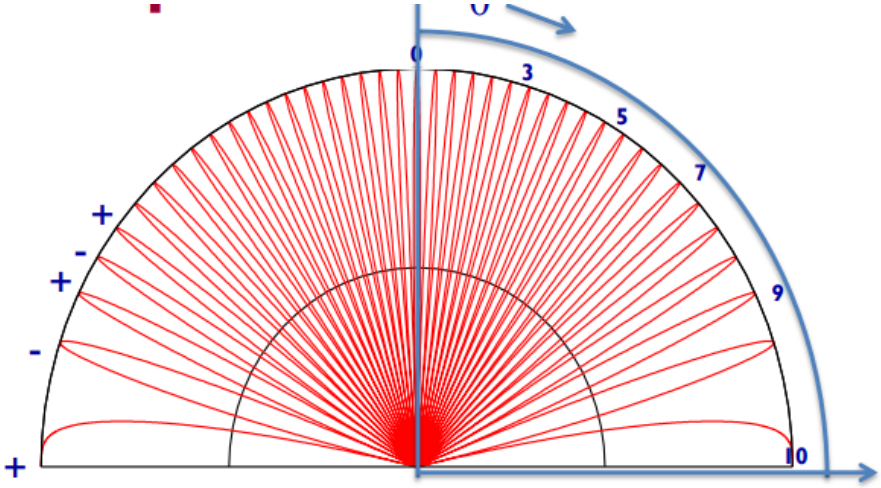


Figure 28: The absolute value of the response for u = 10, as a function of angle. The 'lobes' of the response pattern alternate in sign.

- The previous plot represents a **meridional cut** through the fringe pattern — a 2D slice taken along the direction of the baseline vector $\vec{b}$.
- In full 3D space, the fringe pattern forms a set of **coaxial cones**, centered and aligned along the baseline vector.
- These cones correspond to surfaces of equal geometric delay, where the interferometer output remains constant.
- When visualized in 2D (e.g., for $u = 4$), the fringe pattern appears as alternating bright and dark regions — a simplified cross-section of the full cone structure.

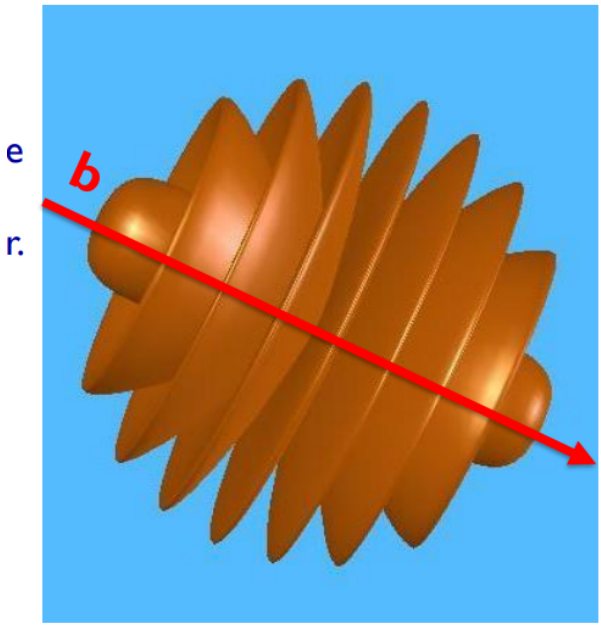


Figure 29

- Viewed directly along the baseline vector, the fringes form a **bulls-eye pattern** — a series of concentric circles centered on the phase center.

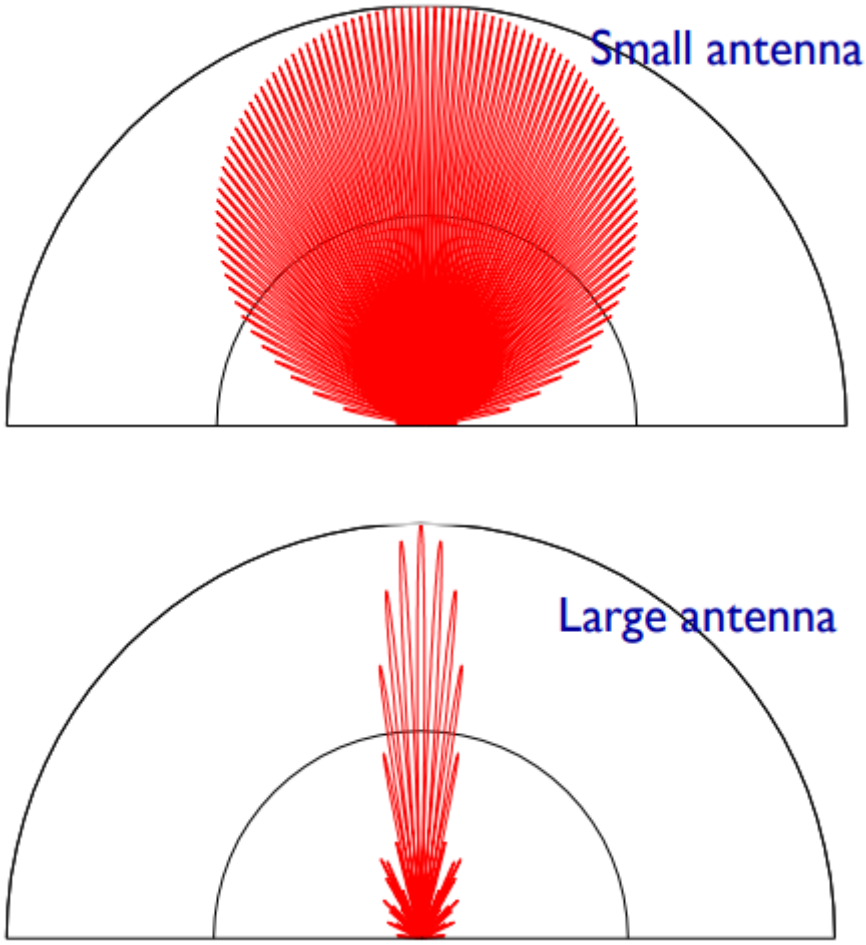


Figure 30: Top Panel: The interferometer pattern with a cos(q)-like sensor response. • Bottom Panel: A multiplewavelength aperture antenna has a narrow beam, but also sidelobes.

3.5 Response from an Extended Source

- For an extended source, the voltage output from a single antenna is the spatial integral of the electric field E over the antenna's primary beam:

$$V(t) = \iint E(\hat{s}, t)\, d\Omega$$

- The **correlator response** between two antennas becomes:

$$R_C = \langle V_1(t) V_2^*(t) \rangle$$

- Assuming the emission is **spatially incoherent**, and interchanging the averaging

and integration operations, the response simplifies to:

$$R_C = \iint I_\nu(\hat{s}) \cos\left(\frac{2\pi\nu}{c}\vec{B}\cdot\hat{s}\right) d\Omega$$

- Here, $I_\nu(\hat{s})$ is the **brightness distribution** on the sky, and $\vec{B}\cdot\hat{s}$ represents the geometric delay.
- This integral directly links the **measured interferometer response** R_C to the **sky brightness** $I_\nu(\hat{s})$, modulated by the interferometric fringe pattern.

3.6 Fringes

- In interferometry, we don't directly image the sky — instead, we observe how the **sky brightness is projected onto the interferometer's fringe pattern**.
- Each interferometer baseline samples the sky by measuring how well the brightness pattern aligns with its fringe pattern — this is mathematically an **inner product**.
- The observed visibility $V(u,v)$ is:

$$V(u,v) = \iint I(l,m)\, e^{-2\pi i(ul+vm)}\, dl\, dm$$

- This is exactly the **2D Fourier transform** of the sky brightness $I(l,m)$, where (u,v) are the baseline components in units of wavelength.
- Each pair of antennas samples one spatial frequency — i.e., one point in the (u,v)-plane — and records a visibility.
- Thus, image reconstruction in radio astronomy is fundamentally a process of **inverse Fourier transforming** these measured visibilities.

3.7 Mathematics

Any real function $I(\vec{s})$ can be decomposed into its **even** and **odd** components:

$$I_{\text{even}}(\vec{s}) = \frac{I(\vec{s}) + I(-\vec{s})}{2}, \quad I_{\text{odd}}(\vec{s}) = \frac{I(\vec{s}) - I(-\vec{s})}{2}$$

Suppose the sky brightness has a component with **odd symmetry**, i.e., $I(-\vec{s}) = -I(\vec{s})$. The interferometer fringe pattern is a cosine function, which is even:

$$\cos\left(\frac{2\pi\nu}{c}\vec{B}\cdot\vec{s}\right) = \cos\left(\frac{2\pi\nu}{c}\vec{B}\cdot(-\vec{s})\right)$$

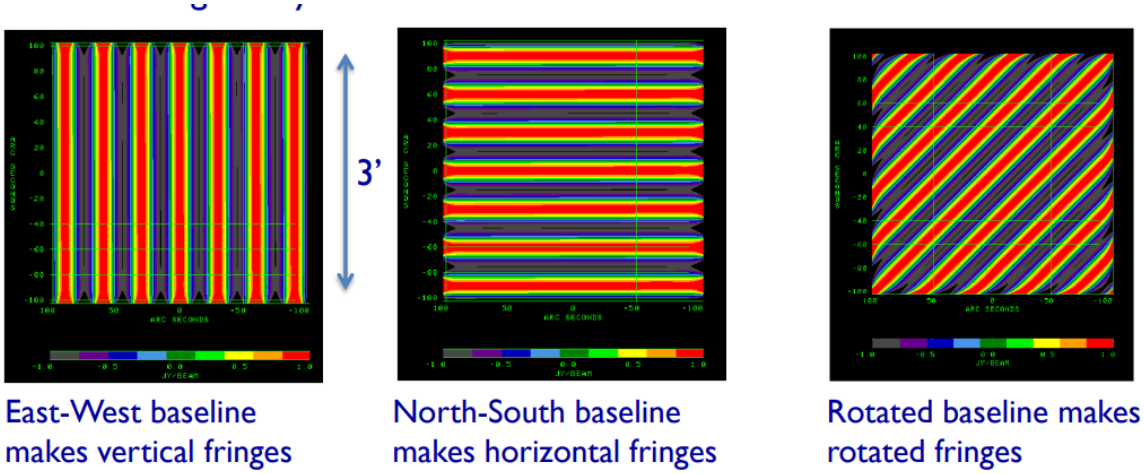


Figure 31: Fringes oreints as baselines orients

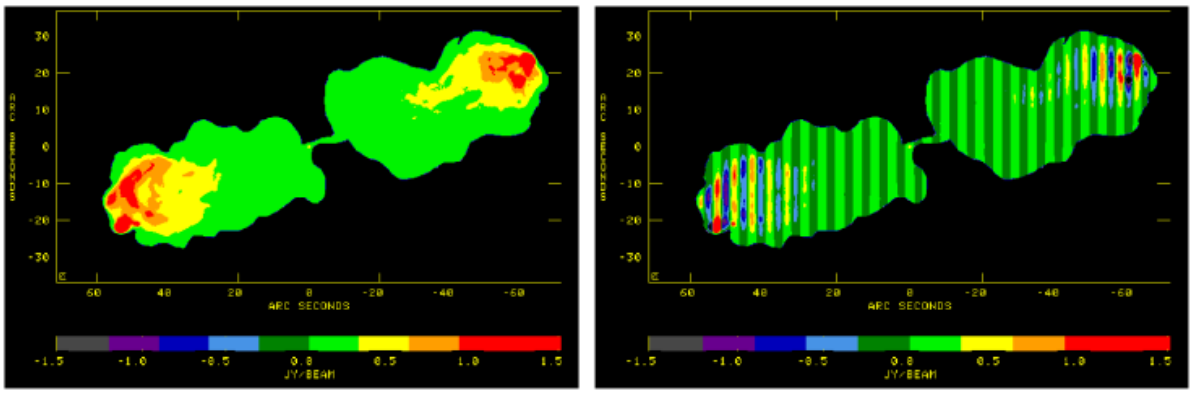


Figure 32: What we actually observe - Actual intensity pattern projected over fringe pattern

Hence, the correlator response to an odd-symmetric brightness distribution is zero:

$$R_C = \iint I_{\text{odd}}(\vec{s}) \cos\left(\frac{2\pi\nu}{c}\vec{B}\cdot\vec{s}\right) d\Omega = 0$$

So the cosine correlator captures only the **even part** of the brightness distribution:

$$R_C = \iint I_{\text{even}}(\vec{s}) \cos\left(\frac{2\pi\nu}{c}\vec{B}\cdot\vec{s}\right) d\Omega$$

To measure the **odd-symmetric** components, we need a **sine fringe pattern**, which is an odd function. This is achieved by introducing a **90° phase shift** in one antenna, forming a sine correlator:

$$R_S = \iint I_{\text{odd}}(\vec{s}) \sin\left(\frac{2\pi\nu}{c}\vec{B}\cdot\vec{s}\right) d\Omega$$

Now, we define the **complex visibility**:

$$V(\vec{B}) = R_C - iR_S$$

Substituting the integrals gives:

$$V(\vec{B}) = \iint I(\vec{s})\, e^{-2\pi i\vec{B}\cdot\vec{s}/\lambda}\, d\Omega$$

This is precisely the **Fourier transform** of the sky brightness $I(\vec{s})$, and forms the foundation of radio interferometric imaging.

3.8 Real Sensor Effects

- The idealized isotropic sensors assumed earlier are not physically realizable.
- Real antennas have a **directional voltage gain pattern** $A(\vec{s}, \vec{s_0})$, where:
  - $\vec{s}$ is the incoming sky direction,
  - $\vec{s_0}$ is the antenna's pointing direction.
- This gain pattern affects the observed sky signal by attenuating it based on direction, both in amplitude and phase.
- The modified interferometric response becomes:

$$V(\vec{B}) = \iint A_1(\vec{s})A_2^*(\vec{s})I(\vec{s})\, e^{-2\pi i\vec{B}\cdot\vec{s}/\lambda}\, d\Omega$$

- Here, A_1 and A_2 are the **complex voltage gain functions** of the two antennas, incorporating both amplitude and phase effects.

- If both antennas are identical and perfectly aligned, $A_1 A_2^* = |A(\vec{s})|^2$, the **power gain response**.
- A common approach to correct this effect is to **divide the image by the antenna beam** (primary beam correction).
- However, this correction can amplify noise and is not always optimal — especially near the beam edges where gain is low.

3.9 Finite Bandwidth

- Real interferometers operate with a **finite, non-zero bandwidth** $\Delta\nu$.
- Each frequency component within the band generates its own fringe pattern, with angular spacing roughly given by:

$$\theta \sim \frac{\lambda}{B}$$

  where B is the baseline length.
- All fringe patterns share a central maximum at the $n = 0$ fringe (the meridional plane).

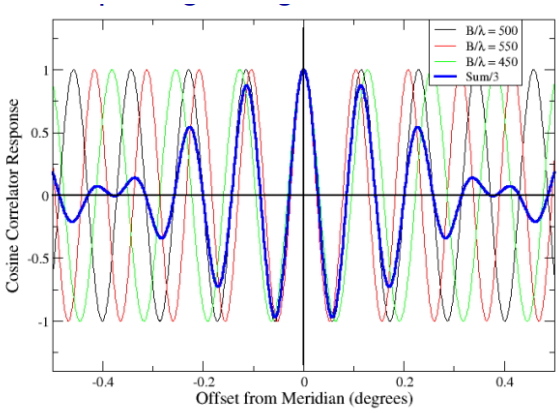


Figure 33: For a bandwith it appears as a wave packet, blue coulured is what we measure

- The total response over the finite bandwidth is computed by integrating the visibility across frequencies:

$$V = \iint \left(\frac{1}{\Delta\nu} \int_{\nu_0 - \Delta\nu/2}^{\nu_0 + \Delta\nu/2} I(\vec{s}, \nu)\, G_1(\nu) G_2^*(\nu)\, e^{-i2\pi\nu\tau_g}\, d\nu \right) d\Omega$$

  where:

  - G_1, G_2 are the frequency-dependent gain functions,
  - $\tau_g = \vec{B} \cdot \hat{s}/c$ is the geometric delay.

- If the source intensity $I(\vec{s}, \nu)$ is constant across the bandwidth and $G_1 = G_2 = 1$, the integral simplifies using the Fourier identity:

$$V = \iint I(\vec{s}) \cdot \mathrm{sinc}(2\pi\Delta\nu\,\tau_g) \cdot e^{-i2\pi\nu_0\tau_g}\, d\Omega$$

  where:

$$\mathrm{sinc}(x) = \frac{\sin(x)}{x}$$

- The sinc term causes **bandwidth decorrelation** — attenuation of the fringes away from the meridional plane.

- **Observations off the meridian** are particularly affected:
  - The maximum response is at the meridional plane ($\tau_g = 0$).
  - Each baseline has its own meridional plane, so only a small sky patch near the **zenith** avoids attenuation for all baselines.

- To observe sources off-center, we **shift the fringe pattern** toward the desired direction by adding a compensating **time delay** to the antenna closer to the source.

- This process is called **geometric delay compensation**, and is essential in wide-field imaging.

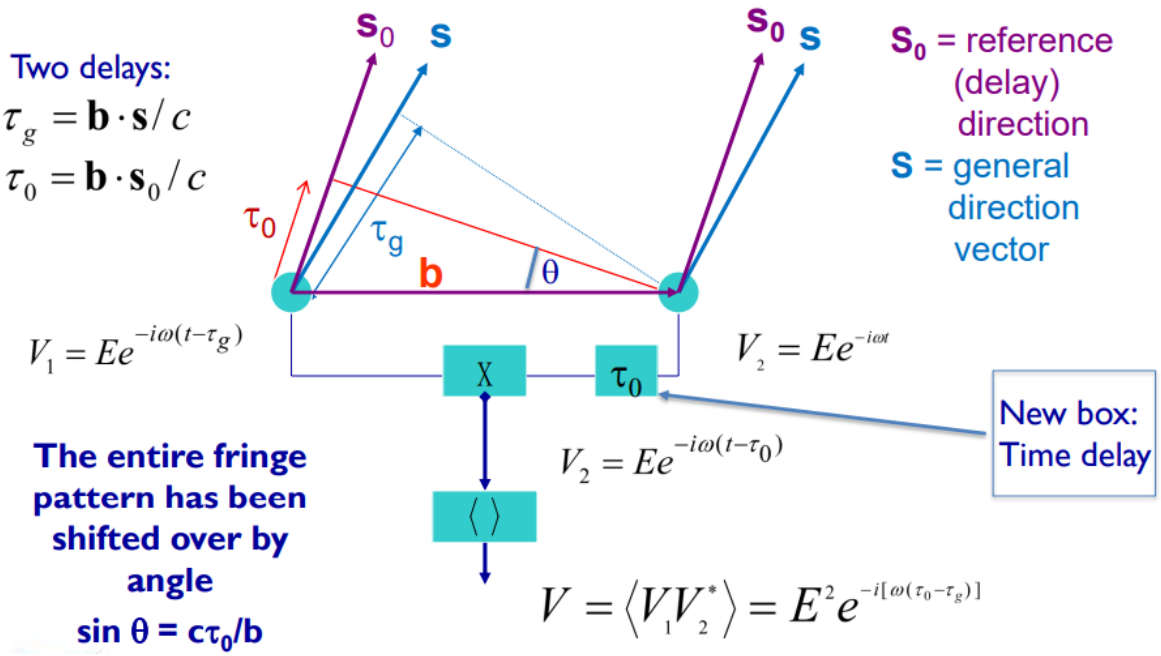


Figure 34: Time delay box adds time

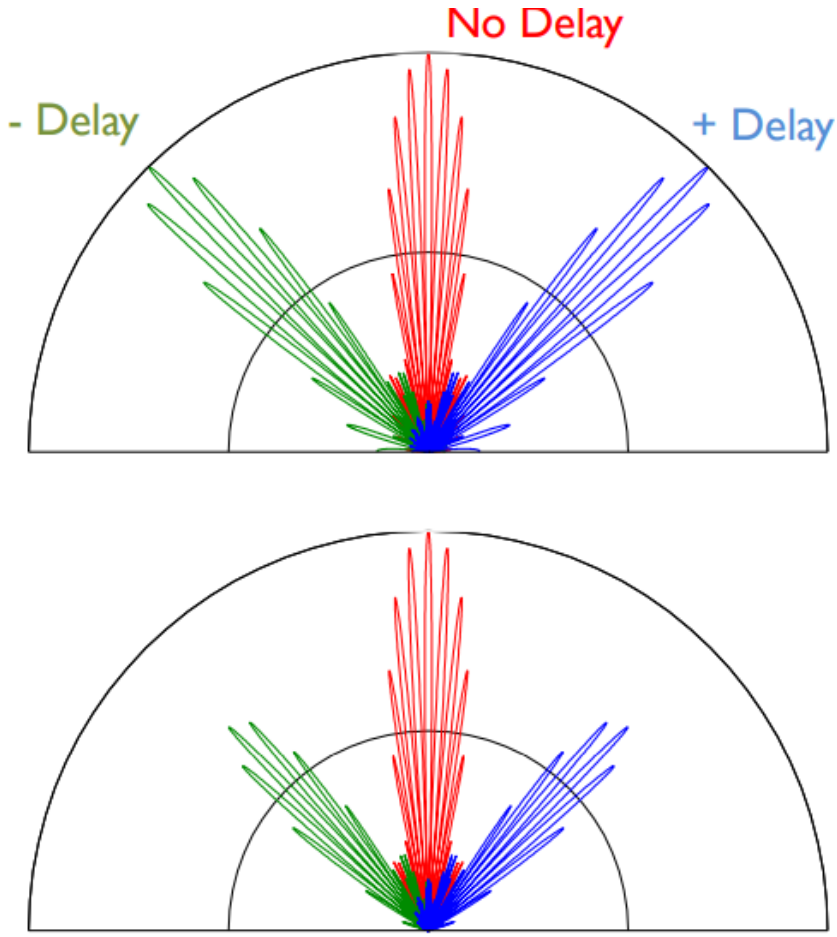


Figure 35: • Top Panel: Delay has been added and subtracted to move the delay pattern maximum to the source location, presuming an isotropic sensor. • Bottom Panel: A cosinusoidal sensor pattern is added, to illustrate losses from a fixed sensor.

3.10 Rotating Reference Frames

- Most interferometers are located on the surface of the Earth, a **rotating platform**.
- From the Earth's frame, celestial sources appear to move across the sky due to Earth's rotation.
- To track a moving source, we **continuously shift the coherence pattern** (fringe pattern) of the interferometer by adjusting the geometric delay.
- This is achieved by adding a time delay Δt, adjusted continuously with high precision:

$$\Delta t \frac{1}{10\Delta\nu}$$

to avoid bandwidth decorrelation.

- Adjusting time delay allows the fringes to remain centered on the moving source, preventing **fringe washing**.
- However, there's also a **phase effect** — as the source moves, it crosses the fringe pattern, introducing rapid phase changes.
- The rate at which fringes move across the sky due to Earth's rotation is called the **natural fringe rate**:

$$f_{\text{fringe}} = u\,\omega\cos\delta$$

- To avoid decorrelation due to fringe phase changes:
  - Delays must be updated frequently, and
  - **Phase shifts** must be applied to keep the signal coherent.

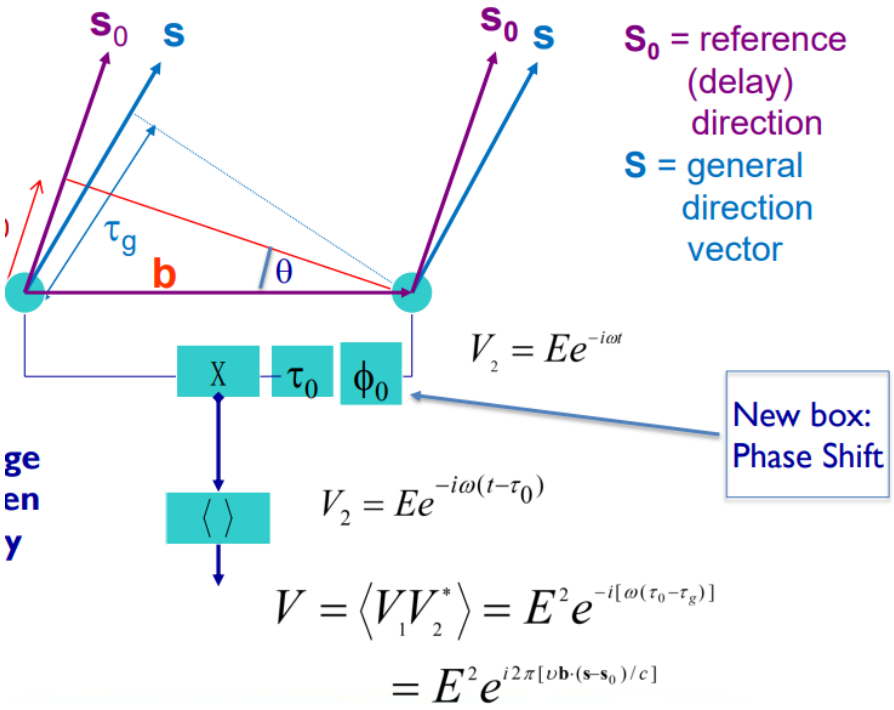


Figure 36

- Therefore, modern interferometers correct both:
  - **Time delays** — to maintain coherence across bandwidth.
  - **Phase shifts** — to keep up with fringe movement and reduce sample rate demands.

4 Geometry

4.1 (u,v) Planes and (u,v,w) Volumes

- Suppose measurements of visibility $V_\nu(\vec{b})$ are taken entirely on a plane.
- A significant simplification occurs if we align our coordinate system such that one axis is **normal to this plane**.
- Define the coordinate system as:
$$(u, v, w)$$
where:
  - u, v, and w are the components of the baseline vector $\vec{b}$ measured in units of wavelength,
  - w is aligned **perpendicular to the measurement plane**.
- The unit direction vector $\vec{s}$ to a source in the sky has components:
$$\vec{s} = (l, m, n)$$
where:
$$n = \sqrt{1 - l^2 - m^2}$$
- The simplification arises because $|\vec{s}| = 1$, so only two variables (l, m) are needed to describe the direction.
- A downside of 2D (u, v)-coverage is that it assumes all baselines lie in a plane, neglecting the third dimension w.
- This leads to degraded resolution in directions not aligned with the interferometer's primary plane.
- **Example:**
  - East-West (E–W) interferometers provide no North-South (N–S) resolution for sources near the **celestial equator**.
  - A snapshot observation with the **VLA** of a source near the **horizon** will have poor resolution in the vertical (elevation) direction.
- In general, resolution is poor along directions p

4.2 3-D ('Volume') Interferometers

• Over time, their baselines move through a (u,v,w) volume

• Arrange 'w' to point to the source (phase tracking center), and orient the (u,v) plane so the 'v' axis points towards the north, and 'u' towards the east

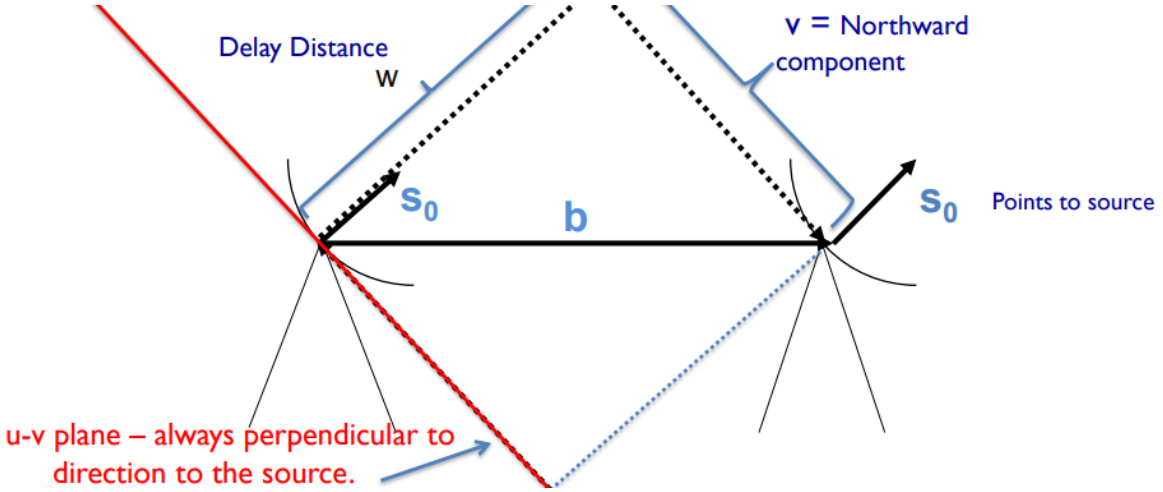


Figure 37

Solutions to Non-Coplanar Baselines

- **1. 3D Transform**:
  - Grid visibilities in 3D and perform a 3D Fourier transform.
  - Sky appears on a unit sphere.
  - *Downside:* Computationally expensive and mostly sparse.
- **2. Snapshot Imaging**:
  - Sum a series of 2D snapshots.
  - Requires reprojecting coordinates for each snapshot.
  - Limited deconvolution accuracy.
- **3. Faceted Imaging**:
  - Divide the sky into small facets.
  - Apply local phase corrections and recalculated baselines per facet.
  - Enables localized calibration.
- **4. W-Projection**:
  - Mathematically project visibilities onto the $w = 0$ plane.
  - Treats array as coplanar without resampling the sky.

4.3 UVW Coordinates and Baseline Loci

- The baseline vector in Earth coordinates is $\vec{B} = (B_x, B_y, B_z)$.
- To convert to interferometric coordinates (u, v, w), we use the transformation:

$$\begin{bmatrix} u \\ v \\ w \end{bmatrix} = \frac{1}{\lambda} \begin{bmatrix} \sin H & \cos H & 0 \\ -\sin\delta\cos H & \sin\delta\sin H & \cos\delta \\ \cos\delta\cos H & -\cos\delta\sin H & \sin\delta \end{bmatrix} \begin{bmatrix} B_x \\ B_y \\ B_z \end{bmatrix}$$

- H is the hour angle, and δ is the source declination.
- Over 24 hours, each baseline traces an **ellipse** in the (u, v) plane:

$$u^2 + (v - B_z \cos\delta_0 \sin\delta_0)^2 = B_x^2 + B_y^2$$

  where a and b depend on the baseline length and source declination.
- Since sky brightness $I(\vec{s})$ is real, visibility satisfies:

$$V(-u, -v) = V^*(u, v)$$

  so each observation contributes two symmetric points in the (u, v) plane.
- For purely East–West baselines ($B_x = B_z = 0$), the ellipse lies along the u-axis with no vertical (v) extent.

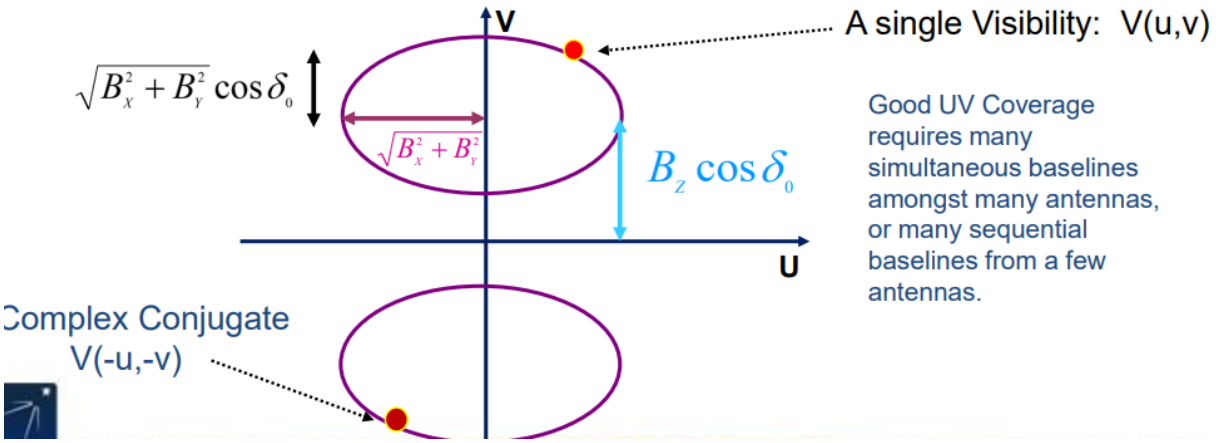


Figure 38

- E–W interferometer arrays are poorly suited for observing sources at low declinations.
- At $\delta = 0°$, the (u, v) coverage collapses to a **line**, resulting in poor resolution in the N–S direction.

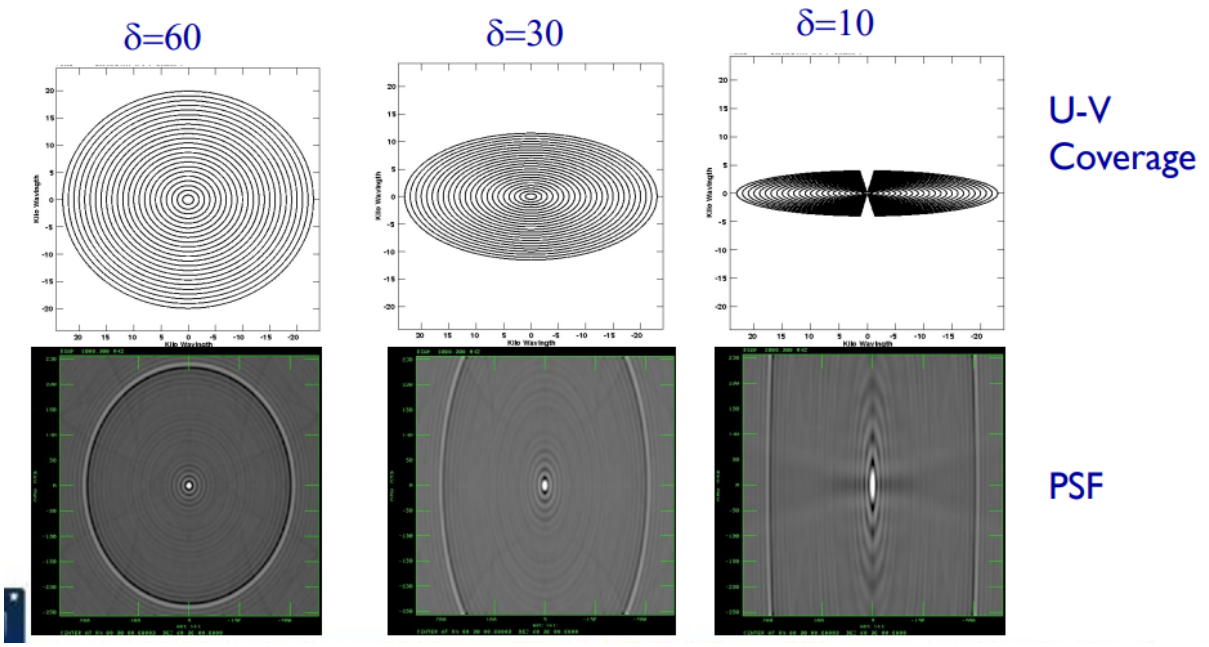


Figure 39

– To achieve good 2D angular resolution at all declinations, **N–S baseline components** must be included.

– This requires a more complex array layout with **many more antennas** to ensure adequate (u, v)-plane sampling.

4.4 Examples of Visibilities

A Point Source

– Consider an unresolved point source located at the **phase center**.

– Its brightness distribution is a delta function:

$$I(l, m) = I_0\, \delta(l, m)$$

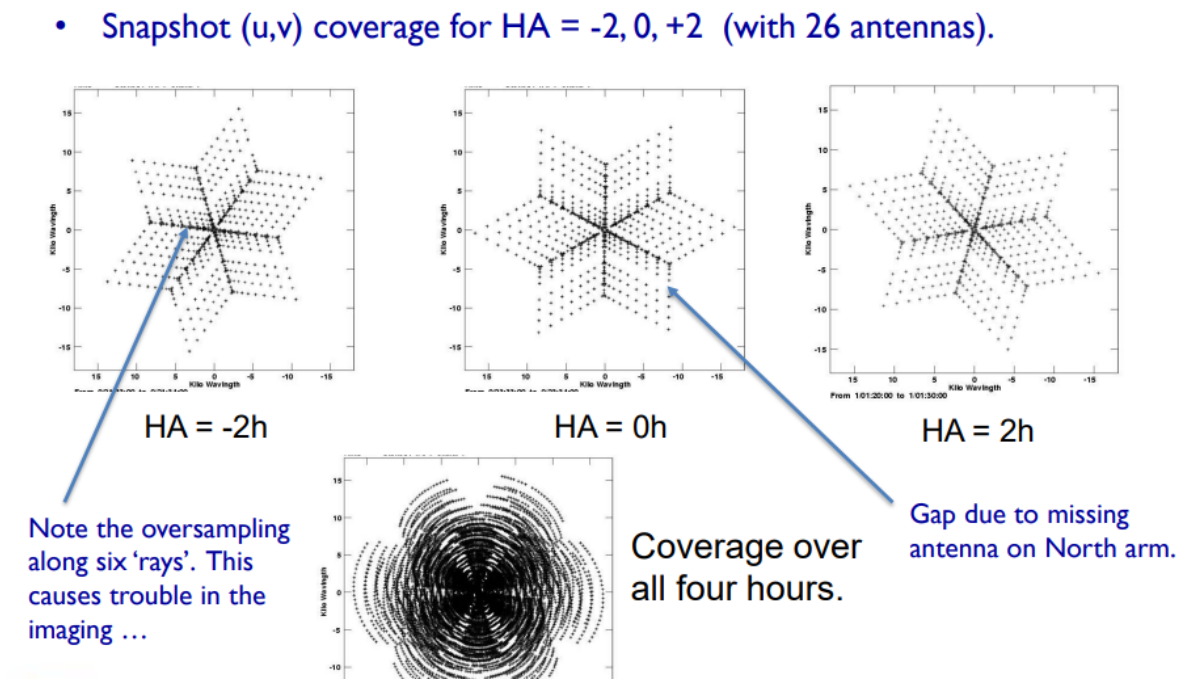


Figure 40

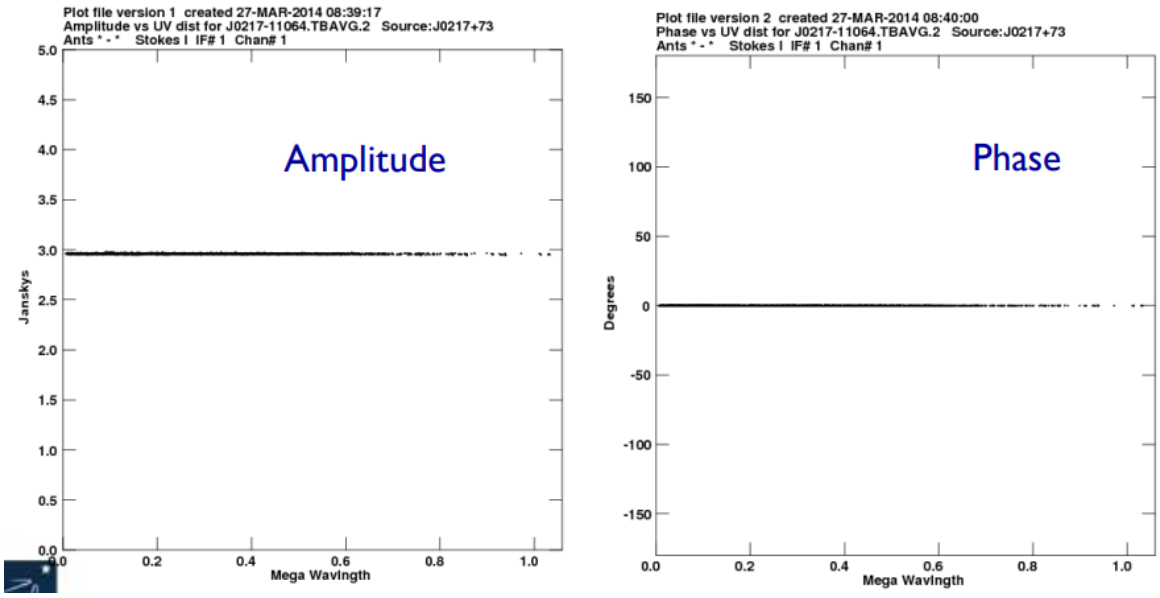


Figure 41

– The visibility function is the 2D Fourier transform of the brightness:

$$V(u,v) = \iint I(l,m)\, e^{-2\pi i(ul+vm)}\, dl\, dm = I_0$$

– Therefore, the visibility is a **constant** across all baselines.

– In real observations, zooming into the visibilities reveals **noise** and possibly hidden **structure**.

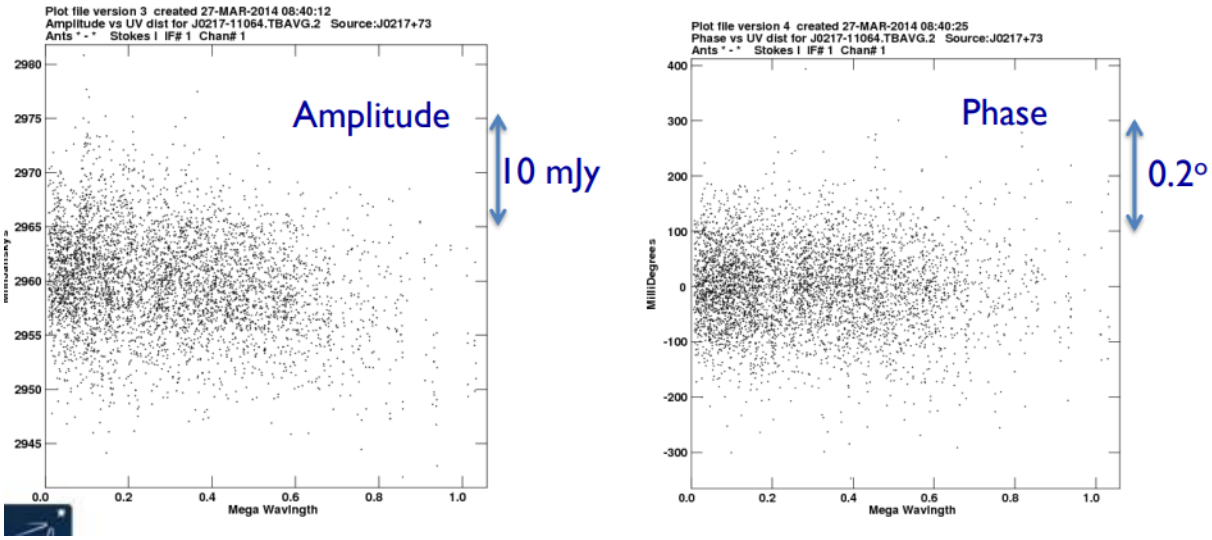


Figure 42

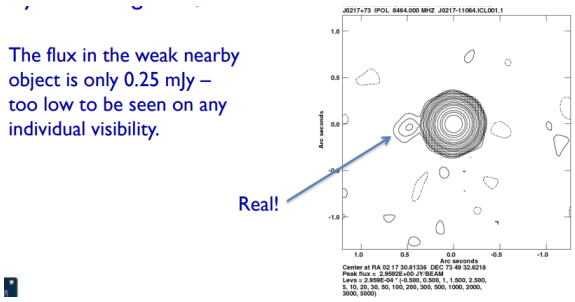


Figure 43

4.5 3C273 – The First Known Quasar

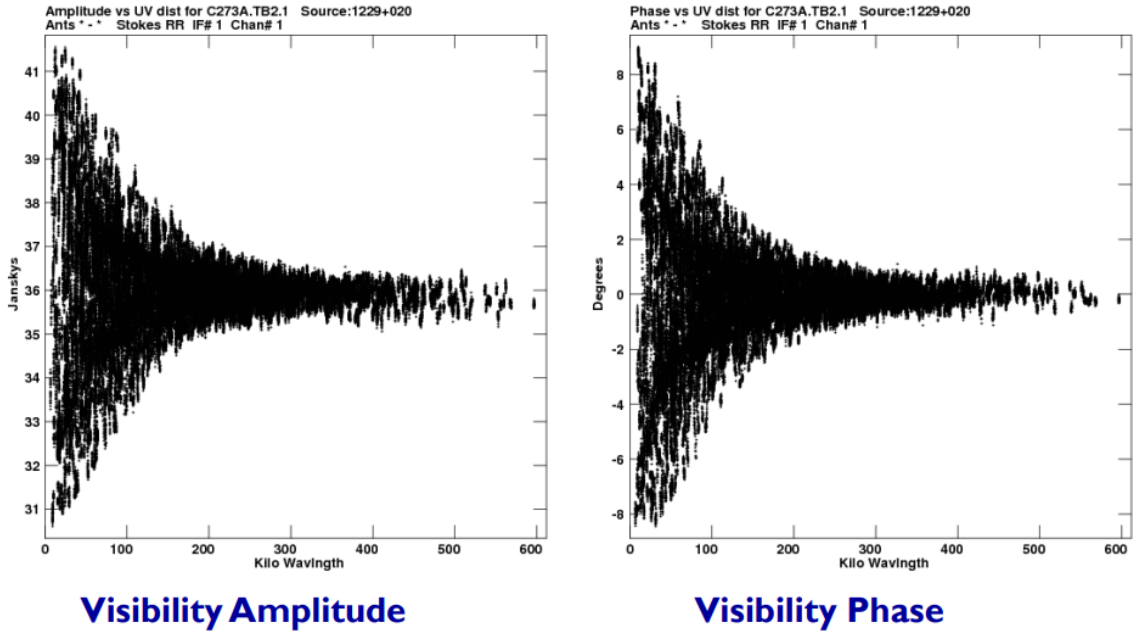


Figure 44

- The visibility amplitudes remain high on long baselines, showing:
  - A **36 Jy unresolved point source**.
- The zero phase and lack of a phase gradient across baselines indicate:
  - The point source is **centered at the phase center**.
- Visibility amplitude drops off at ~ 100–$200\,\mathrm{k}\lambda$, meaning:
  - The **extended structure is resolved out**.
  - Angular size of extended emission is ~ 1–$2''$.
- Rapid oscillations in amplitude and phase imply:
  - Presence of a **larger, elongated structure**.

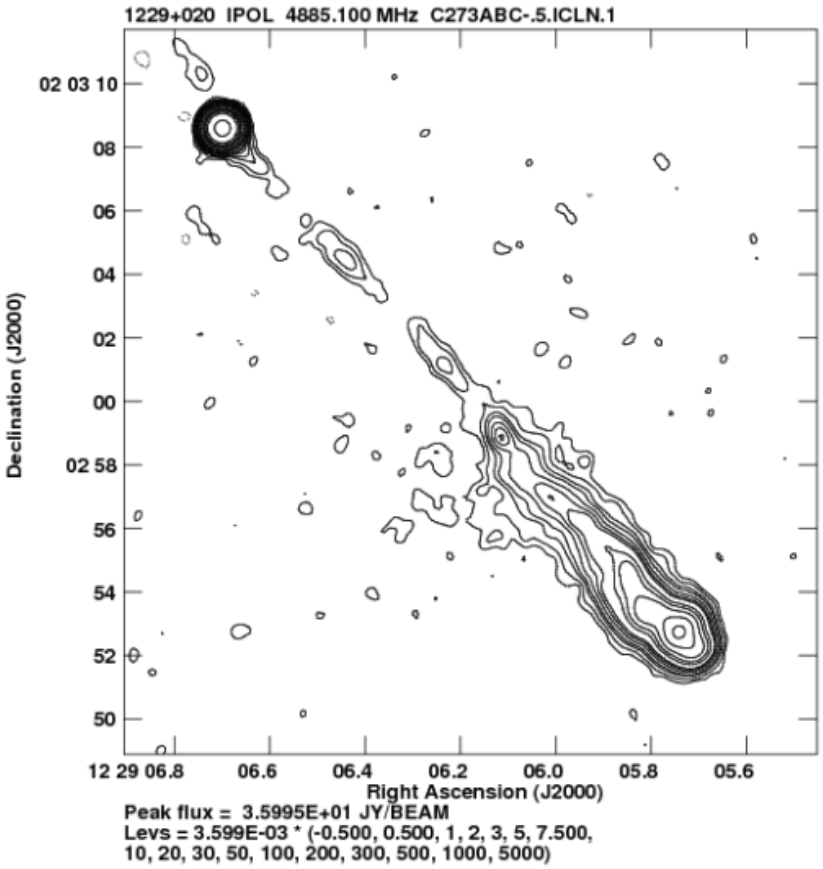


Figure 45: Final Image

5 Calibration

5.1 The Measurement Equation

Electromagnetic Signal Representation

– An incoming planar EM wave propagating along the z-axis is described by:

$$\mathbf{e} = \begin{bmatrix} e_x \\ e_y \end{bmatrix}$$

where e_x, e_y are the complex electric field components in orthogonal directions.

Jones Matrix Formalism

- Linear transformations from physical effects (e.g., atmosphere, feeds, electronics) are represented as:

$$\mathbf{e}' = \mathbf{J}\mathbf{e}$$

- Multiple effects are applied in sequence:

$$\mathbf{e}' = \mathbf{J}_n \dots \mathbf{J}_2 \mathbf{J}_1 \mathbf{e} = \mathbf{J}\mathbf{e}$$

- These transformations ultimately result in voltages at antenna feeds a and b:

$$\mathbf{v} = \begin{bmatrix} v_a \\ v_b \end{bmatrix} = \mathbf{J}\mathbf{e}$$

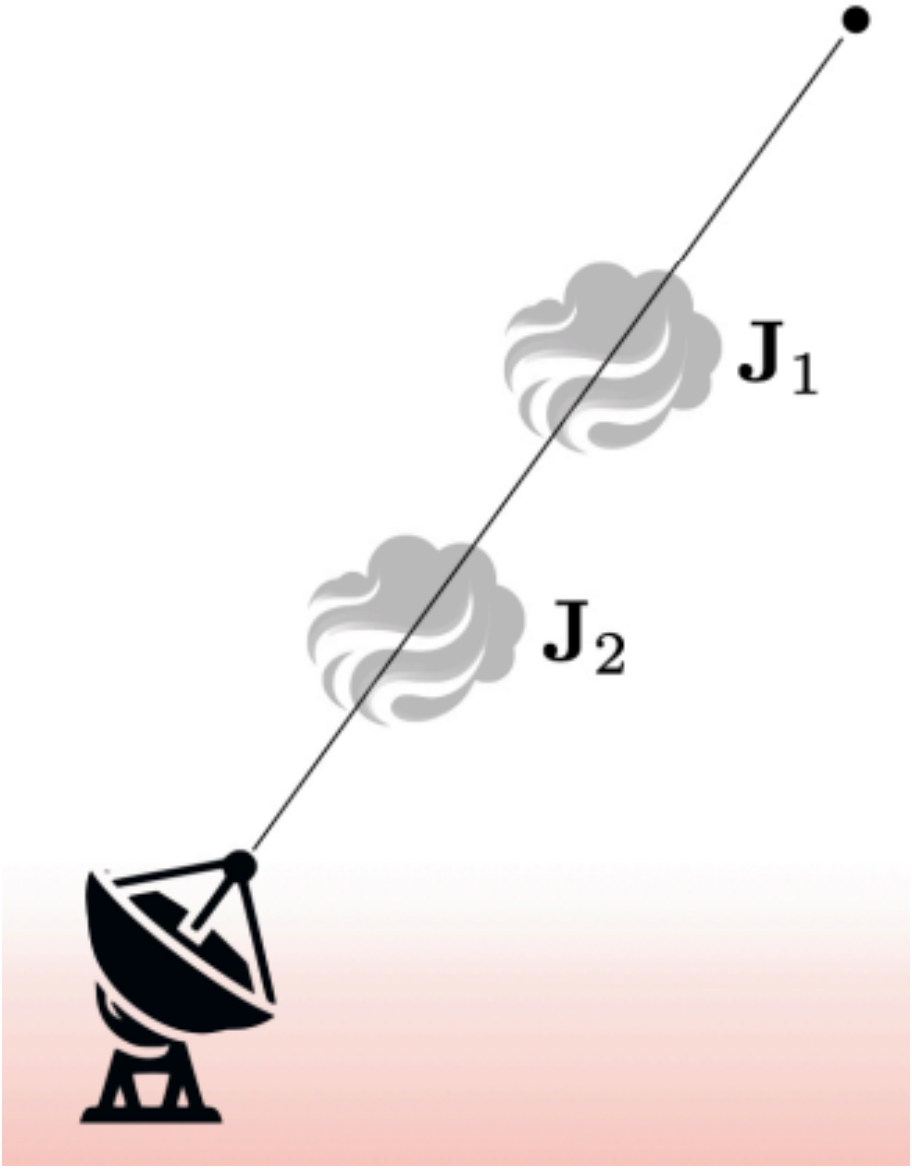


Figure 46

Observed Visibilities

– For a baseline $p - q$, the observed visibility matrix is:

$$\mathbf{V}_{pq} = \mathbf{J}_p \, \mathbf{B} \, \mathbf{J}_q^H$$

where:

∗ $\mathbf{J}_p, \mathbf{J}_q$ are Jones matrices for antennas p and q

* **B** is the **brightness matrix**, containing Stokes parameters:

$$\mathbf{B} = \begin{bmatrix} I+Q & U+iV \\ U-iV & I-Q \end{bmatrix}$$

– For a source from direction (l, m, n), the geometric delay adds a phase term:

$$\mathbf{K}_p = e^{-2\pi i(u_p l + v_p m + w_p(n-1))}$$

– Final form of the **Measurement Equation** becomes:

$$\mathbf{V}_{pq} = \mathbf{J}_p \, \mathbf{K}_p \, \mathbf{B} \, \mathbf{K}_q^H \, \mathbf{J}_q^H$$

or by defining source coherency $\mathbf{X}_{pq} = \mathbf{K}_p \, \mathbf{B} \, \mathbf{K}_q^H$:

$$\mathbf{V}_{pq} = \mathbf{J}_p \, \mathbf{X}_{pq} \, \mathbf{J}_q^H$$

– **We measure:** $\mathbf{V}_{pq}$ — complex visibility matrix.
– **We want to find:** Sky brightness **B** or coherency $\mathbf{X}_{pq}$.

Calibration

– Calibration solves for unknown Jones terms $\mathbf{J}_p$, $\mathbf{J}_q$, i.e., instrumental effects.
– This is done by minimizing:

$$\min \left\| \mathbf{D} - \mathbf{G}\mathbf{M}\mathbf{G}^H \right\|$$

where:

* **D**: Measured data (visibilities)
* **M**: Predicted model visibilities
* **G**: Antenna-based complex gain matrix

– Solved using iterative optimization (e.g., gradient descent).
– Once **G** is estimated, corrected data is obtained via:

$$\mathbf{D}_{\text{corr}} = \mathbf{G}^{-1}\mathbf{D} \left(\mathbf{G}^H\right)^{-1}$$

5.2 Calibrator Sources

Primary Calibrators

– A small set of well-monitored sources (e.g., 3C catalog objects or planets at higher frequencies).

- Observed every few hours to account for slowly-varying instrumental effects.
- Roles:
  - **Absolute Flux Calibration:** Converts correlator units to physical units (Jy); requires precisely known calibrator flux.
  - **Bandpass Calibration:** Corrects frequency-dependent gain variations; the calibrator's spectrum must be known accurately.
  - **Polarisation Calibration:** Calibrator's polarisation properties are used to correct the instrument's polarisation response.

Secondary Calibrators

- More numerous, but less precisely monitored than primaries.
- Chosen near the target source for time-dependent corrections (e.g., gain fluctuations, parallactic angle changes).
- Often observed frequently during the observation to track rapid instrumental changes.
- Requirements:
  - Preferably compact and bright (unity-amplitude point-source model).
  - VLBI-grade positional accuracy helps preserve astrometry.
  - Amplitude gains scaled to the primary calibrator to avoid flux ambiguity.
- Note: Spectral axis handling depends on instrument (e.g., SPWs in VLA/ALMA are handled natively).

5.3 Examples – What Calibration Can Actually Do!

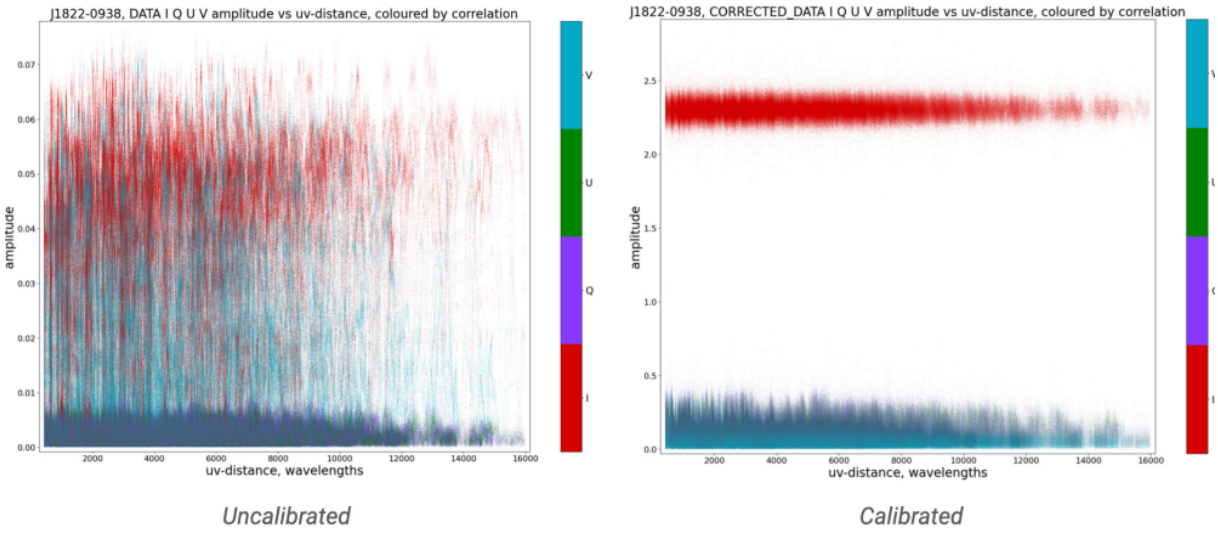


Figure 47

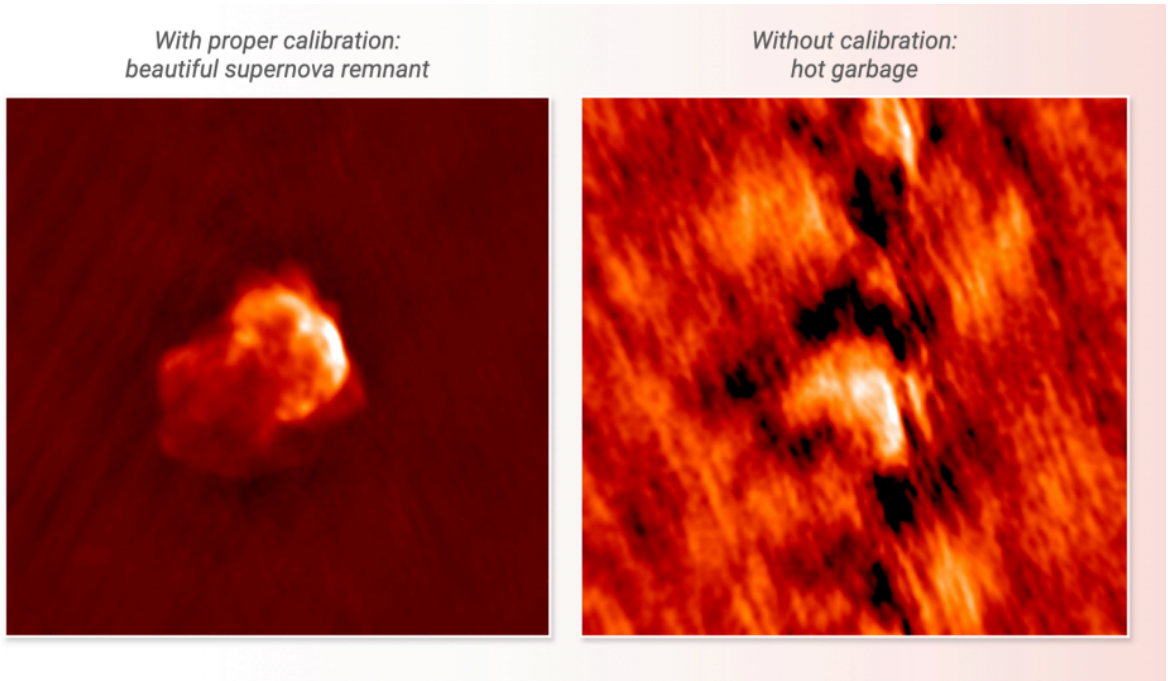


Figure 48

6 Introduction to Imaging and Deconvolution

6.1 Visibility

- In radio interferometry:
  - **Observed:** Complex visibilities $V(u, v)$, measured at different baseline spacings.
  - **Desired:** Brightness distribution on the sky $I(l, m)$.
- These are related through the **van Cittert-Zernike theorem**:

$$V(u,v) = \iint I(l,m)\, e^{-2\pi i(ul+vm)}\, \frac{dl\, dm}{\sqrt{1-l^2-m^2}}$$

  where:
  - $V(u, v)$: visibility function (measured)
  - $I(l, m)$: sky brightness distribution (desired)
  - (l, m): direction cosines on the sky
  - (u, v): baseline components in units of wavelength
- For small angles (i.e., $l^2 + m^2 \ll 1$), the equation simplifies to a 2D Fourier transform:

$$V(u,v) \approx \iint I(l,m)\, e^{-2\pi i(ul+vm)}\, dl\, dm$$

- Hence, to obtain the sky brightness $I(l, m)$, we compute:

$$I(l,m) = \iint V(u,v)\, e^{+2\pi i(ul+vm)}\, du\, dv$$

- This inversion is only accurate if the visibility is fully sampled in the (u, v)-plane.

6.2 Gridding

Computational Complexity

- $\mathbf{M}$ = Number of visibilities, $\mathbf{N} \times \mathbf{N}$ = image size.
- **Direct Fourier Transform:** $\mathcal{O}(MN^2)$
- **Discrete Fourier Transform (DFT):** $\mathcal{O}(N^4)$
- **Fast Fourier Transform (FFT):** $\mathcal{O}(N^2 \log_2 N)$

Image Grid Setup

– Predict angular resolution θ_{syn} from:

$$\theta_{\text{syn}} \approx \frac{\lambda}{B_{\max}}$$

– Choose image pixel size $\Delta\theta$ such that:

$$\Delta\theta \approx \frac{\theta_{\text{syn}}}{4 \text{ to } 5}$$

– Choose image size $M \times N$ (in pixels) for desired field of view:

$$\text{FOV} = M \cdot \Delta\theta \times N \cdot \Delta\theta$$

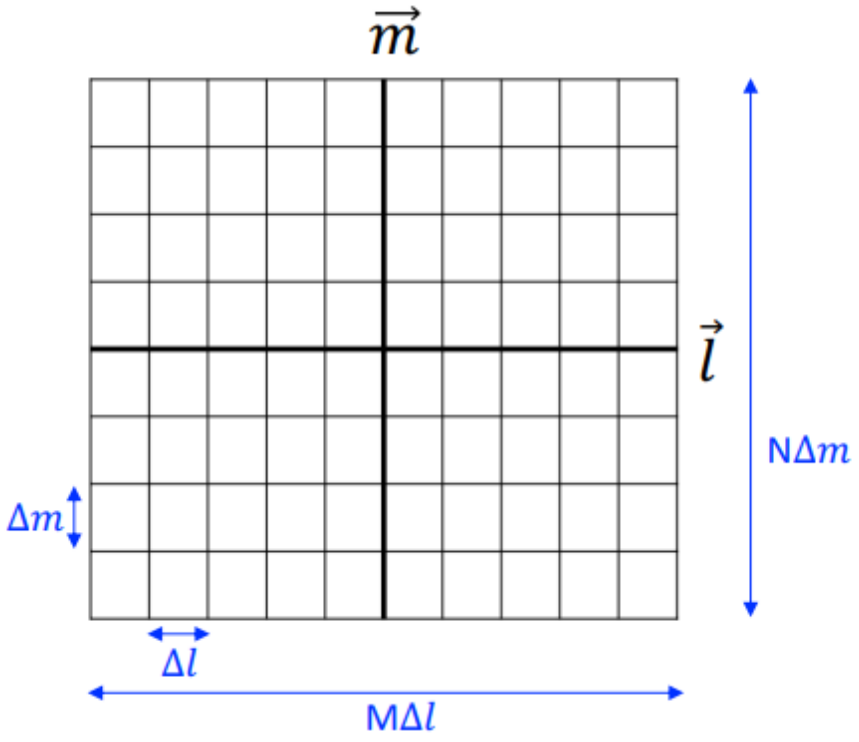


Figure 49

– UV cell size is inversely related to image size:

$$\Delta u = \frac{1}{M \cdot \Delta\theta}, \quad \Delta v = \frac{1}{N \cdot \Delta\theta}$$

– UV grid will also have $M \times N$ cells.

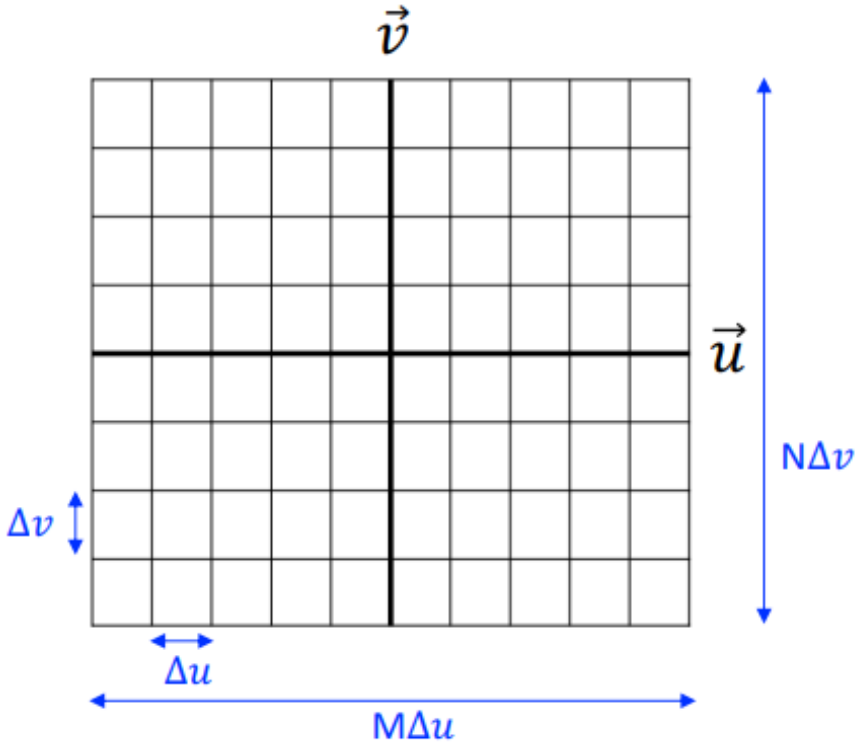


Figure 50

1. **Nearest Neighbor:**

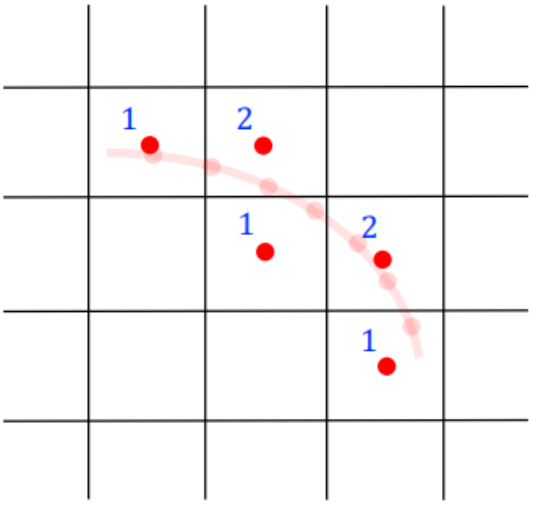


Figure 51

- Assign each visibility to the nearest UV cell.
- If multiple visibilities fall into one cell, compute the weighted average.
- Maintain a separate **gridded weight map**.

2. Convolutional Resampling:

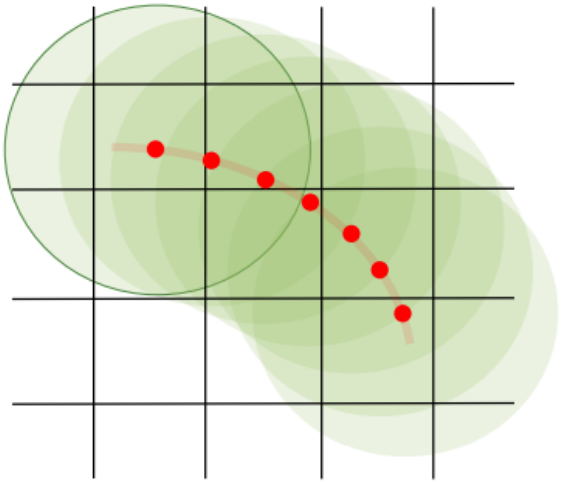


Figure 52

- Use a convolution kernel (e.g., prolate spheroidal function) to distribute each visibility over multiple UV cells.
- Oversample the kernel to minimize interpolation errors.

Incomplete Sampling: The Sampling Function

- Due to incomplete UV coverage, the measured visibility is:

$$V_{\text{meas}}(u, v) = S(u, v) \cdot V(u, v)$$

  where $S(u, v)$ is the **sampling function**, defined as:

$$S(u, v) = \sum_i \delta(u - u_i, v - v_i)$$

- $S(u, v)$ is 1 at sampled locations, 0 elsewhere.

Dirty Image and Deconvolution

- The **inverse Fourier transform** of $V_{\text{meas}}(u, v)$ gives the **dirty image** $I_D(l, m)$:

$$I_D(l, m) = \mathcal{F}^{-1}\left[S(u, v) \cdot V(u, v)\right]$$

- This becomes:

$$I_D(l, m) = I(l, m) * B_D(l, m)$$

  where:

  * $B_D(l, m) = \mathcal{F}^{-1}[S(u, v)]$ is the **dirty beam** or **PSF**.
  * $*$ denotes convolution.

- **Deconvolution** aims to recover $I(l, m)$ from I_D by removing PSF effects (e.g., CLEAN algorithm).

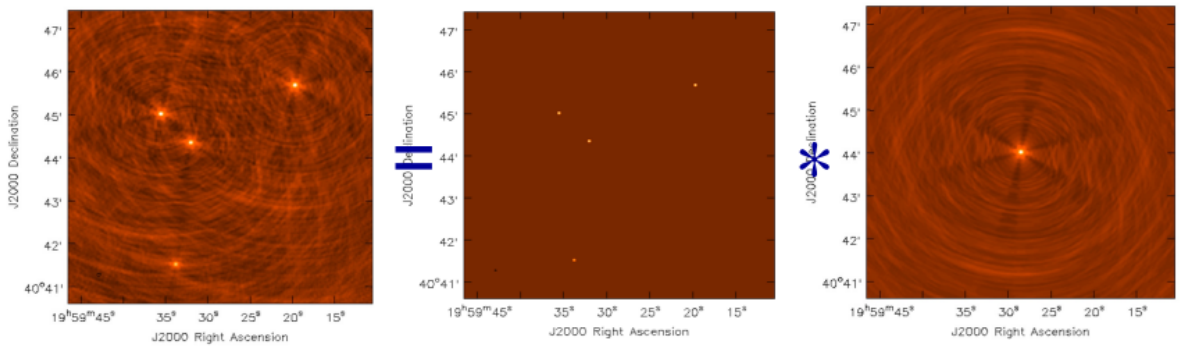


Figure 53

6.3 PSF

The PSF is computed from the iFFT of the gridded weights. We define the 'clean beam' as a Gaussian fit to the central peak of the PSF. Properties of the PSF can be adjusted through manipulation of these weights, at the expense of image sensitivity: **natural weighting**: no adjustments; optimal averaging **uniform weighting**: constant weight per cell **super-uniform weighting**: constant weight density **robust (Briggs) weighting**: in between uniform and natural **(outer) UV taper**: weight decreases with increasing UV-distance

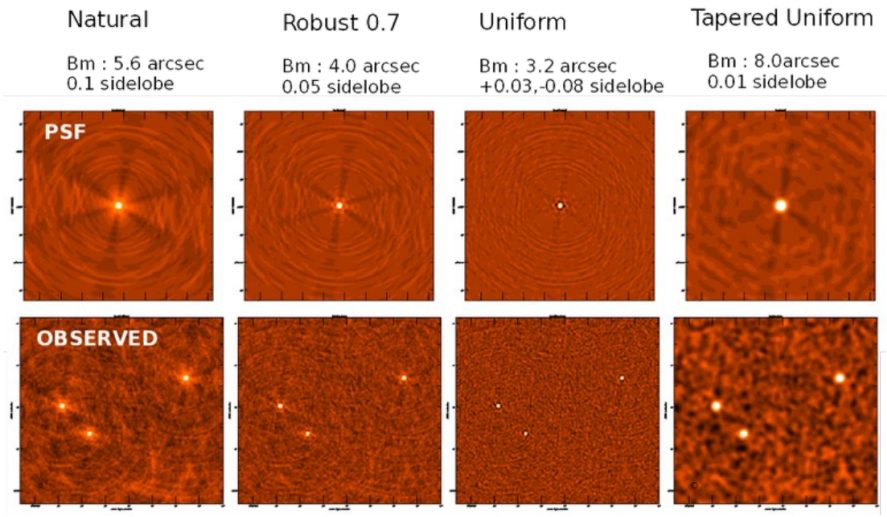


Figure 54

7 Journey Till Now

7.1 Completed Objectives

- Gained theoretical understanding of fundamental radio astronomy concepts such as units (Jy), emission mechanisms (synchrotron, thermal, line emission), and the structure of radio telescopes.
- Explored principles of radio interferometry, including visibility functions, fringe formation, and aperture synthesis.
- Studied how antenna and receiver properties affect observational data: beam patterns, mounts, polarization, and surface errors.
- Understood imaging pipeline: sampling, gridding, Fourier inversion, dirty image formation, PSF, and deconvolution techniques.
- Learned calibration techniques and the radio interferometric measurement equation (RIME) with Jones matrices and Stokes parameters.
- Completed literature review and developed structured documentation using LaTeX.

8 Learning Sources

- **NRAO Website and Online Material**: The National Radio Astronomy Observatory's website offers comprehensive tutorials, documentation, and CASA

guides that have been instrumental in understanding both theoretical and practical aspects of radio astronomy.

- **20th Synthesis Imaging Workshop Lectures**: Lecture videos and notes from the 20th Synthesis Imaging Workshop provide detailed explanations on radio interferometry, calibration, imaging, and deconvolution processes.
- **Essentials of Radio Astronomy (ERA) Book by Condon and Ransom**: This textbook serves as a foundational resource, covering core topics such as flux density, radiative processes, interferometric imaging, and the physics behind radio observations.

CASA Imaging

Contents

1 Working with CASA

CASA stands for Common Astronomy Software Applications and is an open-source software package developed primarily for the analysis and reduction of radio astronomy data. It is widely used in the field of radio astronomy, particularly for working with data from large radio telescopes and interferometers like the VLA (Very Large Array), ALMA (Atacama Large Millimeter/submillimeter Array), and others.

1.1 Getting familiar with the dataset

In this report, we're going to use CASA to look at a GMRT dataset : We first proceed with extracting out dataset. We do so by extracting the .ms file from the .tgz archive using the command:

tar -xvf 32_089_9688_GSB.ms.tar

We keep it in the directory we plan to use and then continue to open CASA as follows. We set up CASA from the terminal using the following command in the directory storing our MS dataset:

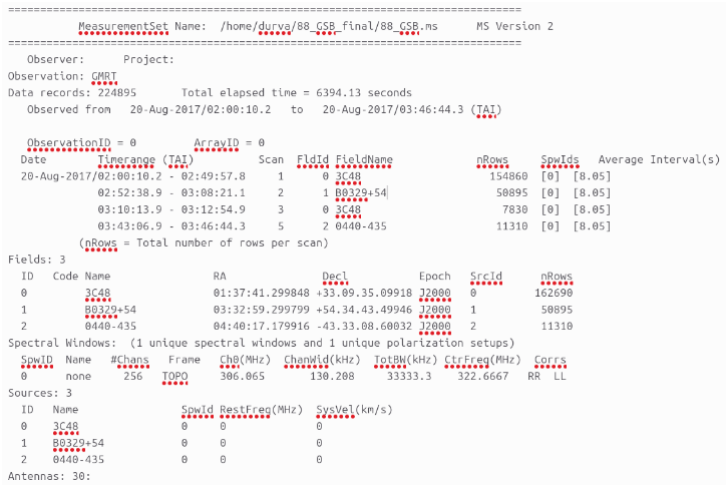
env -u SESSION_MANAGER casa –logfile 88_GSB.log

We go on to analyze the dataset more by looking at its listobs. We do so by using the command:

listobs(vis='32_089_9688_GSB.ms', listfile='88_GSB.listobs')

We get the following list giving us information about the MS dataset.

Total time elapsed: 6394.13 seconds
Total antennas used: 30
Flux calibrator: 3C48
Gain calibrator: 0440-435
Target: B0329+54
We note that there are 4 scans taken: 1,2,3,5. Only one spectral window is present. This C-band instrument configuration uses dual polarization and hence the spw has 256 channels that are 2.0MHz wide.



```
================================================================================
           MeasurementSet Name:  /home/durva/88_GSB_final/88_GSB.ms      MS Version 2
================================================================================
   Observer:      Project:
Observation: GMRT
Data records: 224895       Total elapsed time = 6394.13 seconds
   Observed from   20-Aug-2017/02:00:10.2   to   20-Aug-2017/03:46:44.3 (TAI)

   ObservationID = 0         ArrayID = 0
  Date        Timerange (TAI)          Scan  FldId FieldName             nRows     SpwIds   Average Interval(s)
  20-Aug-2017/02:00:10.2 - 02:49:57.8     1      0 3C48                  154860  [0]  [8.05]
              02:52:38.9 - 03:08:21.1     2      1 B0329+54               50895  [0]  [8.05]
              03:10:13.9 - 03:12:54.9     3      0 3C48                    7830  [0]  [8.05]
              03:43:06.9 - 03:46:44.3     5      2 0440-435               11310  [0]  [8.05]
          (nRows = Total number of rows per scan)
Fields: 3
  ID   Code Name                RA               Decl           Epoch   SrcId      nRows
  0         3C48                01:37:41.299848 +33.09.35.09918 J2000   0         162690
  1         B0329+54            03:32:59.299799 +54.34.43.49946 J2000   1          50895
  2         0440-435            04:40:17.179916 -43.33.08.60032 J2000   2          11310
Spectral Windows:  (1 unique spectral windows and 1 unique polarization setups)
  SpwID  Name   #Chans   Frame   Ch0(MHz)  ChanWid(kHz)  TotBW(kHz) CtrFreq(MHz)  Corrs
  0      none     256   TOPO     306.065        130.208     33333.3    322.6667   RR  LL
Sources: 3
  ID   Name             SpwId RestFreq(MHz)  SysVel(km/s)
  0    3C48             0     0              0
  1    B0329+54         0     0              0
  2    0440-435         0     0              0
Antennas: 30:
```

Figure 1: List of observations of the dataset 32_089_9688_GSB.ms

Now that we are familiar with the dataset, we go on to look at the antenna configuration of this antenna array. To do this, we use the following command:

plotants(vis='32_089_9688_GSB.ms', figfile='88_GSB.png')

We get the following antenna configuration:

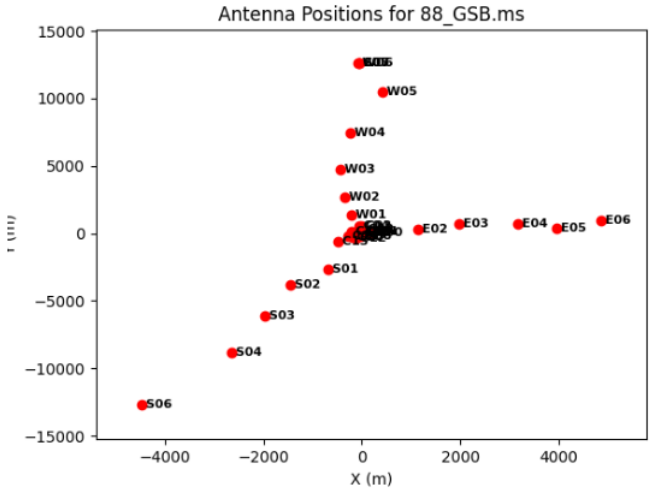


Figure 2: Antenna configuration of GMRT antennas

We observe that antennas 'W06', 'S06' and 'E06' are on the extreme ends of the array. Now we move on to analyzing the dataset using different plots.

1.2 Understanding the data

Now, we move on to analyze the data of this MS using different axes. We follow the command 'plotms' to open up the GUI interface of the CASA plotting tool which allows

us to plot visibility data from a Measurement Set to help inspect and analyze radio interferometric observations.
plotms is a tool in CASA that allows you to visualize various aspects of your radio interferometric data, typically in the form of visibility plots. This tool is widely used to inspect and analyze data before and after calibration, helping you detect issues such as noisy channels, bad data, or calibration problems [**plotms**].
We select the MS file in the 'Data' section of our 'Plot' tab. We set the axes from the 'Axes' section, iteration axis from the 'Page' section and we can colorize our plot by checking the 'colorize' option in the 'Display' section and select a medium to colorize it with.
We now go on and plot various different graphs and draw inferations.

1. Amplitude vs UVwave
   Firtly, we plot Amplitude vs UVwave for different fields or calibrators. The amplitude vs. uv-field plot shows how the signal amplitude varies across different baselines or points in the uv-plane. This is important because it gives you a sense of the coverage and sampling of the uv-plane by the interferometric array.

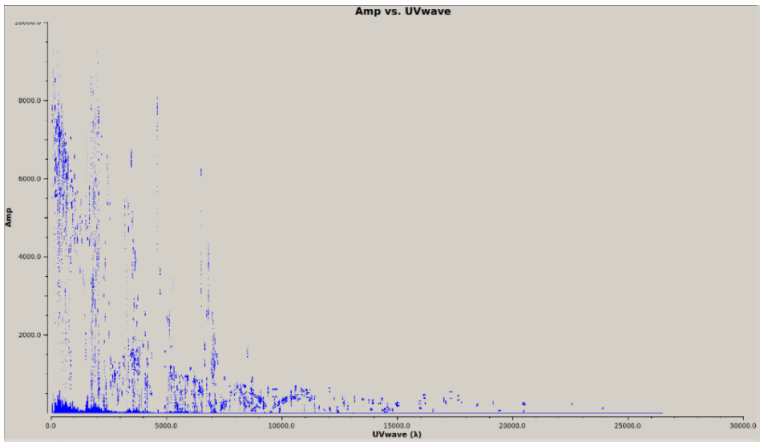


Figure 3: Graph for Amplitude vs UVwave for the flux calibrator

We notice that for field 2, the amplitude drops sharply after a distance. This can be a very big resolved source because the Fourier transform of constant function is a delta function. As we show earlier, big sources get "spread out" in the Fourier domain, leading to a falloff in the measured amplitude at larger baselines, unlike point sources that give a sharp spike.

But plotting the same graph for the gain calibrator gives us a different result.

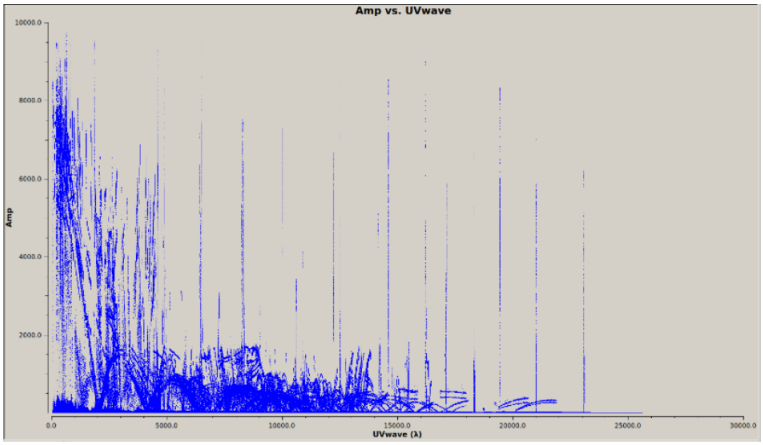


Figure 4: Graph for Amplitude vs UVwave for the gain calibrator

In the case of Field 0 , we can observe for a small starting scale of UV the amplitude is consistent i.e. more about a constant as compared to Field 2. This source is somewhat less resolved and hence, kind of a point source. We can also plot it for the science target.

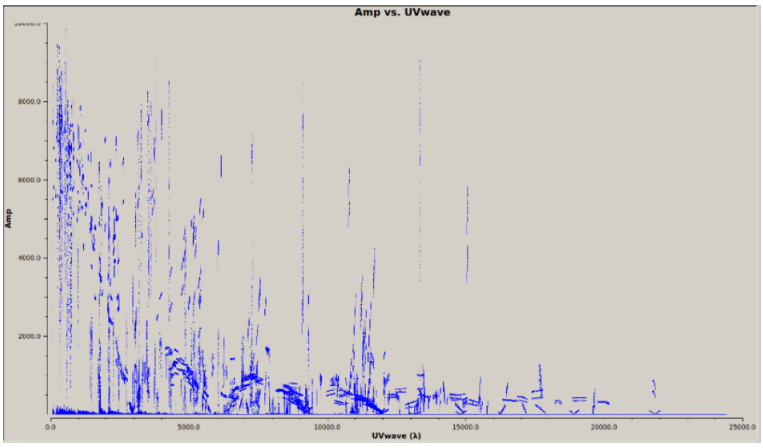


Figure 5: Graph for Amplitude vs UVwave for the science target

Since this is pre-calibration data, the next step should be to perform the gain and flux calibration. This will likely resolve much of the amplitude fluctuation and improve the consistency of your results.

2. Phase vs Channel
   Next, we plot phase vs channel iterated over antennas. The phase of the signal indicates the relative position and alignment of the source. By plotting phase vs. channel, we can check if there is any frequency-dependent phase variation.

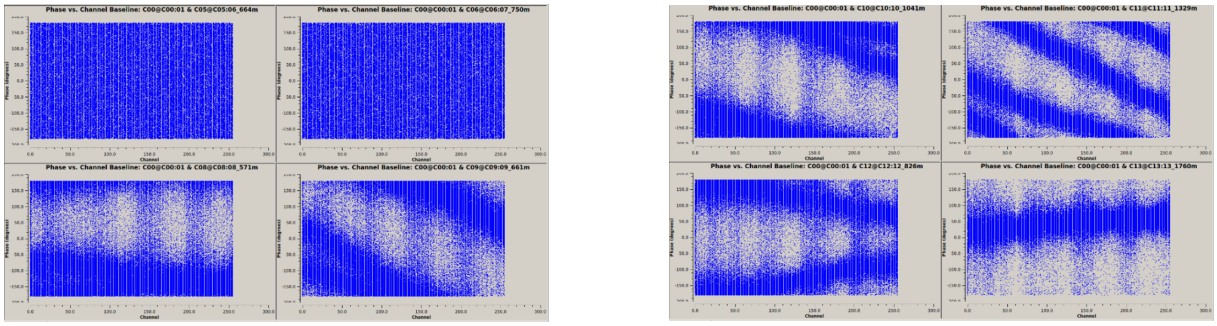


Figure 6: Graph for Phase vs Channel for both the flux and gain calibrators iterated over baselines.

For a particular antenna pair, the plot is completely filled or maybe has a low slope value. This can be interpreted as two ways: that pair may be coherent or have a short baseline. This is because the signal reaches almost at same time. Also, while here we can see there are sloppy lines, they clearly represent delay in that antenna pair.

We now look at the antenna positions zoomed at around the central antennas.

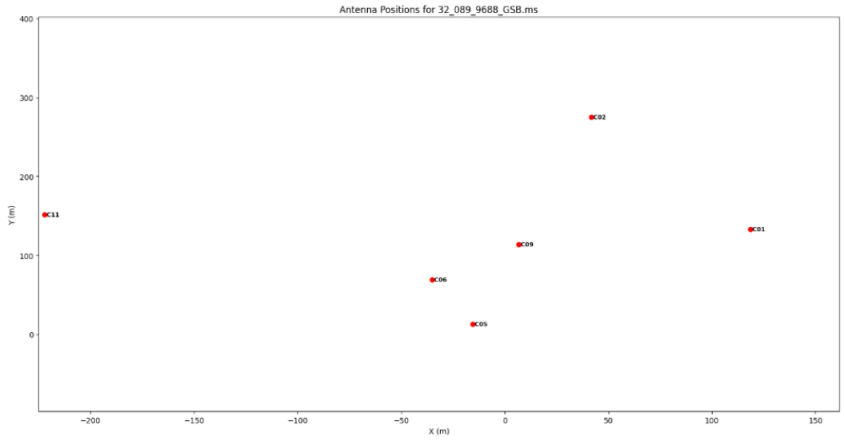


Figure 7: Antenna configuration zoomed at the central part

Now delay can occur due to many reasons signal interuption midway, length of baseline etc. but what we can do is compare. We can see the baseline of C01 & C11(tilted plot showing delay) is greater than C01 & C05(consistent value plot). Hence this can be also one of the reasons of delay.

For the remaining part of this report, we will be looking at the dataset "32_089_9622_GSB.ms". We import it using the same mechanism and observe and analyze the data as given.
We make a note about the following data for the 22_GSB dataset:

Total time elapsed: 5282.81 seconds
Total antennas used: 30

Flux calibrator: 3C48
Gain calibrator: 0837-198
Bandpass calibrator: 0440-435
Target: B0329+54
We note that there are 5 scans taken: 1,2,3,4,6. Only one spectral window is present. This C-band instrument configuration uses dual polarization and hence the spw has 256 channels that are 2.0MHz wide.
We save it as 22_GSB dataset. Let us analyze the data for this dataset. We now go on and plot various different graphs:

1. Ampltiude vs Time
   Plotting Amplitude vs Time over the global y axis iterated over each antenna and operated for field 1 (flux calibrator) first.

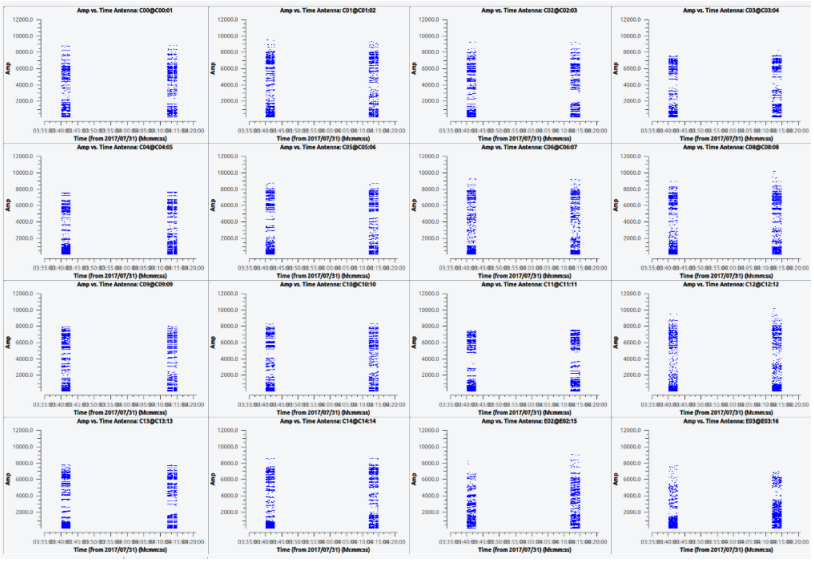


Figure 8: Graph for Amplitude vs Time for the flux calibrator

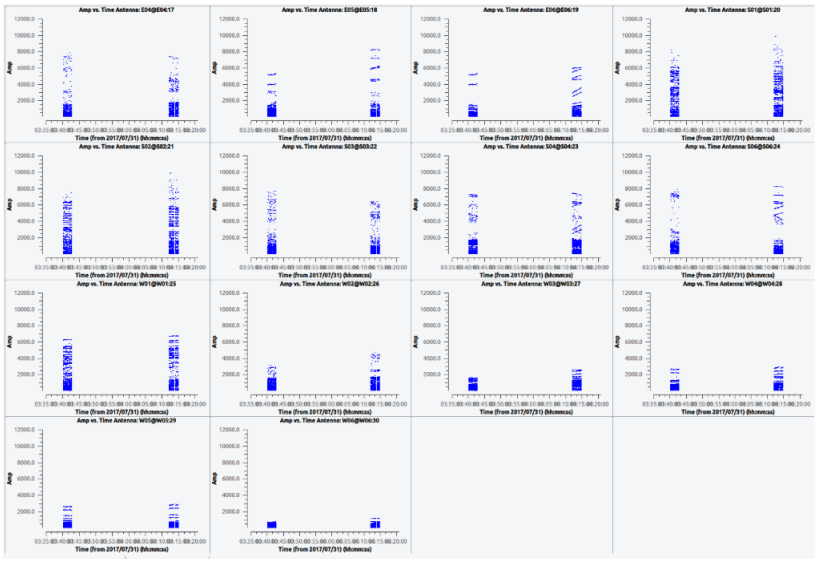


Figure 9: Graph for Amplitude vs Time for the flux calibrator

Now, plotting the same for field 0 which is the phase calibrator.

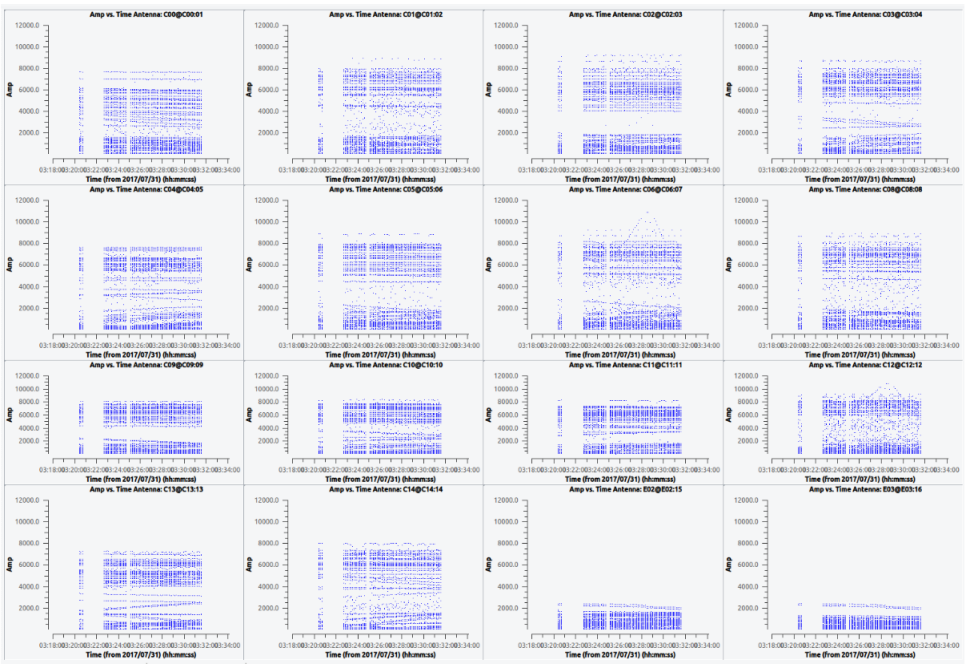


Figure 10: Graph for Amplitude vs Time for the phase calibrator

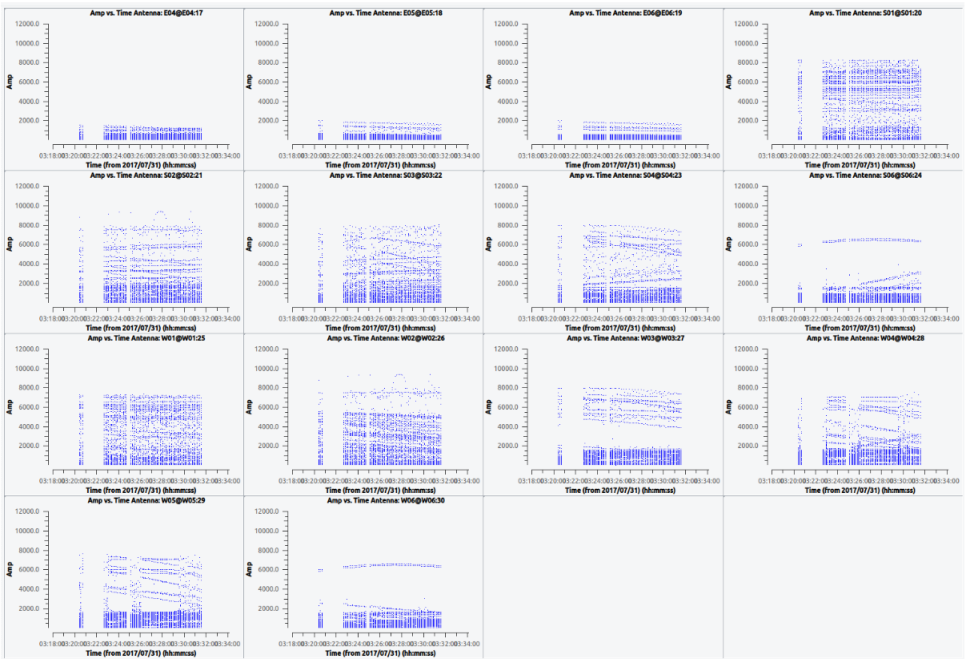


Figure 11: Graph for Amplitude vs Time for the phase calibrator

We notice that antennas E02, E03, E05, E04, W06 and S06 seem to have low amplitudes as compared to the other antennas. We will look into these later.

2. Amplitude vs Channel
We now plot amplitude vs channel graphs for the data of field 1 iterated over antennas. We notice that all the plots have really small curves. This is because of the strong signal rising at the start of the plot in every plot. We will later flag these out.

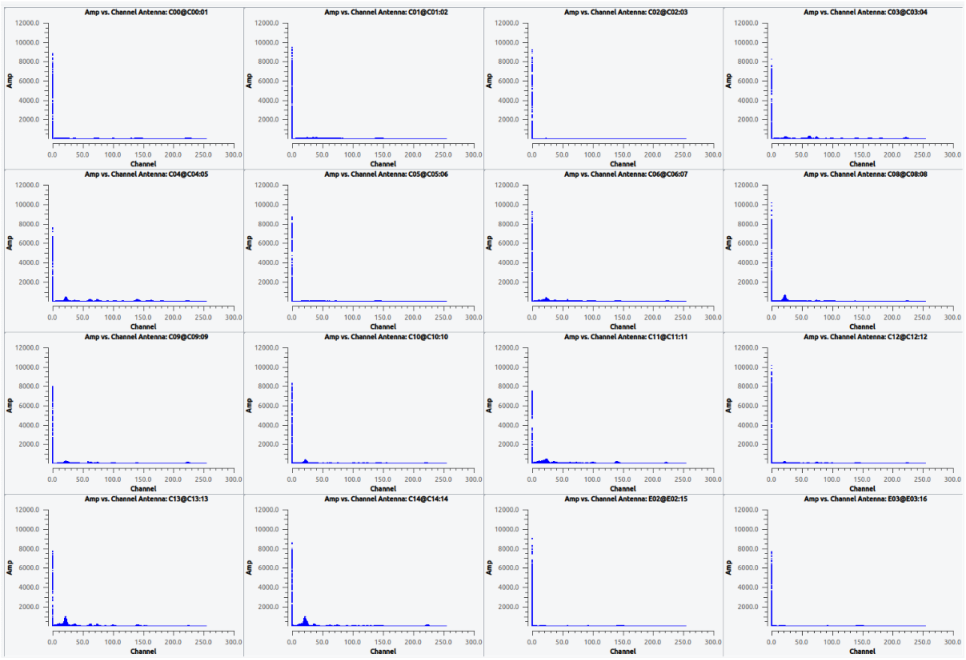


Figure 12: Graph for Amplitude vs Channel for the flux calibrator

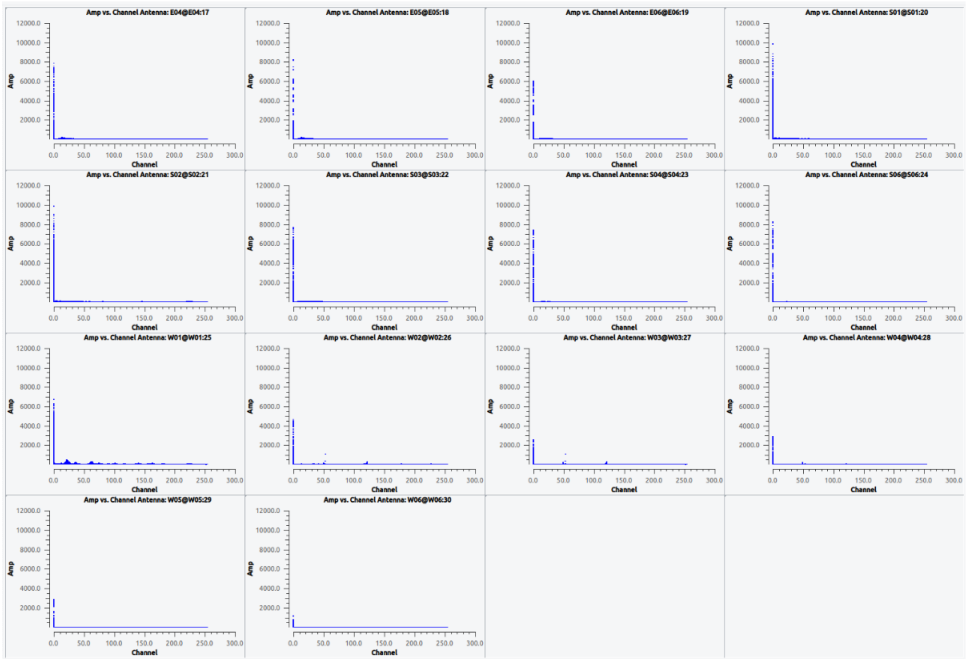


Figure 13: Graph for Amplitude vs Channel for the flux calibrator

3. Phase vs Channel
   Next we go on to plot phase vs channel iterated over baselines over the global y axis for field 1.

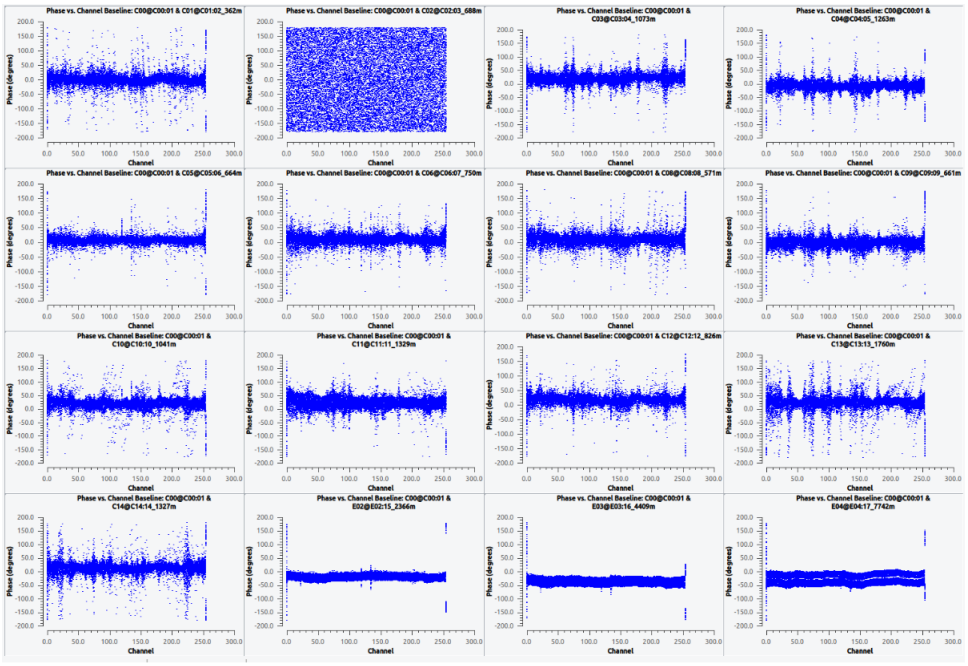


Figure 14: Graph for Phase vs Channel for the flux calibrator

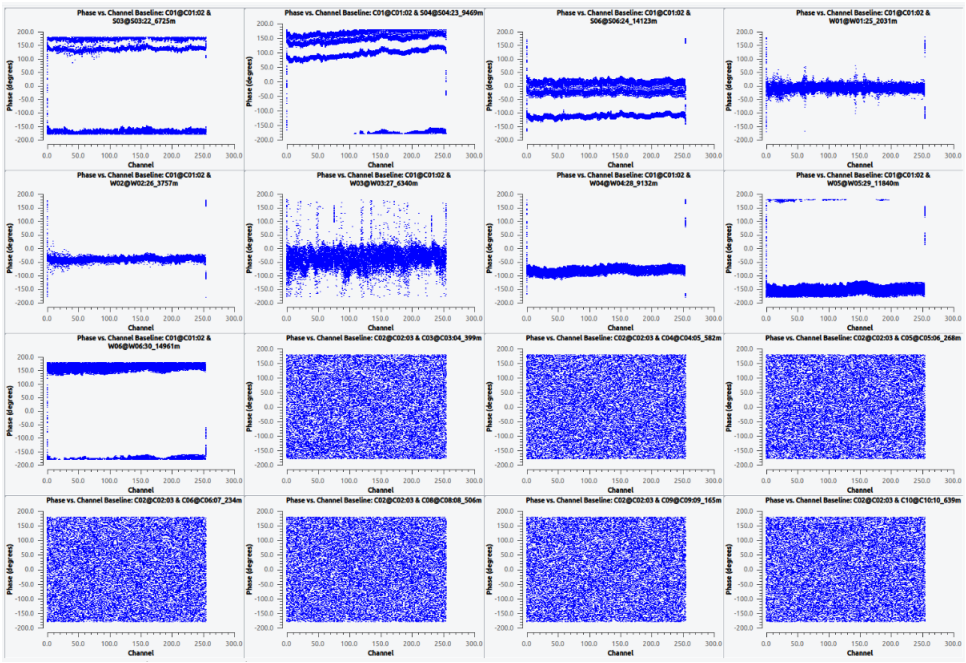


Figure 15: Graph for Phase vs Channel for the flux calibrator

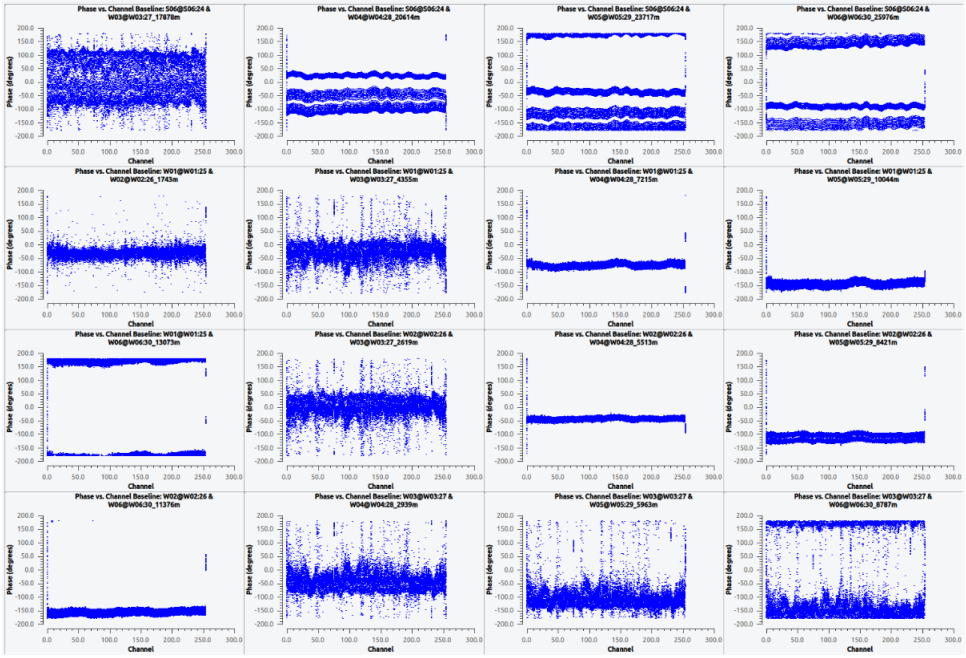


Figure 16: Graph for Phase vs Channel for the flux calibrator

We can clearly conclude from these plots that the C02 antenna is flawed and needs to be flagged. We also note the strong RFI effects from the antenna W03. We go on to look at some other plots to confirm this.

4. Phase vs Time
   We now plot phase over time iterated over baselines and colorized by correlations.

We average over a bin of 16 channels and do these plots only for field 1.

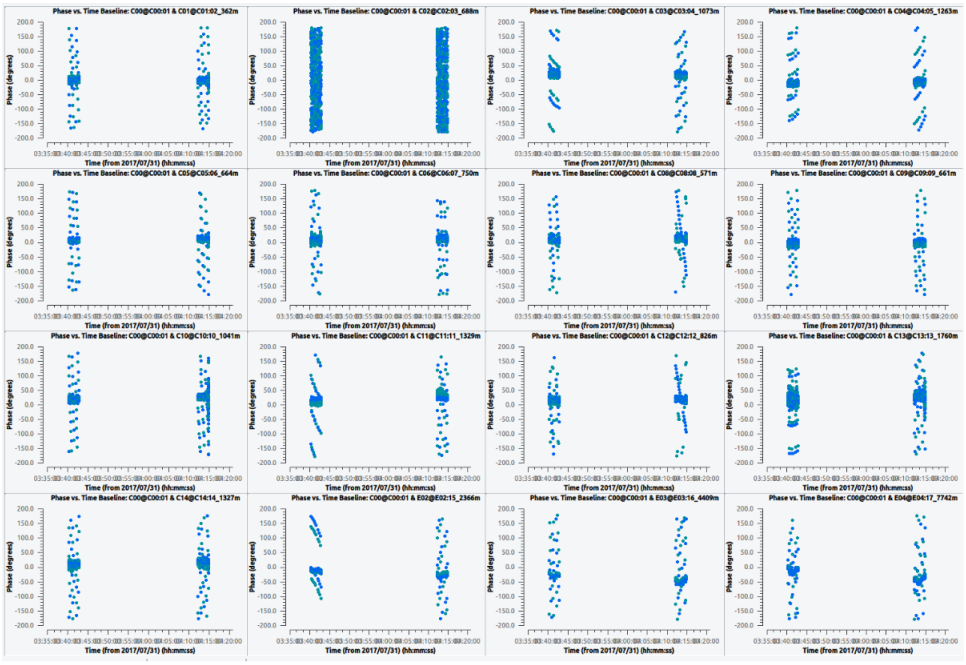


Figure 17: Graph for Phase vs Time for the flux calibrator

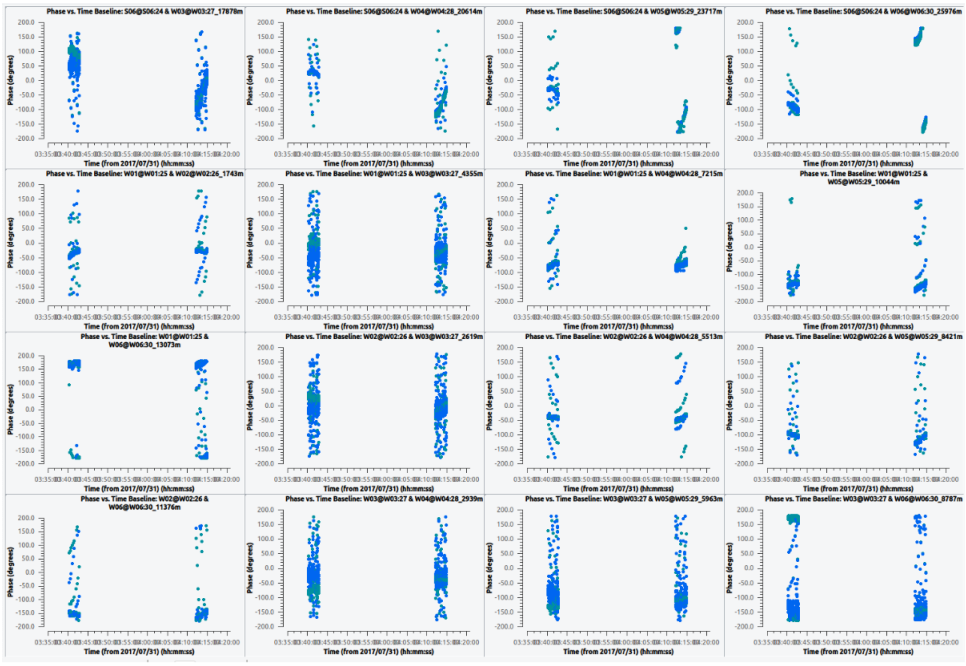


Figure 18: Graph for Phase vs Time for the flux calibrator

We now clearly see that the antenna C02 needs to be removed as it is completely corrupted. Antenna W03 gives us unstable phase change over time and hence should be considered for flagging. We go on to look at one more plot of Antenna vs Time to make sure of this.

5. Antenna vs Time
   Lastly, we look at antenna vs time plots and notice the behavior of the antennas.

Figure 19: Graph for Amplitude vs Time for the flux calibrator

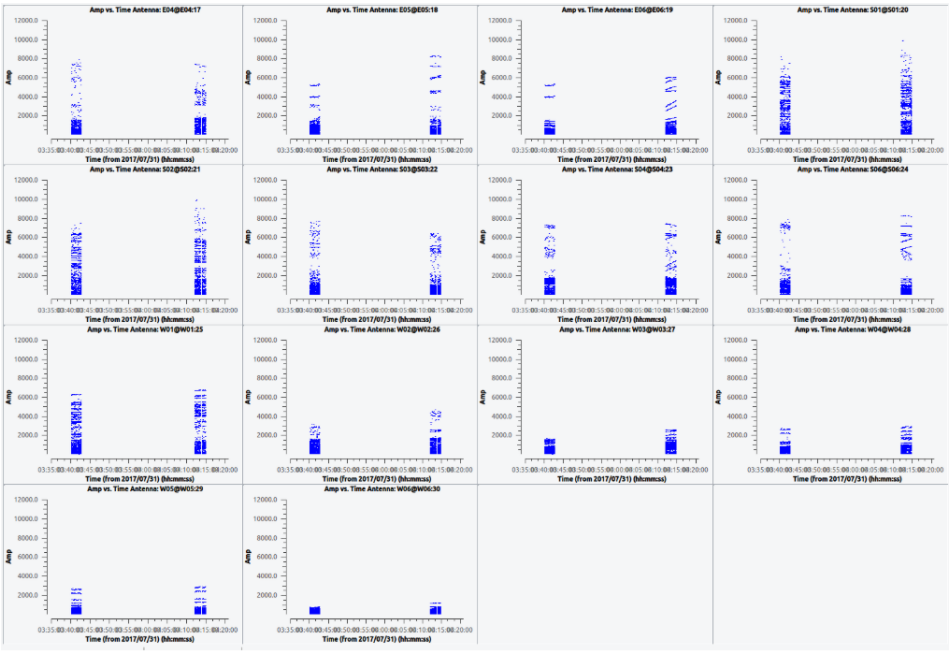


Figure 20: Graph for Amplitude vs Time for the flux calibrator

This looks pretty nice! We know that all our antennas are working, some of them may be corrupted though.

Now, having looked at the behavior of our data, we go on to manually flag the data which behaves badly.

1.3 Flagging

1.3.1 Manual mode

Before moving further, we flag out the bad channels and RFI infected area of our data by careful observation of the dataset. It is always recommended to first safely back up your flags. In this way, if you ever need to undo all the flags we can do so and get the raw data back. This is especially useful for tfcrop and quacking. We use CASA's flagmanager tool. We use the following command before initiating any flagging:
flagmanager(vis='22_GSB.ms', mode='save', versionname='pre_flagging_backup')

1. Bad channel
   Now, we first flag out the first spectral window. We do this as a common practice so as to avoid auto-correlator edge artifacts, poor gain solutions, incorrect fits, or division by zero errors, especially if the data in that channel is zero or unstable. In this case, we clearly saw there is single thin vertical line for channel 0 consistent for all antennas and fields. We decide to flag it using the following command:

```
CASA <7>: flagdata(
...:      vis='32_089_9622_GSB.ms',
...:      mode='manual',
...:      spw='0:0',
...:      savepars=True,
...:      cmdreason='badchan'
...: )
```

   The following commands explain the various meaning of different keywords.

   - `vis`: Name of the input measurement set to be flagged.
   - `mode`: Sets the flagging mode (e.g., `'manual'` , `'clip'`, `'rflag'`, etc.).
   - `spw`: Selects spectral windows and channels to flag (e.g., `'0:0'` means channel 0 of spw 0).
   - `savepars`: If `True`, saves the flagging parameters to an internal file for future reference.
   - `cmdreason`: A user-defined reason or tag for the flagging command (e.g., `'badchan'`).

   We now look again at Amplitude vs Channel for field 1 and observe the plot. This is iterated over antennas.

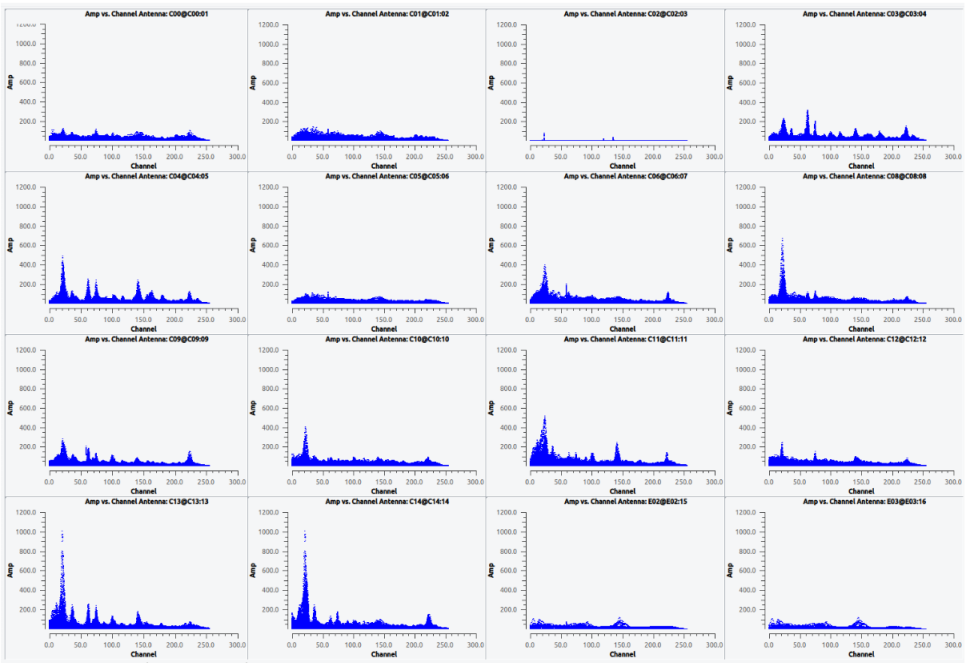


Figure 21: Amplitude vs Channel for the flux calibrator

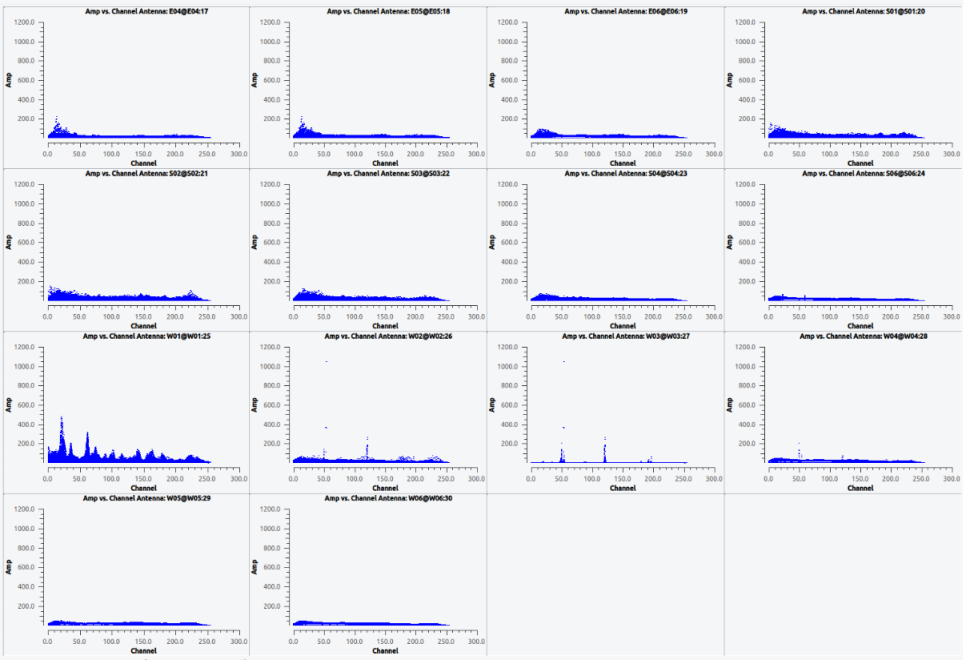


Figure 22: Amplitude vs Channel for the flux calibrator

Now the plot looks much cleaner. We still make note of the extremely low amplitude of Antenna C02 and W03.

2. Quacking
   It is common for the array to require a small amount of time to settle down at the start of a scan. When a scan starts, antennas may still be slewing to the new source

or locking onto the correct position which might cause these few seconds to contain some unreliable phase, amplitude, zeroed or spiky data.
Consequently, it has become standard practice to flag the initial samples from the start of each scan. This is known as 'quack' flagging. We flag the first 10 seconds and ending 10 seconds of our dataset using the following commands:

```
CASA<9>:flagdata(vis='22_GSB.ms',
...:             mode='quack',
...:             quackinterval=10.0,
...:             quackmode='beg',
...:             savepars=True,
...:             cmdreason='quack-start')
```

```
CASA <9>: flagdata(vis='22_GSB.ms',
...:             mode='quack',
...:             quackinterval=10.0,
...:             quackmode='endb',
...:             savepars=True,
...:             cmdreason='quack-end')
```

Let us look at the terms used here:

- `mode`: Set to `'quack'` to flag a fixed time interval at the beginning or end of scans.
- `quackinterval`: Time (in seconds) to flag at the beginning or end of each scan.
- `quackmode`: Sets whether to flag the `'beg'` (start) or `'end'` (end) of the data. It offers other options like `'tail'`, `'both'`.
- `savepars`: If `True`, saves the quack flagging parameters internally.

It is always a good practice to use 'savepars = True'. This tells CASA to save the parameters of the flagging command you just ran into a history table inside the .ms file. Basically, it logs these flags for traceability.

3. Bad Antennas
Now, some antennas may not be bad at all times, but if an antennas stops working while on the target source, it can be difficult to find out. Thus we make a decision based on the secondary calibrator scans. Depending on when such antennas stopped working, we can choose to flag them for that duration. If an antenna is acting badly throughout we can choose to flag it out completely. We also check for both the polarization [1].

We plot phase vs channel of different antennas iterated over baselines.

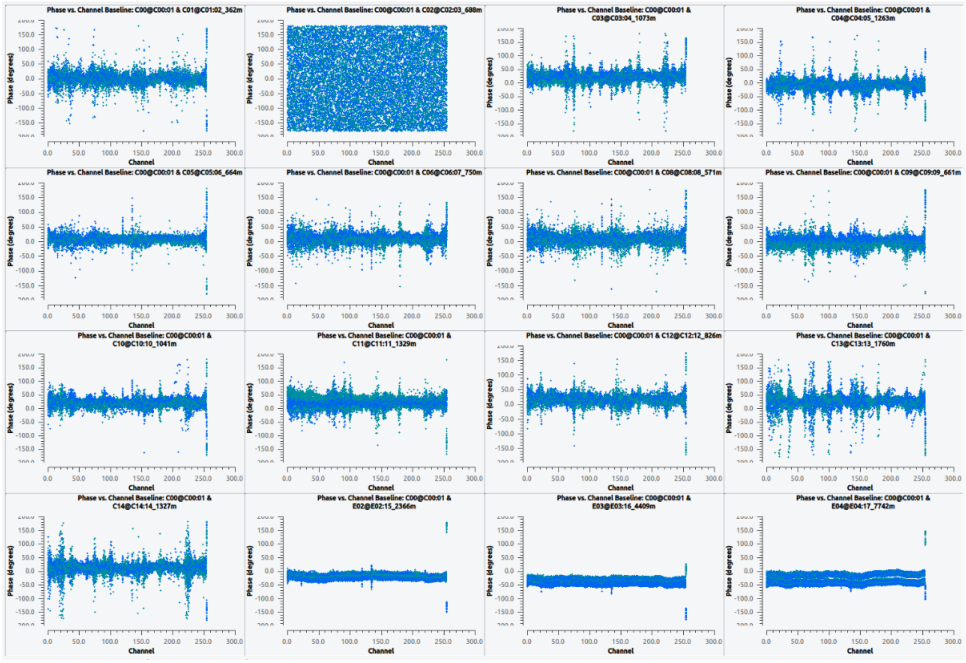


Figure 23: Phase vs Channel for the flux calibrator

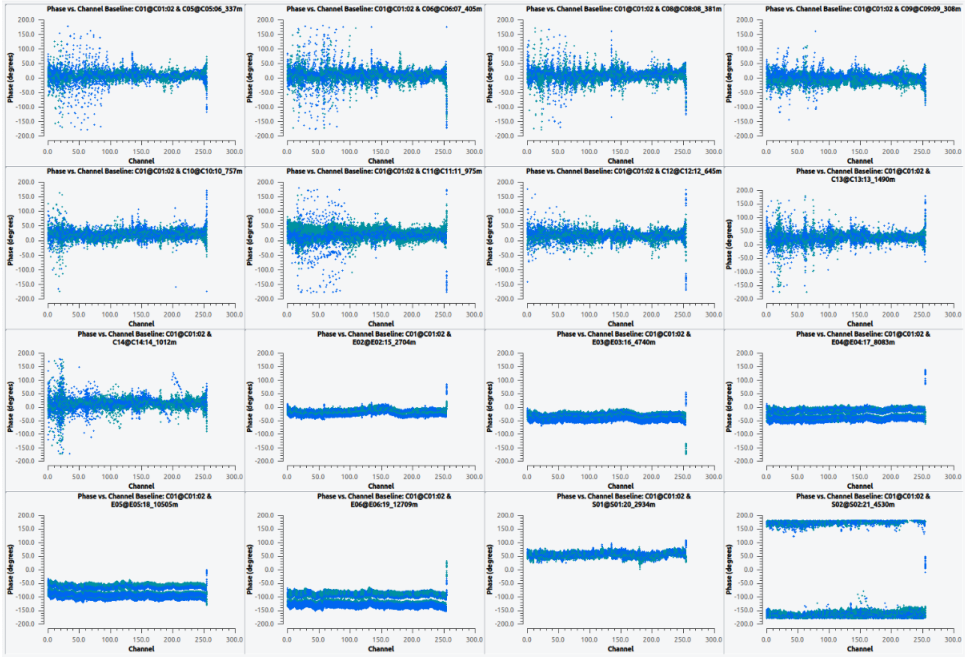


Figure 24: Phase vs Channel for the science target

We notice reasonable but inconsistent RFI over antennas W01, C11, C12, C13 and C14. We observe high and consistent RFI in antenna W03. The antenna C02 is fully corrupted. We go on to flag out antennas W03 and C02 and later use tfcrops to maintain the other RFI. We use the following command:

```
CASA <9>: flagdata(vis='22_GSB.ms',
```

```
...:          mode='manual',
...:          antenna='C02, W03',
...:          savepars=True,
...:          cmdreason='badant')
```

We go on to visualise the data again before applying tfcrops. Since we are done with basic flagging, we plot the graphs to see if there are any other problems. We start off by plotting Amplitude vs Channel colorized over antennas for field 1(flux calibrator). This gives us an idea of the RFI. We get:

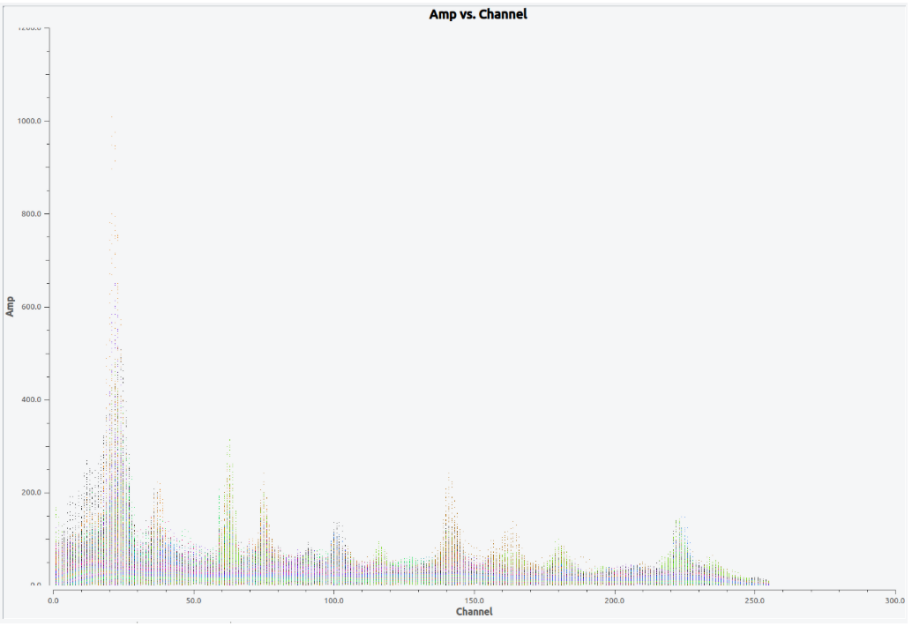


Figure 25: Amplitude vs Channel colorized over antennas for all fields.

We notice that the RFI is strong in the starting of the graph and is reasonably present throughout the entirety of the plot. We take this graph as a reference for all our tfcrop work. We plot Amp vs UVdist colorized over antennas and averaged over 4 seconds and we do this for every field.

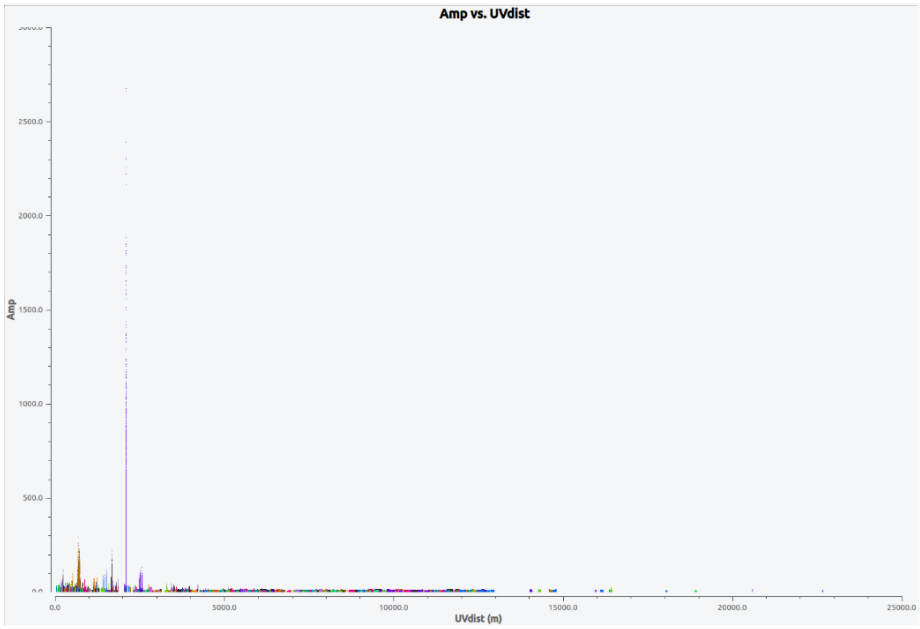


Figure 26: Amplitude vs UVdist for the gain calibrator

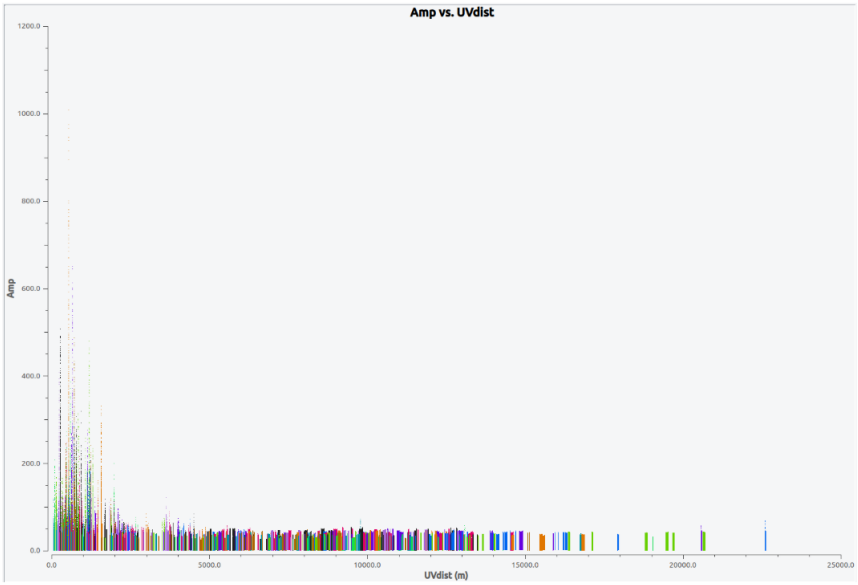


Figure 27: Amplitude vs UVdist for the flux calibrator

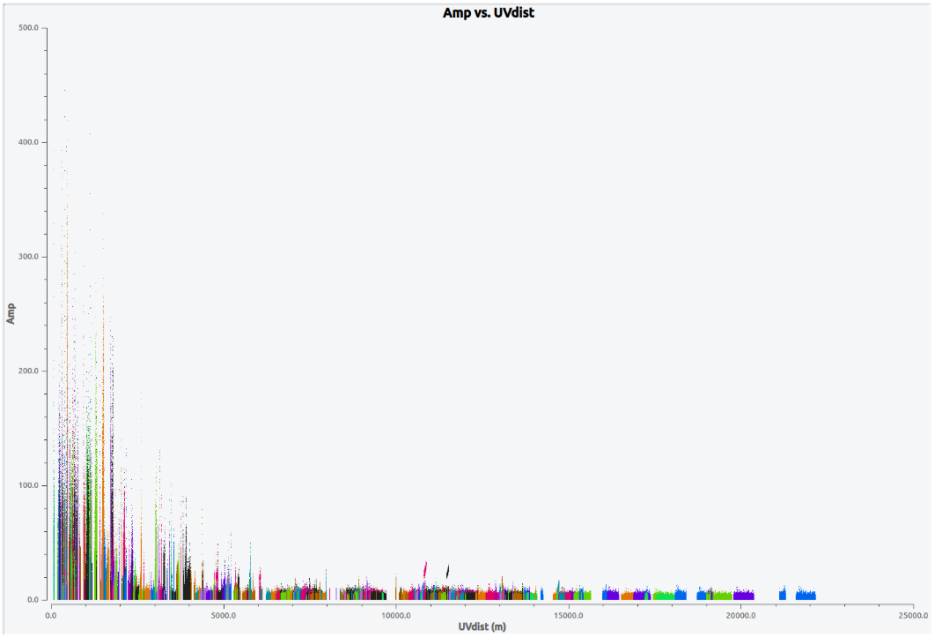


Figure 28: Amplitude vs UVdist for the science target

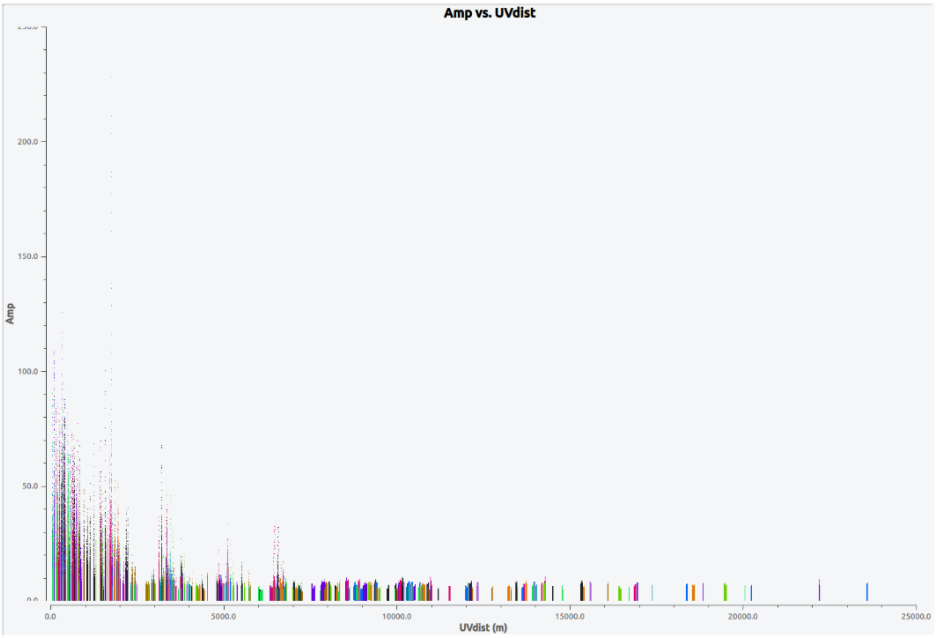


Figure 29: Amplitude vs UVdist for the bandpass calibrator

We notice that all these are extended sources. We also notice a sharp spike in field 1 which might indicate very strong RFI or a faulty antenna. We'll plot Antenna vs Amp to take a look at this.

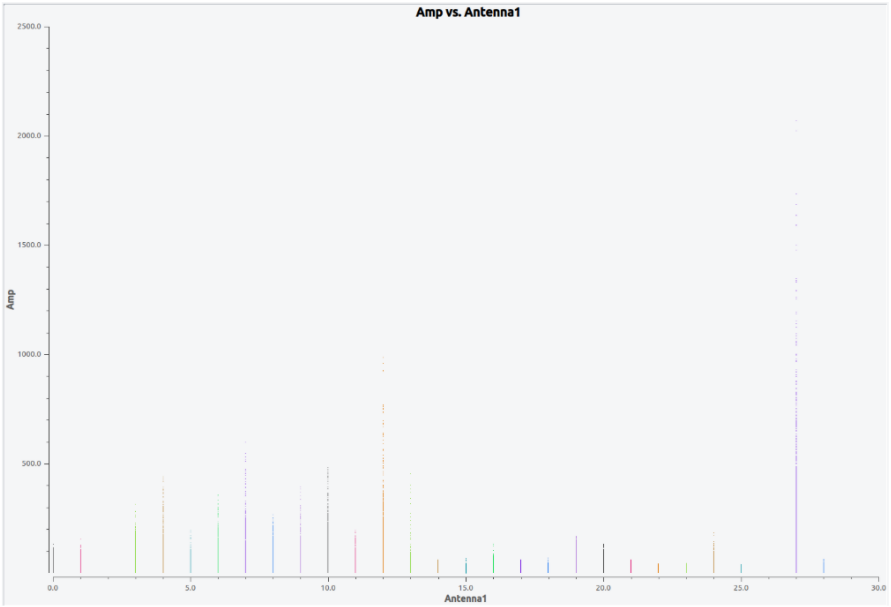


Figure 30: Antenna vs Amplitude for all fields.

We notice that one antenna shows huge RFI spikes, we zoom in and find the antenna W04 to be the primary problem. We will later sort this out using tfcrops. We plot for antenna W04 only for axis amplitude and antenna, no averaging, colorized by antennas.

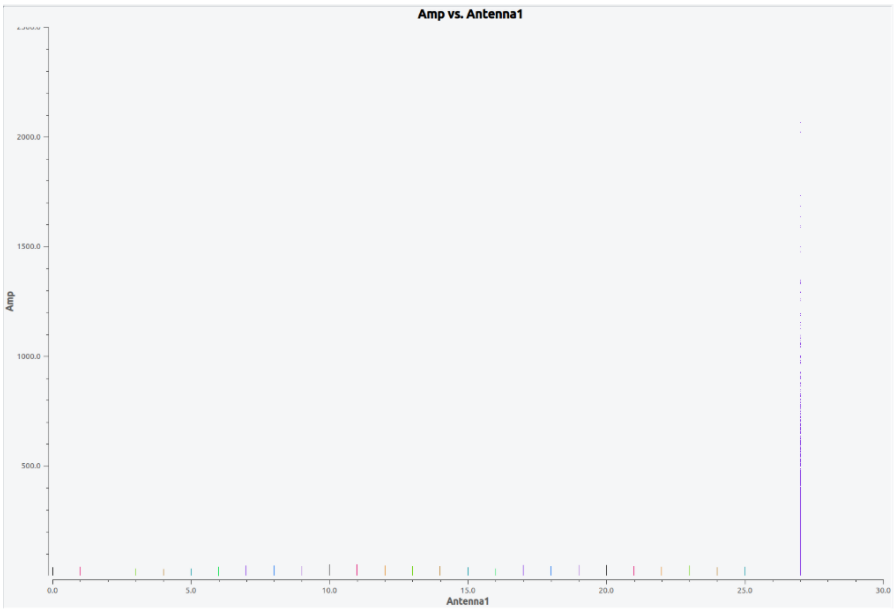


Figure 31: Antenna vs Amplitude for antenna W04.

Now we plot Amplitude vs Time for all fields, no averaging and colorized by fields.

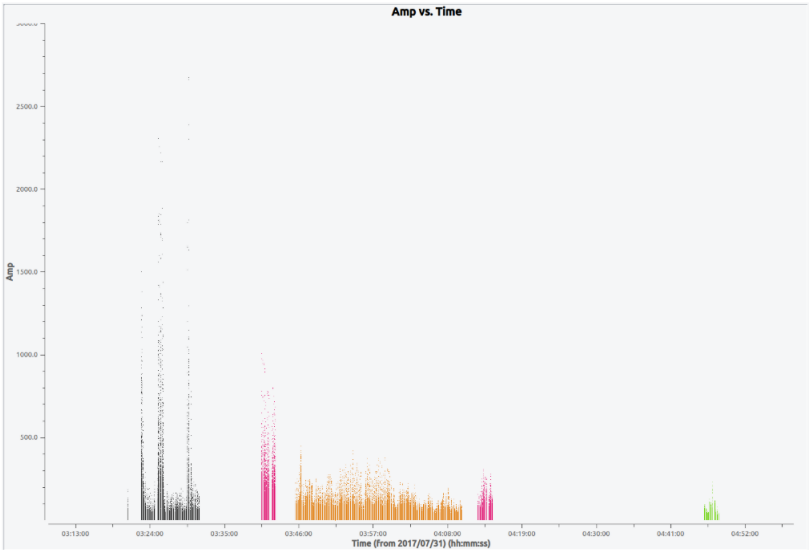


Figure 32: Amplitude vs Time for all fields.

We do the same for field 1.

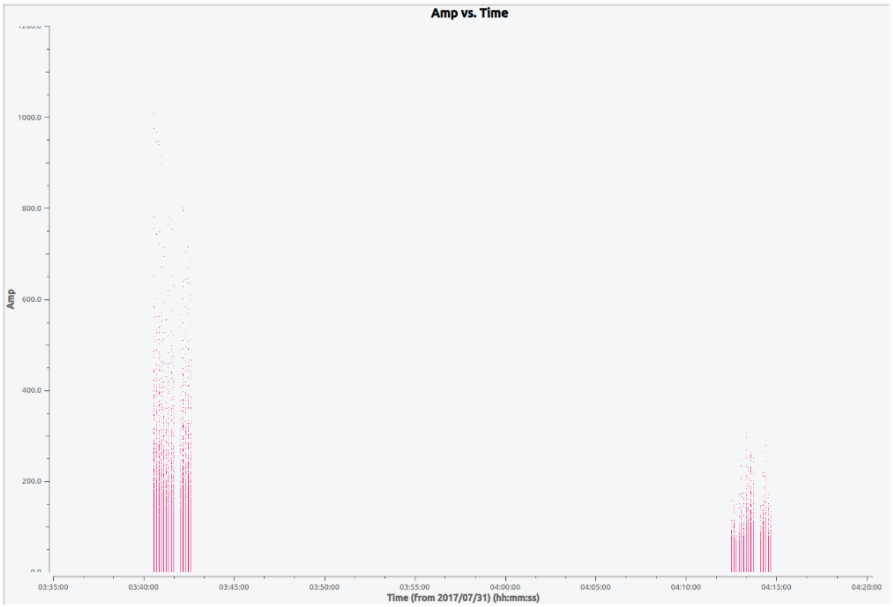


Figure 33: Amplitude vs Time for field 1.

We notice huge RFI spikes in all these graphs. These are bad for further analysis, hence we handle these RFI spikes by using automatic flagging.

1.3.2 TFcrop

Now we're gonna use tfcrop in CASA to flag the RFI data automatically. The tfcrop mode is a time-frequency flagging algorithm used in CASA's flagdata task. It is designed to detect and flag Radio Frequency Interference (RFI) based on statistical outliers in both time and frequency domains. It analyzes visibilities for amplitude outliers in time and

frequency using a sliding window [<**empty citation**>].
In the first step we get an idea about the amount of flagging that will result from our choice of parameters action = 'calculate is the parameter that allows us to see this [<**empty citation**>].
We look at the heatmaps of visibility amplitudes over time and frequency for individual baselines and polarization products but we don't flag anything yet. We use the command:

```
CASA <9>: flagdata(vis='22_GSB.ms',
...:           mode='tfcrop',
...:           spw='',
...:           ntime='scan',
...:           timecutoff=6.0,
...:           freqcutoff=6.0,
...:           extendflags=False,
...:           action='calculate',
...:           display='both')
```

We look at the results for different scans.

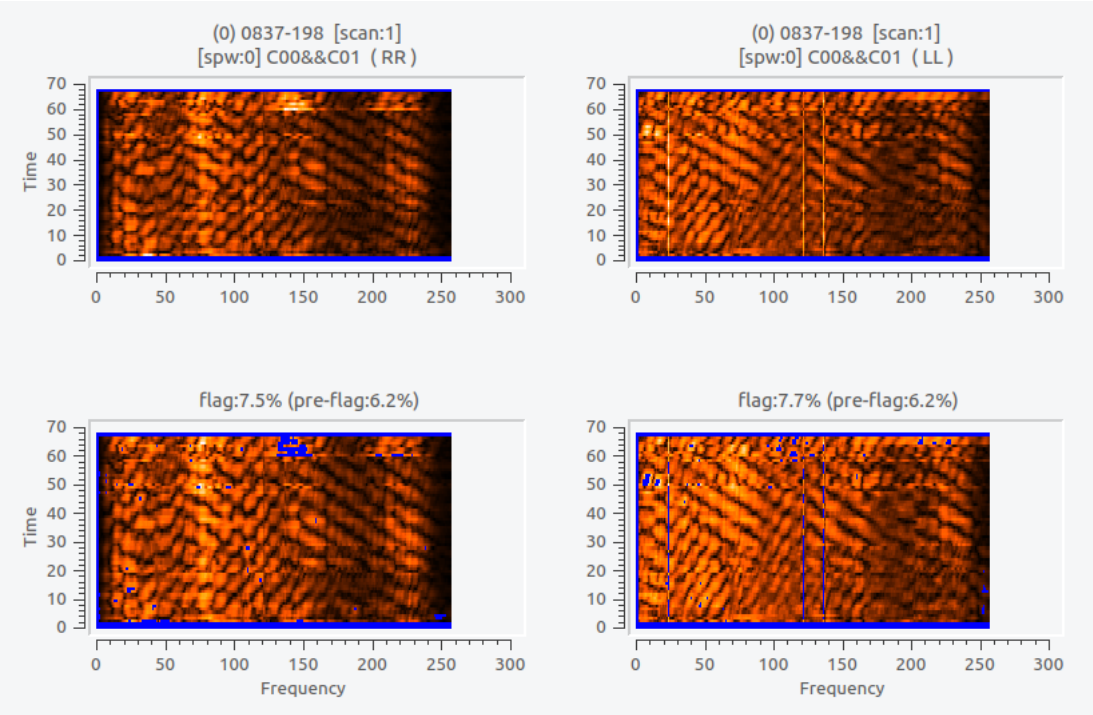


Figure 34: Flagged data for scan 1

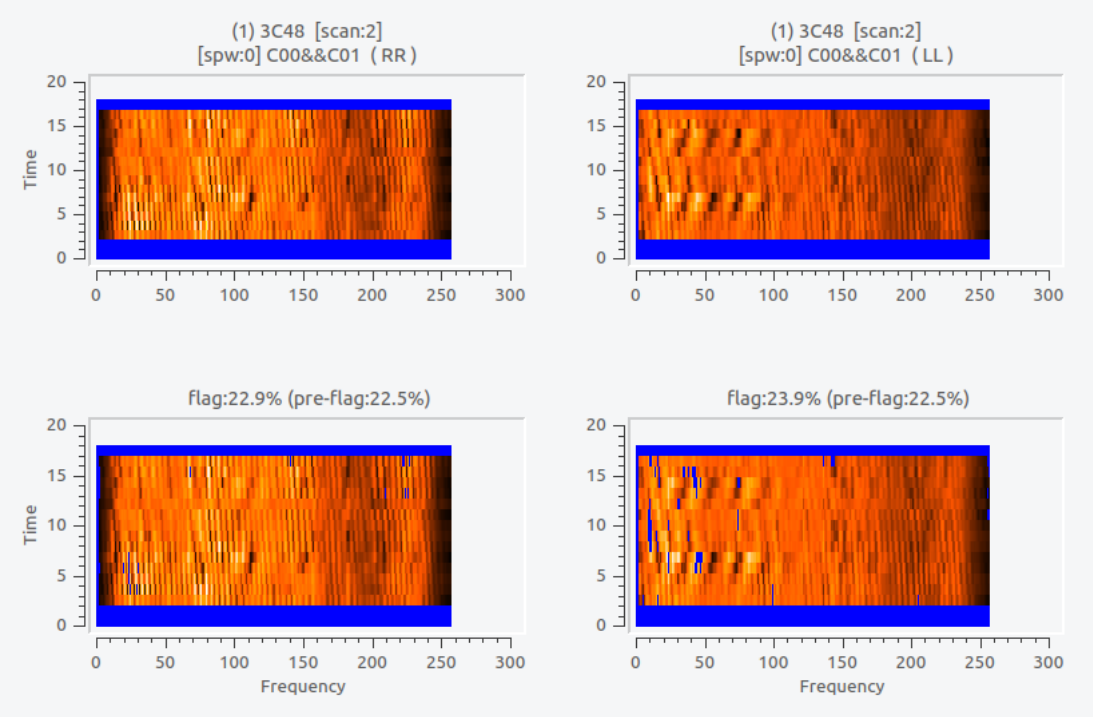


Figure 35: Flagged data for scan 2

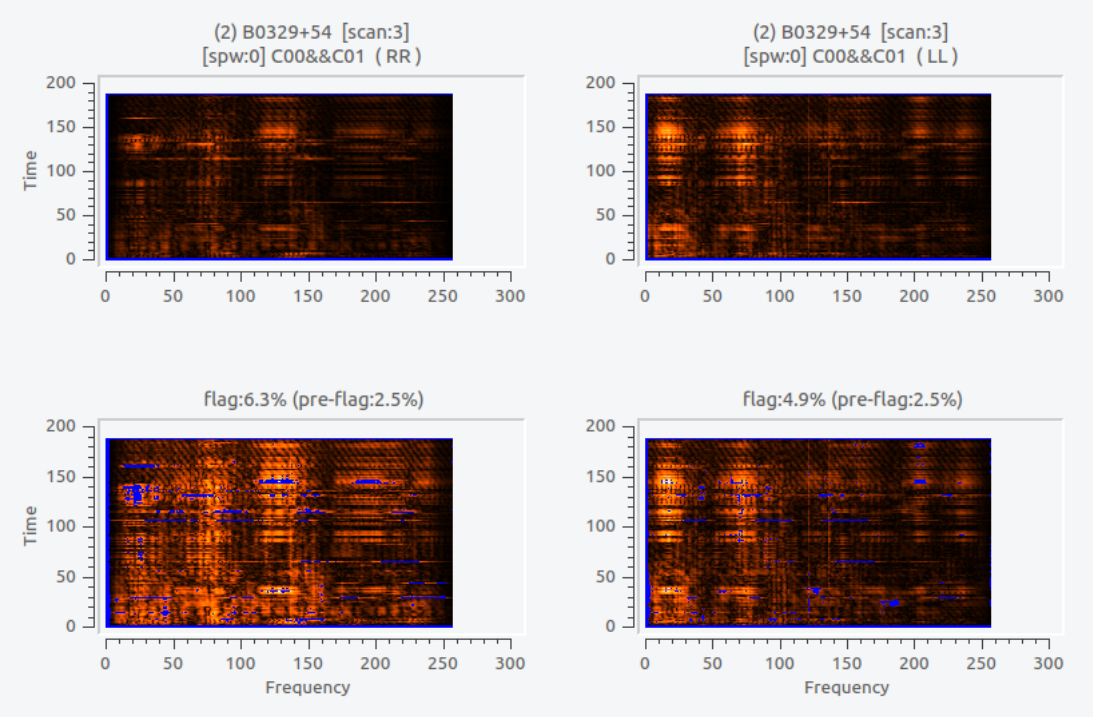


Figure 36: Flagged data for scan 3

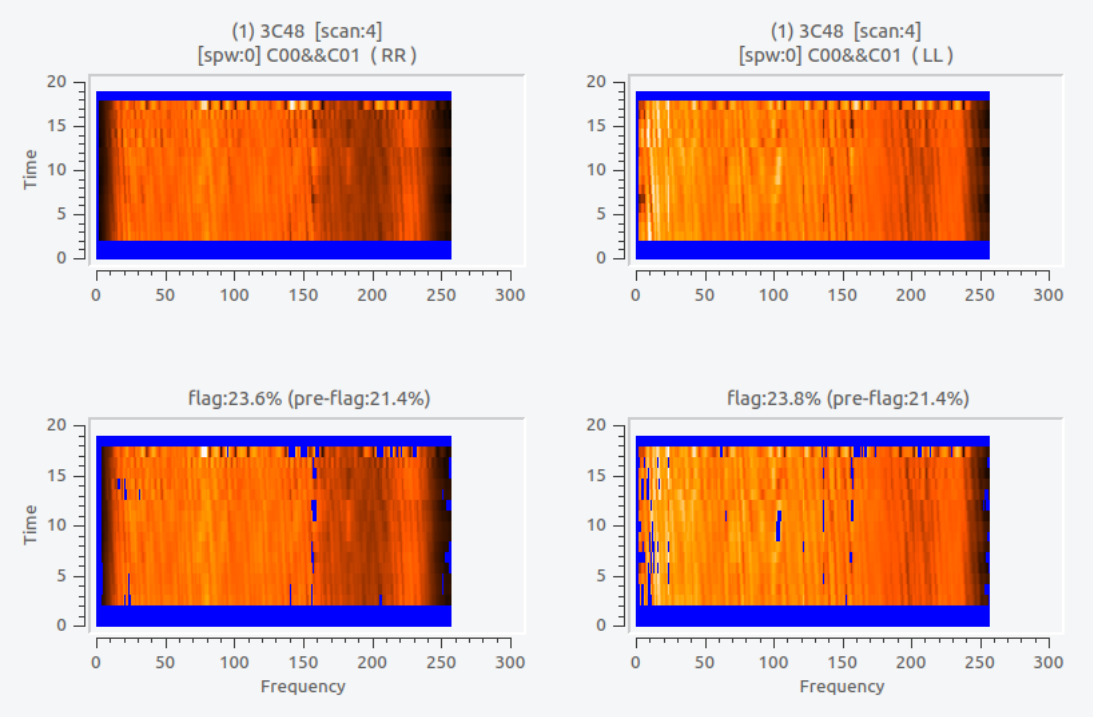


Figure 37: Flagged data for scan 4

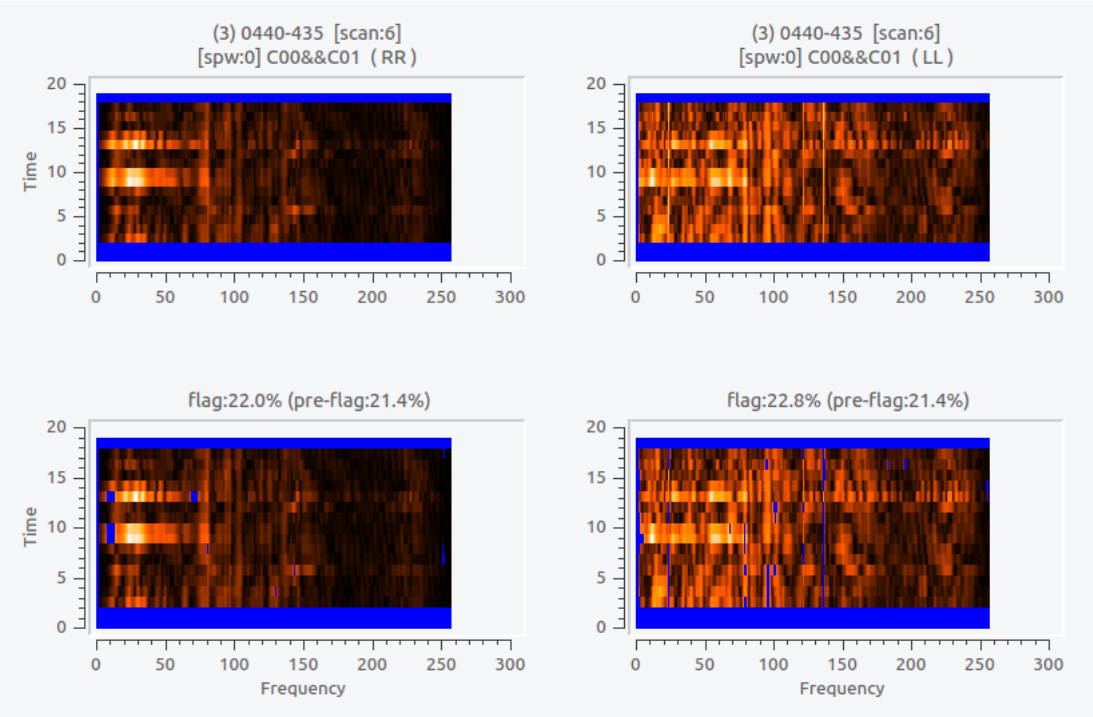


Figure 38: Flagged data for scan 6

We apply tfcrop for the cutoff of 6 since we are still looking at uncalibrated data, hence we should be flagging only the worst RFI.

We follow the given command to apply this flagging:

```
CASA <9>: flagdata(vis='22_GSB.ms',
...:           mode='tfcrop',
...:           spw='',
...:           ntime='scan',
...:           timecutoff=6.0,
...:           freqcutoff=6.0,
...:           extendflags=False,
...:           action='apply',
...:           display='')
```

Let us observe the data once again. We plot Amplitude vs Time iterating over all antennas. We get the following graphs:

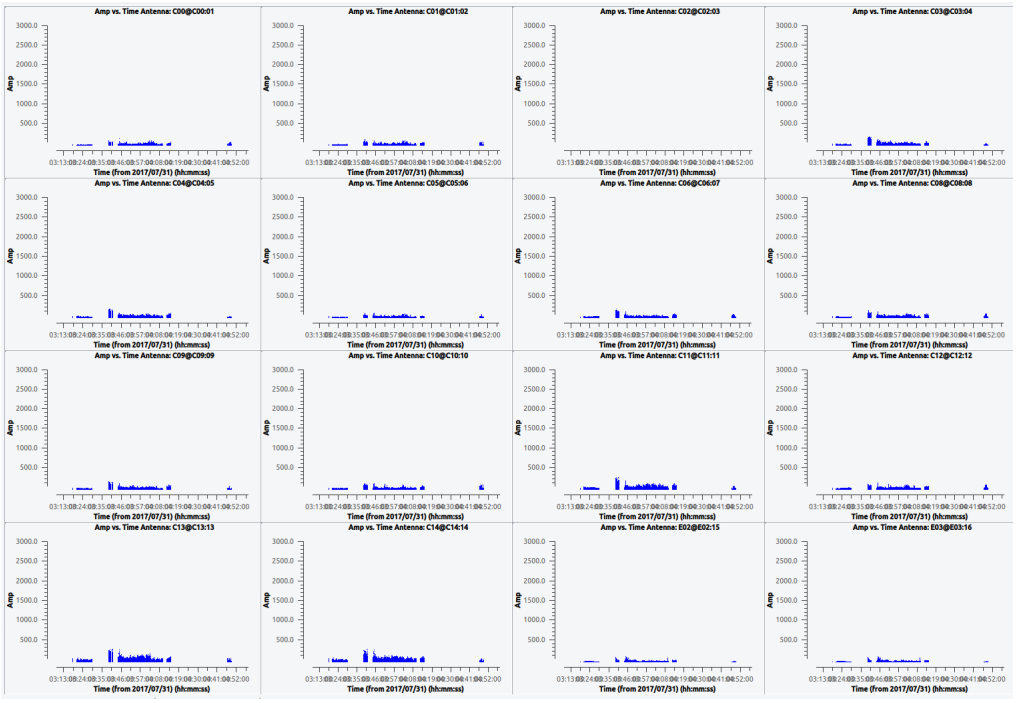


Figure 39: Graph for Amplitude vs Time for all the fields

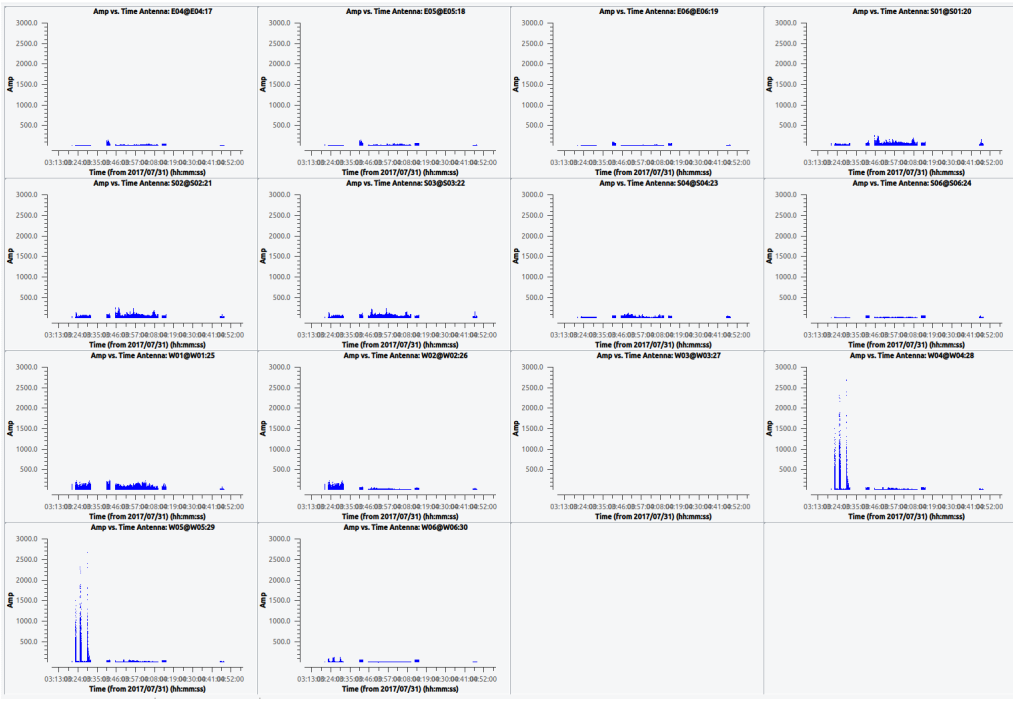


Figure 40: Graph for Amplitude vs Time for all the fields

We notice that the antennas W04 and W05 have very high RFI and show bad behaviour. We flag these out manually using the plotms. We decide to flag out these outliers to the limit of around 50 amplitude. We obtain the following graphs for both these antennas post flagging.

Figure 41: Graph for Amplitude vs Time showing antennas W04 and W05 flagged

2 Pre-calibration

2.1 Reference Antenna

We now look for our reference antenna for calibration. A reference antenna is needed during calibration because interferometers only measure relative phase differences between antenna pairs, not absolute phases. To resolve this ambiguity, one antenna is fixed as the reference, anchoring all phase and gain solutions relative to it. This ensures consistent calibration across time and fields. The reference antenna should be stable, centrally located in the array, and present in most scans. Firstly, we plot antennas vs time to look if any antenna is flawed.

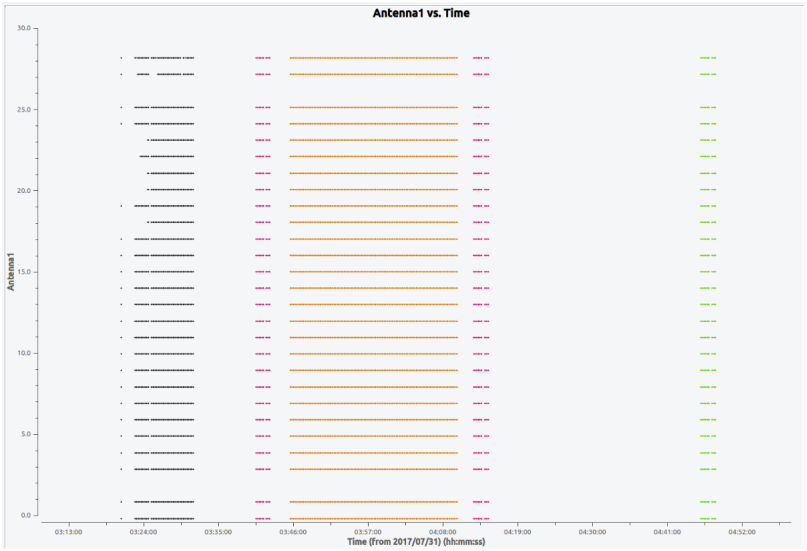


Figure 42: Antenna vs Time colorized by fields.

We observe that W03 and C02 are completely flagged out. As for the rest, we notice that antennas 1, 5, 6, 7, 8, and 10 are missing scans. We do not consider them while looking for a reference antenna. Now we plot graphs on different axes to analyze more.

1. Amplitude vs Time
   We now look at the Amplitude vs Time graphs for all the fields.

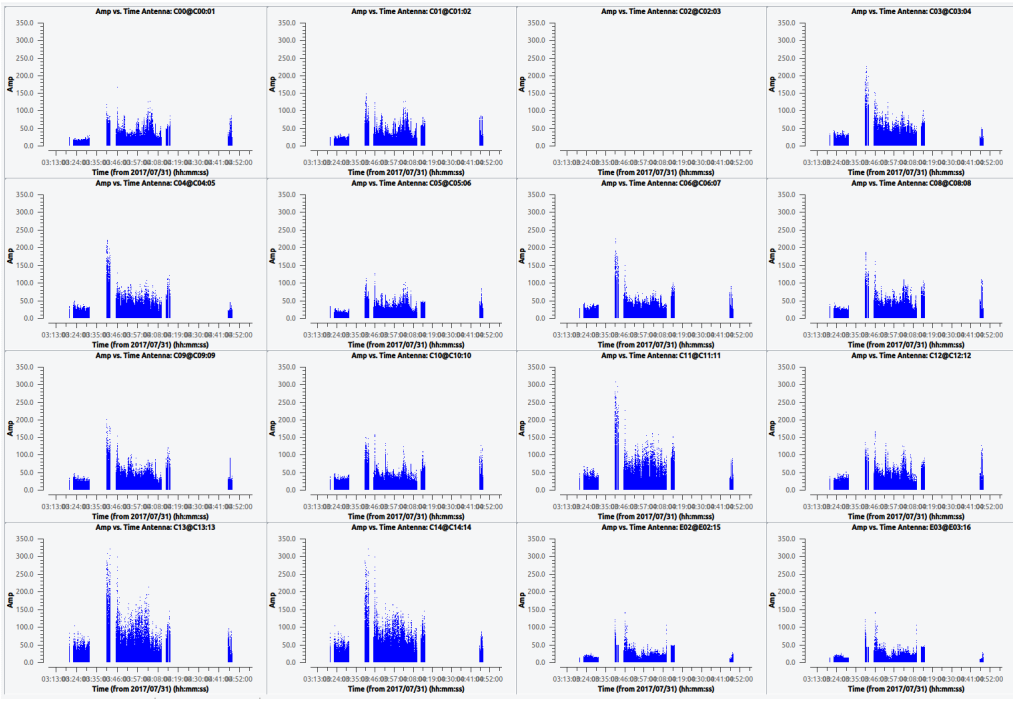


Figure 43: Graph for Amplitude vs Time for flagged data

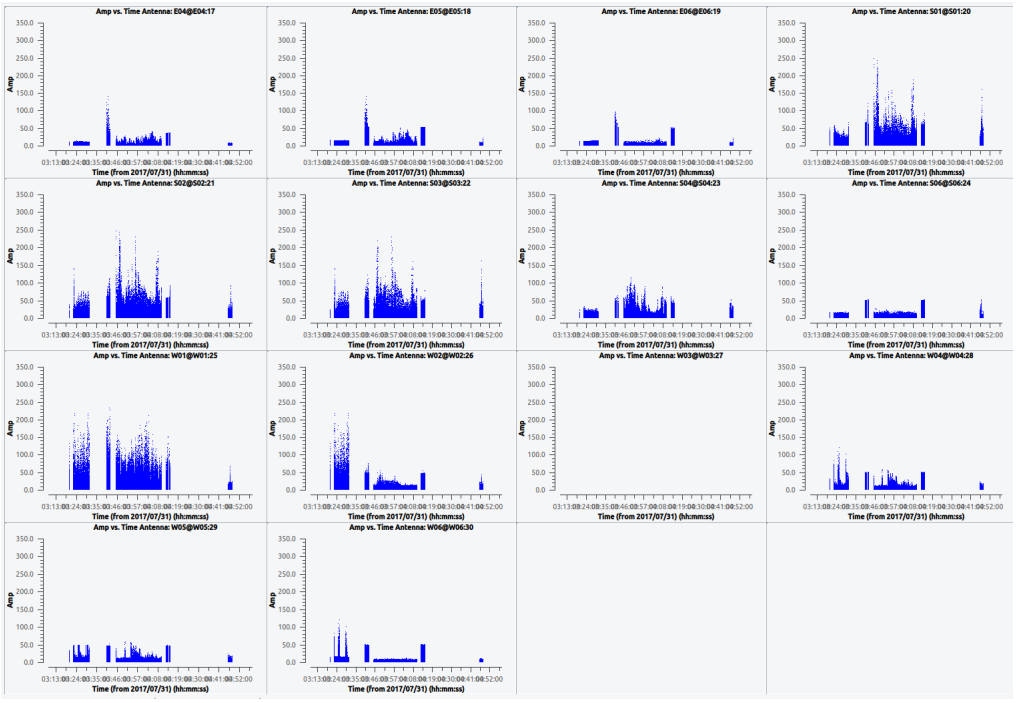


Figure 44: Graph for Amplitude vs Time for flagged data

This looks much better after we flagged it above. The graphs of some antennas show stable amplitudes. We now look at Amplitude vs Time avergaed over 16 seconds for only field 0 and 1 to get a better understanding.

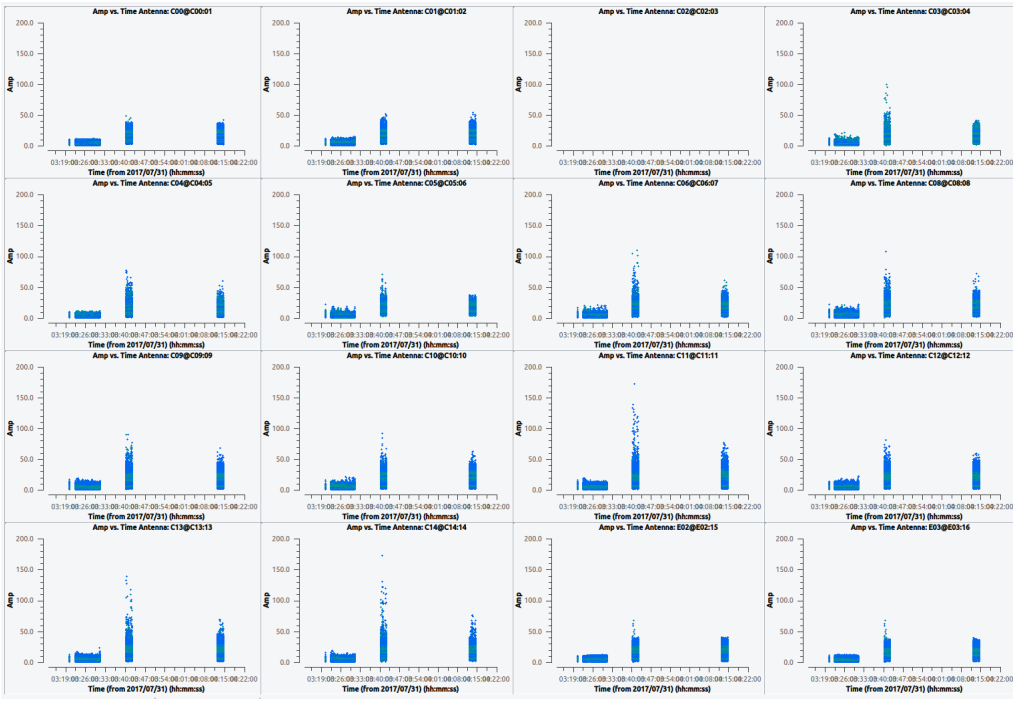


Figure 45: Graph for Amplitude vs Time for the gain and flux calibrator

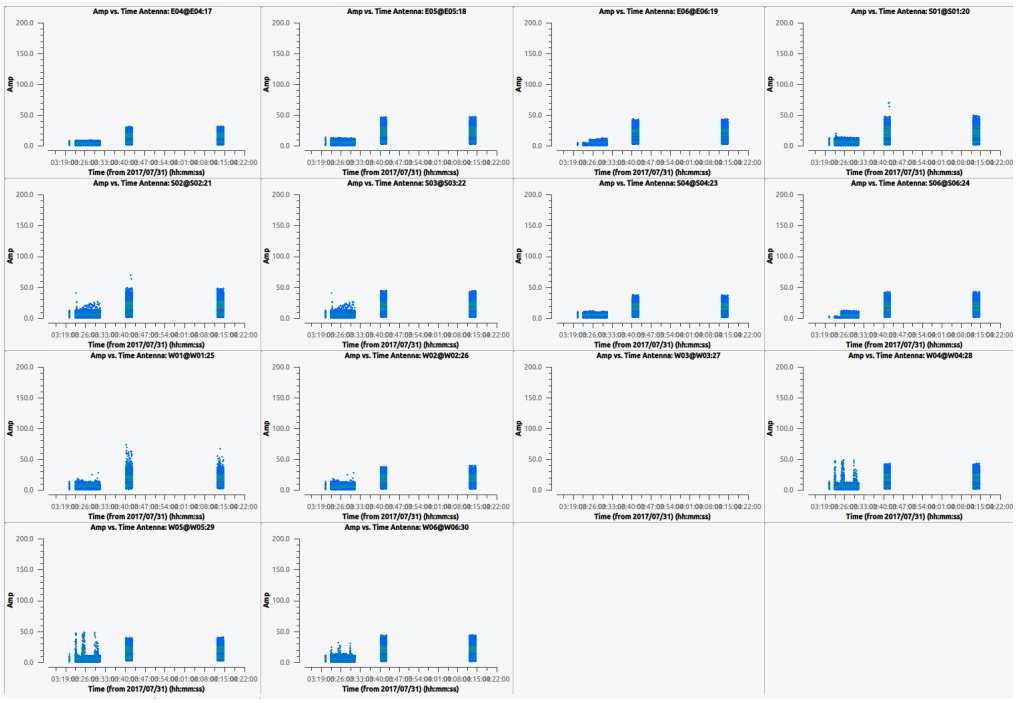


Figure 46: Graph for Amplitude vs Time for the gain and flux calibrator

We will handle other RFI outliers later during calibration. We can now look at Phase vs Time to see which antennas have a stable phase for the gain calibrator.

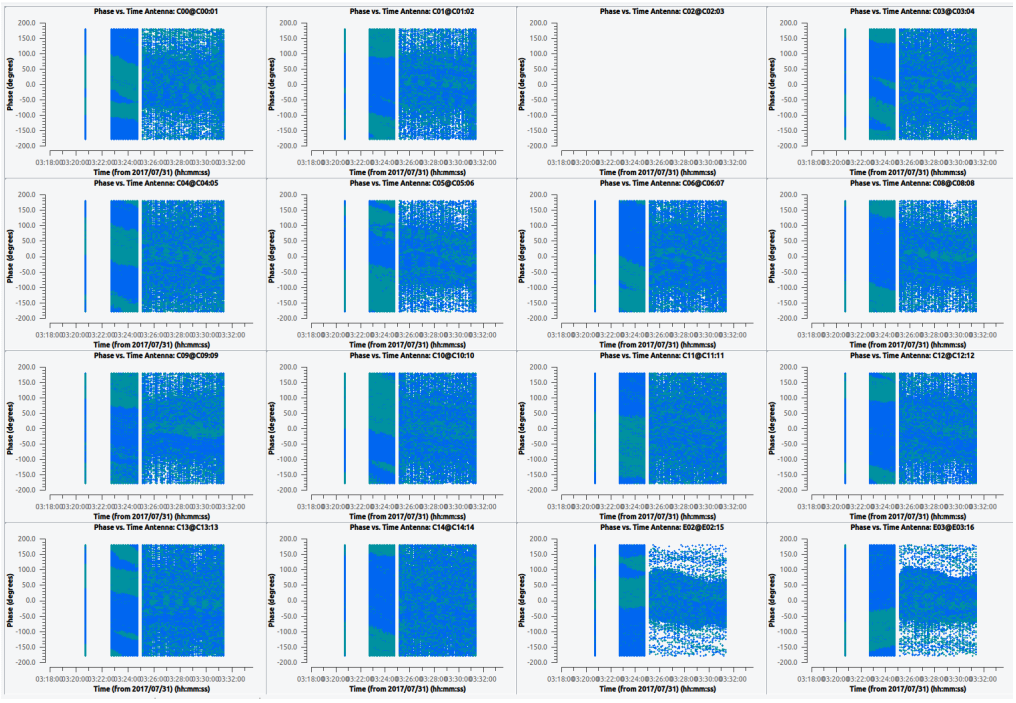


Figure 47: Graph for Amplitude vs Time for the gain calibrator

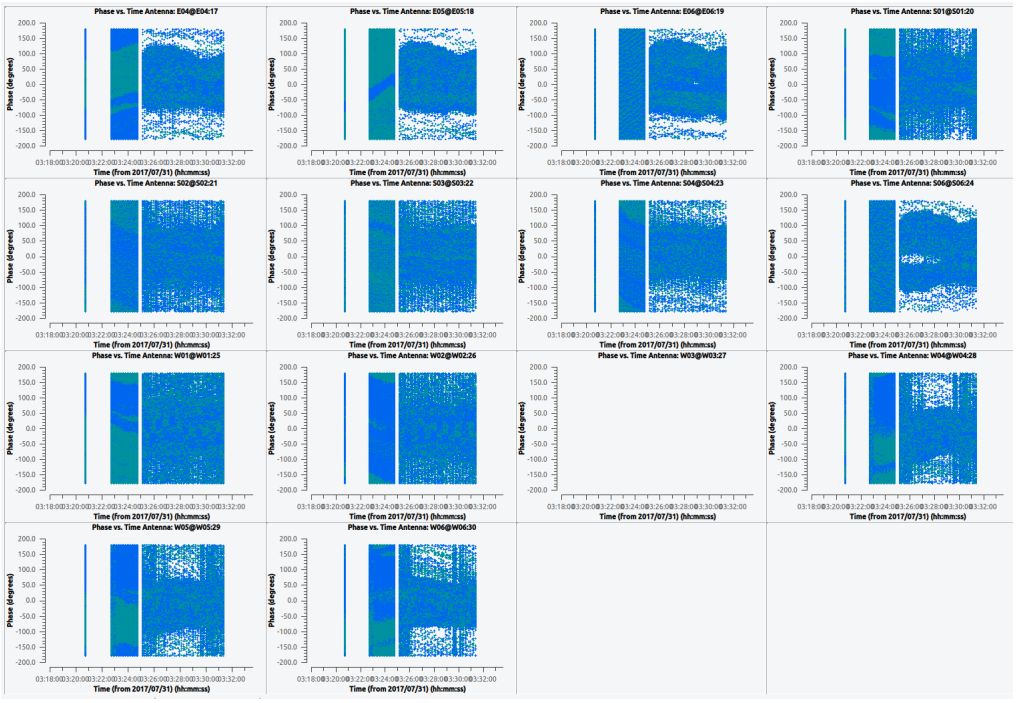


Figure 48: Graph for Amplitude vs Time for the gain calibrator

We notice that the graph of antennas C0, E2 and E5 is comparatively nice. Let us at last look at Phase vs Channel graphs.

2. Phase vs Channel

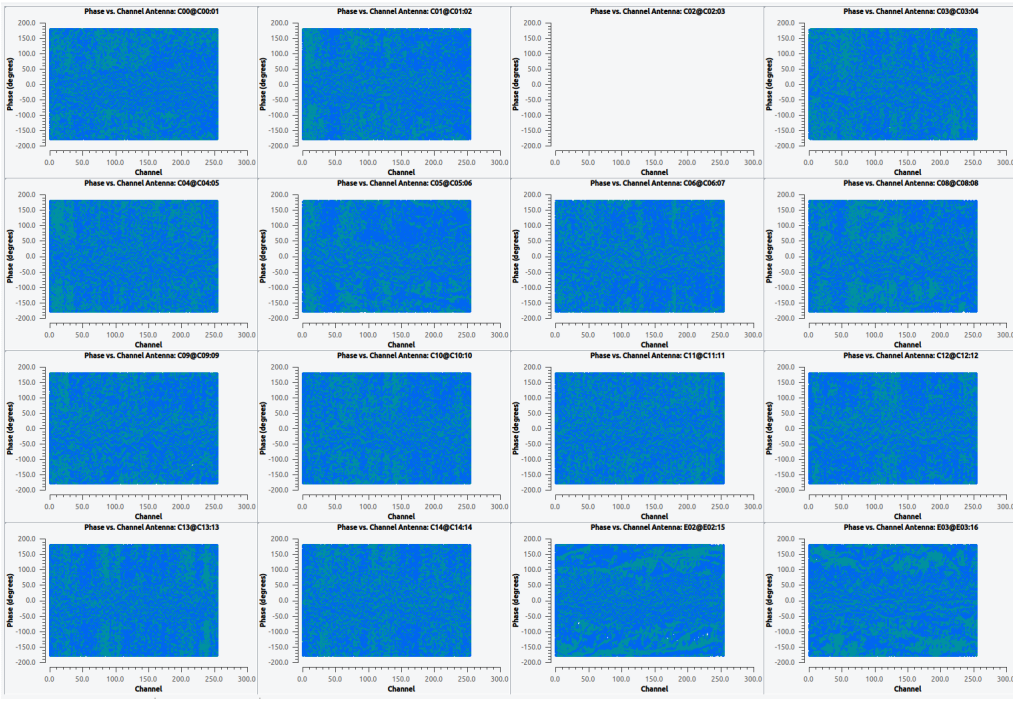


Figure 49: Graph for Phase vs Channel for the flux calibrator

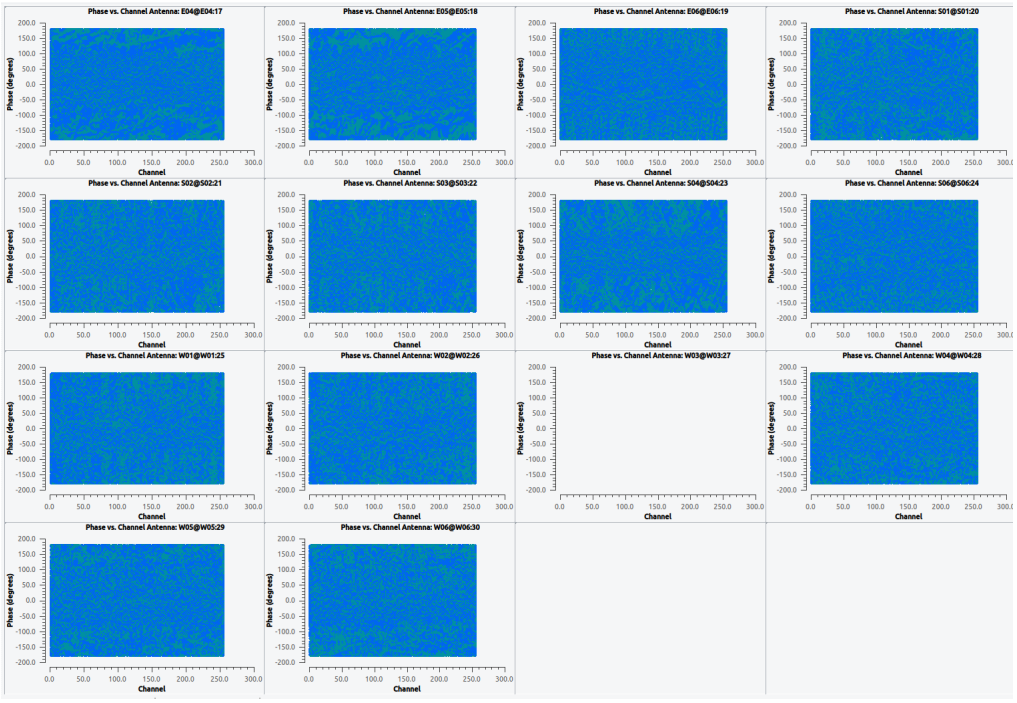


Figure 50: Graph for Phase vs Channel for the flux calibrator

We choose C01 as our reference antenna as it give us a stable phase and amplitude over time.
Now we check the total flagging done using flagdata summary. We get:

‘spw’: ‘0’: ‘flagged’: 13478497.0, ‘total’: 69488640.0
which accounts for 19 percent data flagged. We will now proceed to calibration.

3 Calibration

In an interferometer like the VLA: The signals from all the antennas are combined based on their positions. These combinations (called correlations) depend on knowing exact distances and orientations between every antenna pair, known as baselines. The visibility function (what interferometers measure) depends on the baseline vector (antenna separation in 3D space) and the direction of the source.

We noticed above that some antennas had bad or missing position information at the time of your observation. We can correct for this using gencal in CASA, which will look up the correct antenna positions from the NRAO baseline correction database and apply those corrections to your Measurement Set (MS). Since GMRT antenna positions are usually baked into the Measurement Set already, if you're using GWB or GSB pipeline data, they're generally accurate enough.

The first step is providing a flux density value for the amplitude calibrator J1331+3030. Later, for the final step in determining the calibration solutions, we will use the calibrated gains of the two calibrator sources to transfer the flux density scaling to the secondary gain calibrator (J1822-0938). We use the command:

setjy(vis='22_GSB.ms', listmodels=True)

To calibrate radio data, you need to know the true flux of your calibrator source. CASA uses the task setjy to assign this value. Interferometers (like the VLA) don't measure absolute brightness directly. They measure visibilities (complex correlations), which have arbitrary amplitude units. You need to scale them to Jy (Janskys) using a known calibrator. To do this, you:

1. Pick a calibrator with a known flux (e.g., 3C48).
2. Tell CASA what the correct flux is using setjy.
3. CASA compares the observed visibilities with the known model to derive antenna gains (amplitude scale factors).

So what happens is that we find the correct model image (like 3C48_C.im) based on our frequency. Then we scale the model so that the total flux matches what it should be at each frequency channel. We write this scaled model into your dataset for calibration use and then prints out the flux values it used (e.g., 7.66 Jy at 4.536 GHz). We use the following command:

setjy(vis='22_GSB.ms', field='3C48')

Here, field takes the name of the flux calibrator, standard states the flux scale to use, model is the image of calibrator with known structure and spectrum. usescratch writes

the model directly into the MS which is important for bugs and compatibility. While scalebychan scales the flux per channel, to account for spectral index. We get the following setup output:

```
CASA <4>: setjy(vis='22_GSB.ms', field='3C48', standard='Perley-Butler 2017', model='3C286_C.im', usescratch=True, scalebychan=True, spw='')
Out[4]:
{'1': {'0': {'fluxd': array([45.32011414,  0.        ,  0.        ,  0.        ])},
  'fieldName': '3C48'},
 'format': "{field Id: {spw Id: {fluxd: [I,Q,U,V] in Jy], 'fieldName':field name }}"}
```

Figure 51: Command output of scaling to standard flux

3.1 Initial phase calibration

Now we move onto the initial phase calibration. The reason for this step is to average over the (typically small) variations of phase with time in the bandpass, before solving for the bandpass solution itself. We first plot the amplitude vs channel to see if any data is RFI effected and needs to be removed.

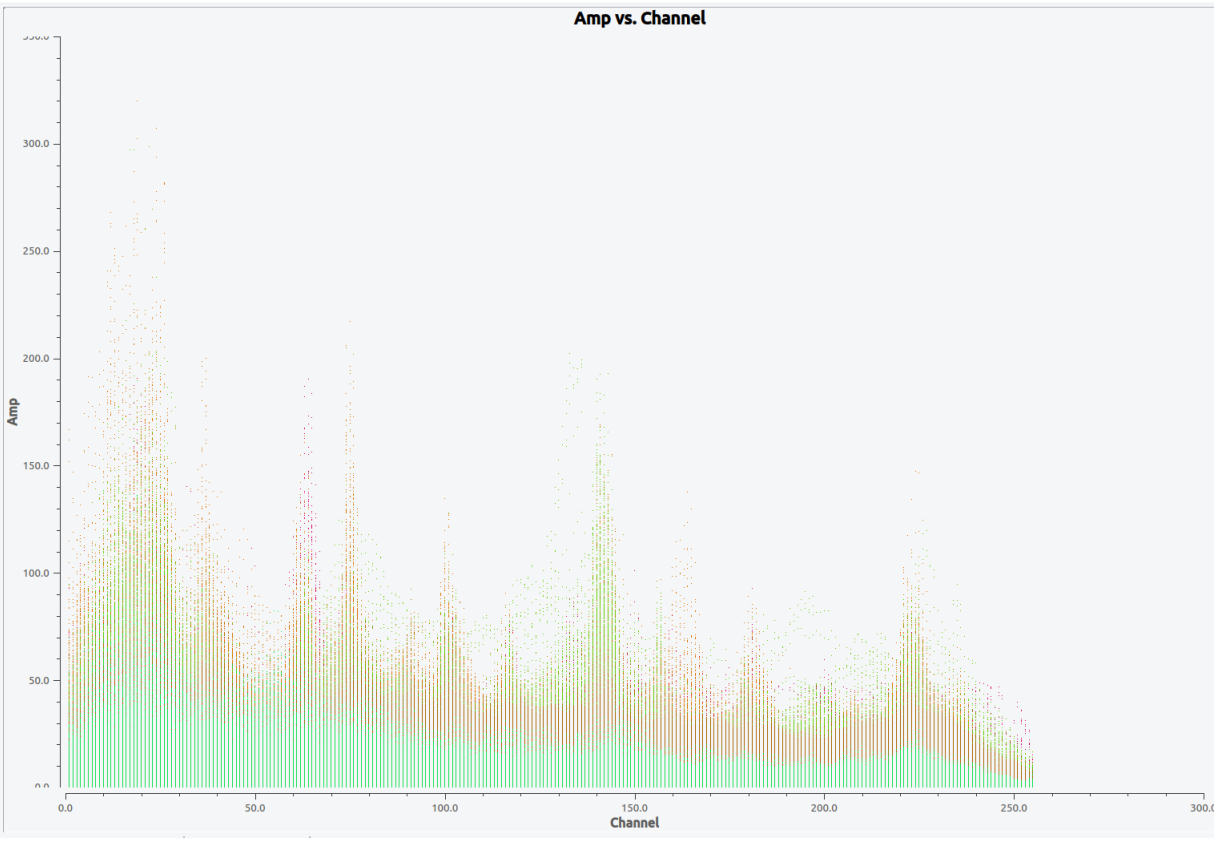


Figure 52: Amplitude vs Channel colorized by fields

We see that Amplitude vs Channel plot for all the calibrators colorized over scans shows us a lot of RFI infected data. So we can choose a subset of the channels from which to determine the gain corrections. We take spw from 50 to 200, as shown in the figure and apply the following command:

```
gaincal(vis='22_GSB.ms', caltable='22_GSB.K1', field='3C48',
        spw='0:50~200', solint='60s', refant='C00',
        solnorm=True, gaintype='K')
```

Here, gaintype='G' tells the system to compute the complex gain solutions, one per antenna per spw per polarization per solution interval. Calmode solves for only the phase portion of the gain whereas solint tracks the phases, a short solution interval is chosen (int refers to a single integration time or 10 seconds for this case). We can change the snr depending upon the source and frequency.

We now look at Amplitude vs time of all the fields except the target field in then spw range 50 to 200. We get:

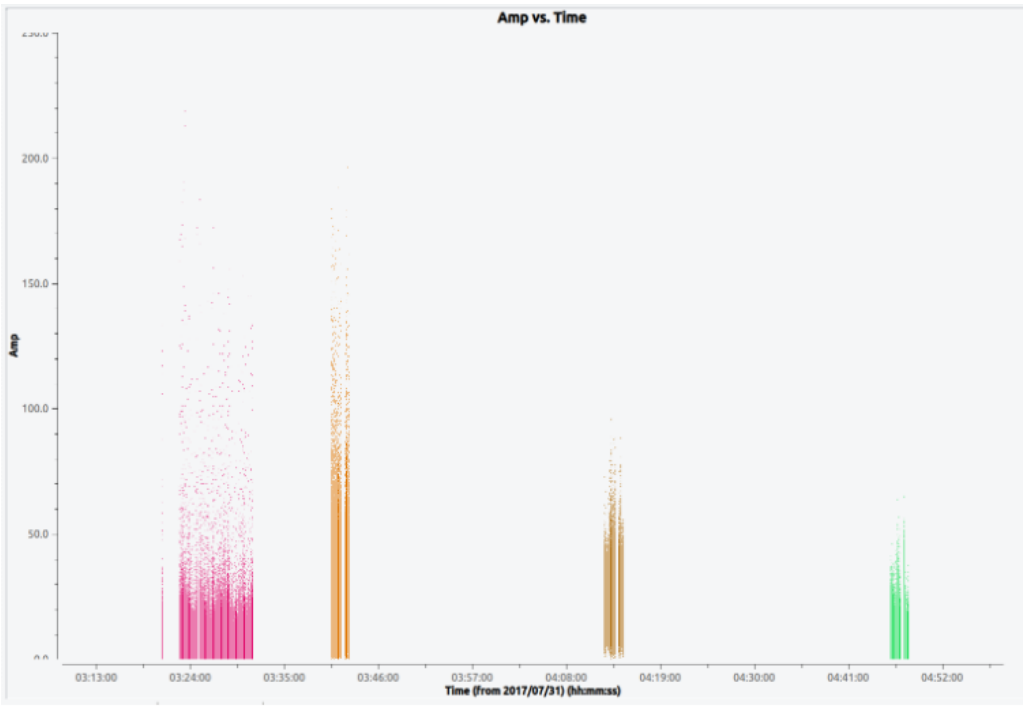


Figure 53: Amplitude vs Time colorized over fields

We can also plot delay vs time for each antenna opening up the delay file we created. We get:

Figure 54: Delay vs Time for every antenna

Figure 55: Delay vs Time for every antennas

We see that it clears up any delay for the calibrators. We now move on to the initial gain calibration.

3.2 Initial Gain Calibration

```
gaincal(vis='22_GSB.ms', caltable='22_GSB.G0', field='3C48',
        spw='0:50~200', solint='int', refant='C00', minsnr=5.0,
        gaintype='G', calmode='p', gaintable=['22_GSB.K1'])
```

Now we follow with:

```
gaincal(vis='22_GSB.ms', caltable='22_GSB.G1', field='3C48',
        spw='0:50~200', solint='inf', refant='C00', minsnr=5.0,
        gaintype='G', calmode='ap',
        gaintable=['22_GSB.K1', '22_GSB.G0'])
```

Now we have the following tables:

1. K1:delays (gaintype 'K')
2. G0: time-dependent phase (gaintype 'G', calmode='p')
3. G1: time-dependent amp + phase (gaintype 'G', calmode='ap')

Now we plot gain phase vs time to take a look at the corrected phase for field 1:

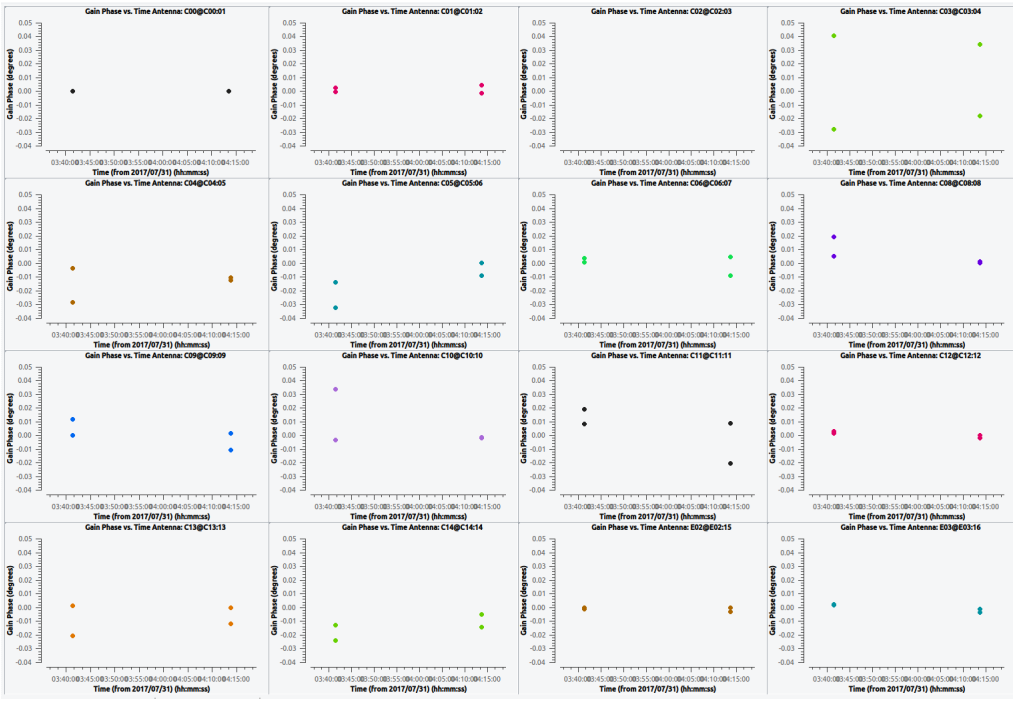


Figure 56: Gain vs Time for the flux calibrator

Figure 57: Gain vs Time for the flux calibrator

These are excellent gain solutions all lying very close to each other! We now move on to bandpass calibration.

3.3 Bandpass calibration

We apply:

```
bandpass(vis='22_GSB.ms', caltable='22_GSB.B1', field='3C48',
         spw='0:50~200', solint='inf', refant='C00', minsnr=5.0,
         solnorm=True, gaintable=['22_GSB.K1','22_GSB.G1'])
```

We plot Amplitude vs Frequency iterated over all antennas and see:

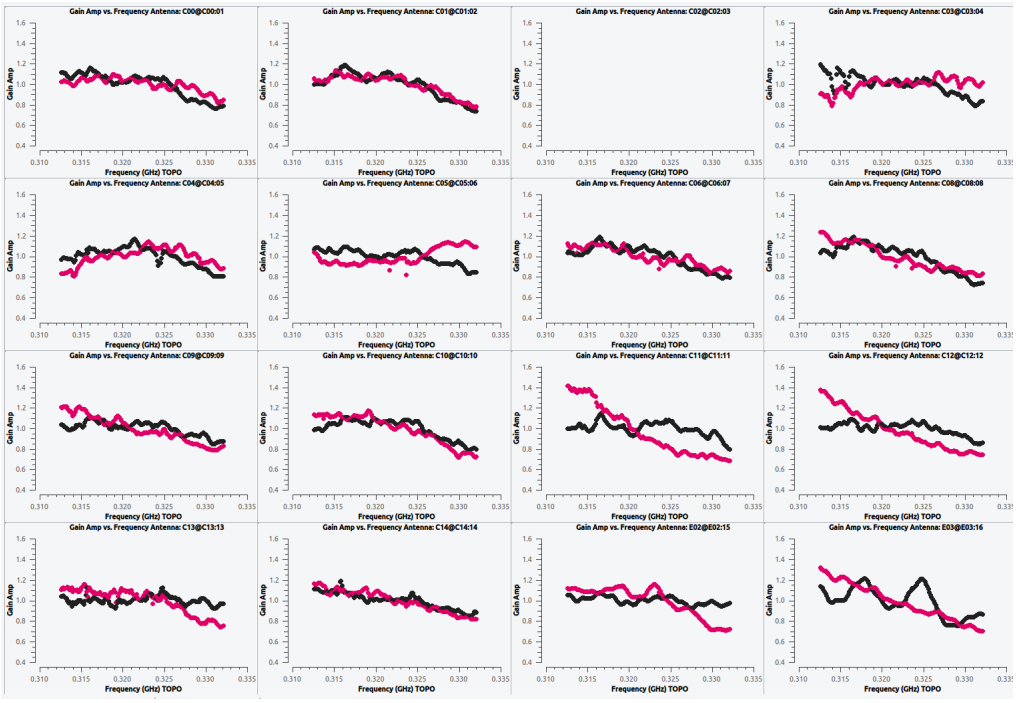


Figure 58: Gain amplitude vs Frequency iterated over antennas

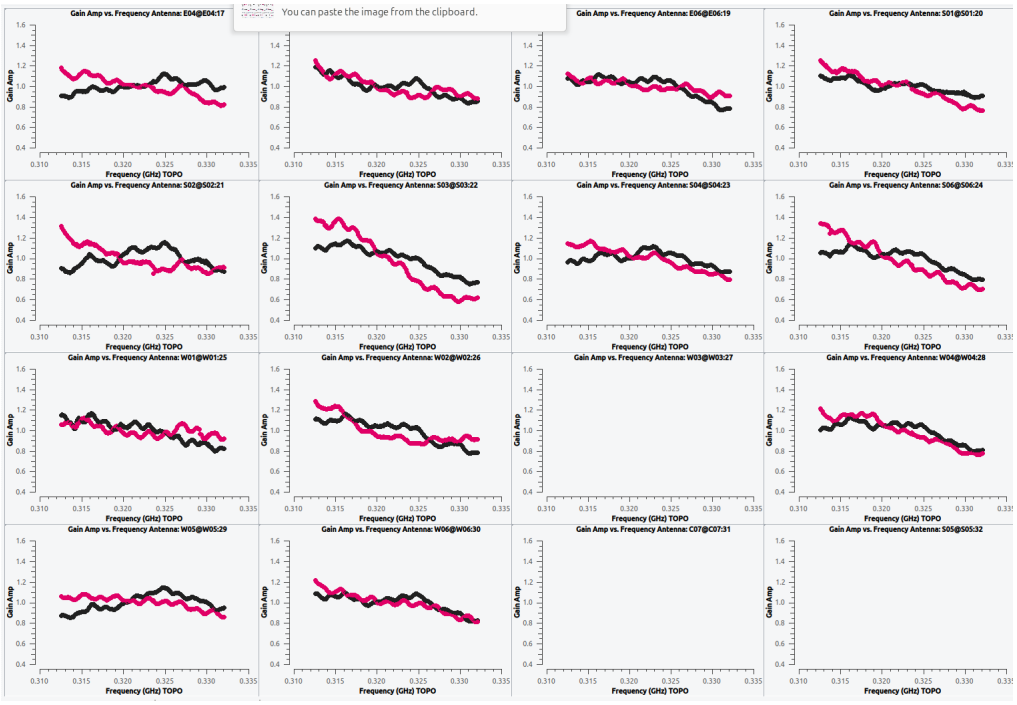


Figure 59: Gain amplitude vs Frequency iterated over antennas

We now go on to apply phase+amplitude calibration on the primary and secondary calibrators by following the given command:

```
gaincal(vis='22_GSB.ms', caltable='22_GSB.AP.G1', field='0,1',
        spw='0:50~200', solint='120s', refant='C00', minsnr=2.0,
        gaintype='G', gaintable=['22_GSB.K1', '22_GSB.B1'],
        interp=['nearest,nearestflag', 'nearest,nearestflag'])
```

3.4 Applying the calibration

Now finally we set the flux density of the phase calibrator:

```
fluxscale(vis='22_GSB.ms', caltable='22_GSB.AP.G1',
          fluxtable='22_GSB.fluxscale', reference='3C48',
          transfer='0837-198')
```

Finally we move onto applying these calibrations to the main target source. We start off with applying calibration to the amplitude and phase calibrators. We use the command:

```
applycal(vis='22_GSB.ms', field='3C48', spw='0:50~200',
         gaintable=['22_GSB.fluxscale', '22_GSB.K1', '22_GSB.B1'],
         gainfield=['3C48', '', ''],
         interp=['nearest', '', ''],
         calwt=[False])


applycal(vis='22_GSB.ms', field='0837-198', spw='0:50~200',
         gaintable=['22_GSB.fluxscale', '22_GSB.K1', '22_GSB.B1'],
         gainfield=['0837-198', '', ''],
         interp=['nearest', '', ''],
         calwt=[False])
```

We now apply it to the target source:

```
applycal(vis='22_GSB.ms', field='B0329+54', spw='0:50~200',
         gaintable=['22_GSB.fluxscale', '22_GSB.K1', '22_GSB.B1'],
         gainfield=['0837-198', '', ''],
         interp=['nearest', '', ''],
         calwt=[False])
```

We let us look at the resultant graphs:

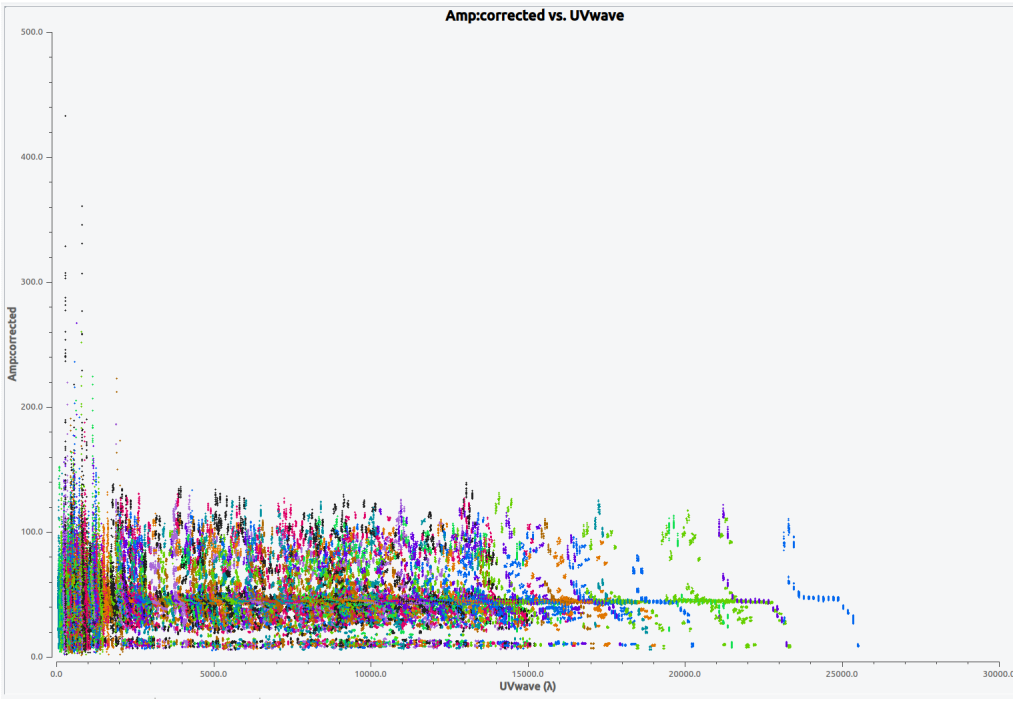


Figure 60: Amplitude vs UVwave colorised over antennas and averaged over 16 channels for flux calibrator

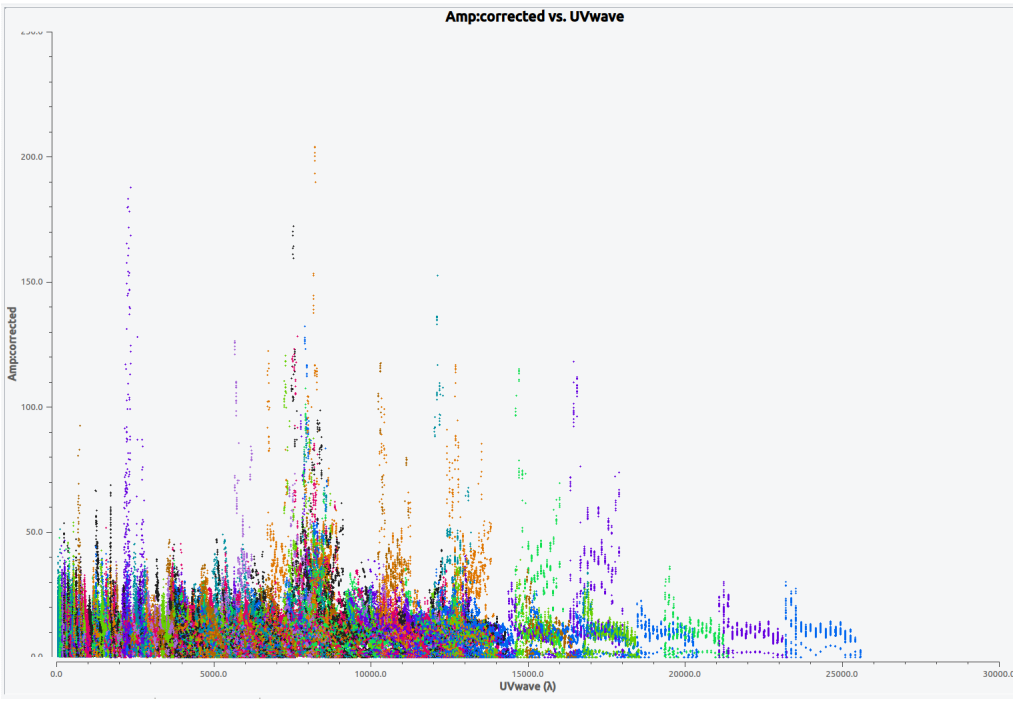


Figure 61: Amplitude vs UVwave colorised over antennas and averaged over 16 channels for gain calibrator

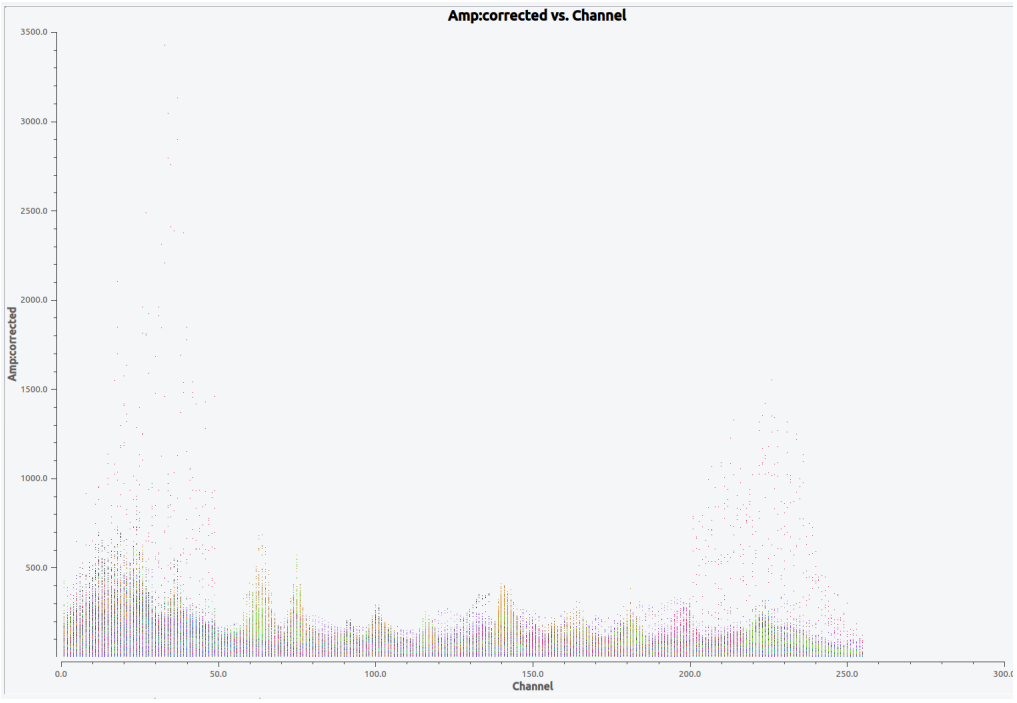


Figure 62: Amplitude vs UVwave colorised over antennas

We see that some RFI still pertains. We will now carry on second round of flagging to get rid of this.

4 Post-calibration flagging and recalibration

After having obtained our calibration tables post the first round of calibration , we apply the tables only to our calibrators and not to the target just yet. We do so to further improve our flagging by removing the RFI which will now be more obvious than before. For example , in the plot seen below , generated from scan 2 of the '32_089_9622_GSB' dataset after having applied the first round of calibration , we can clearly view the RFI towards the end of the bandwidth.

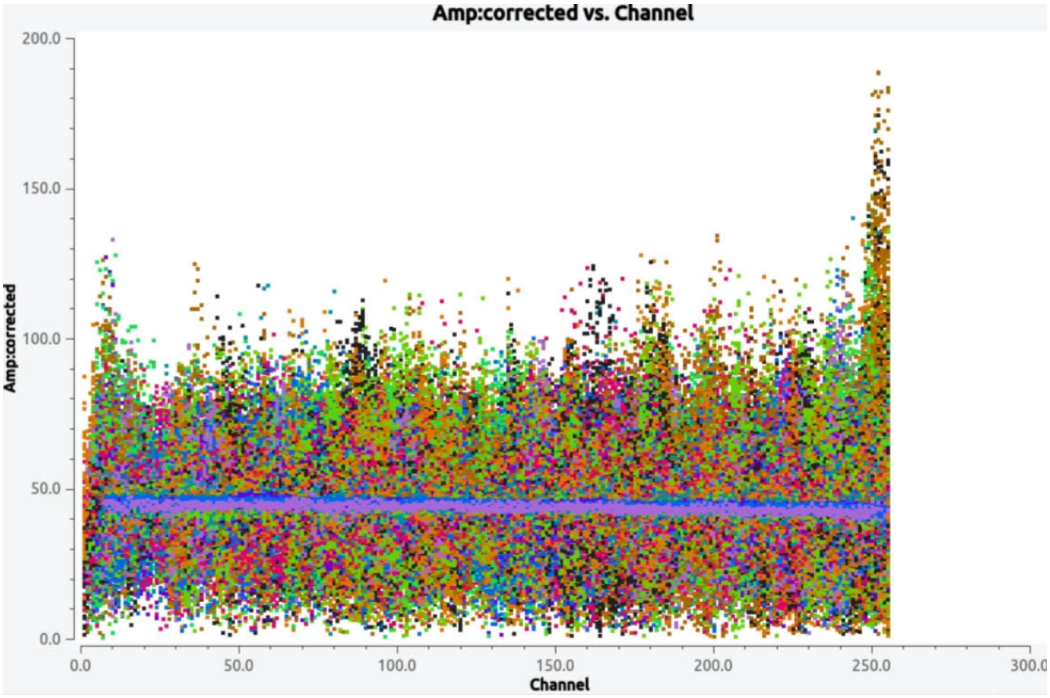


Figure 63: Amp:corrected vs Channel for scan 2, corresponding to the field '3C48'

At this point , we can commit to more aggressive flagging due to the distinction between RFI and data being clearer. We thus use the 'rflag' task here which computes flags on a global level , i.e. , taking all baselines into account , while respecting scan boundaries. The 'tfcrop' task , which we generally use in pre calibration flagging, computes flags at an individual baseline level.

Here , some trial and error is required to fine-tune our input parameters and obtain flagging that correctly flags all RFI. Thus , multiple iterations with different 'timedevscale' and 'freqdevscale' values , among other parameters , may be required.
A very important note here , with regards to the input parameters , is that the 'datacolumn' field must be set to 'CORRECTED' to flag the calibrated data which is present in the newly created 'CORRECTED' data column of our measurement set.
For example , one such command in the process could look like:

```
flagdata(vis='32_089_9622_GSB.ms', mode='rflag', scan='2',
         timedevscale=7, freqdevscale=7, action='apply',
         display='both', datacolumn='CORRECTED')
```

One important disciplinary measure to remember while using ‘rflag’ or ‘tfcrop’ is to first use ‘action = calculate’ , check the plots that come up to see if they’re actually flagging the RFI the way we want them to and then use ‘action = apply’ only if the flagging is satisfactory.

We thus obtain much better plots for our calibrators post this flagging.

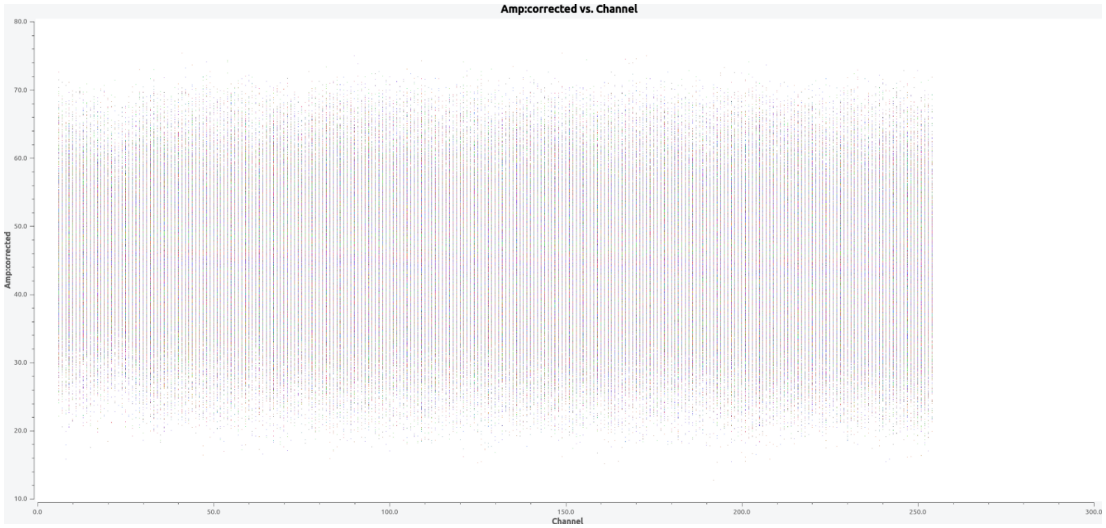


Figure 64: Amp:corrected vs channel for scan 2 , corresponding to the field ‘3C48’ , having removed most RFI thus obtaining a more or less constant amplitude across the entire bandwidth.

Having now obtained better and cleaner calibrator data, we restart the calibration process in the hopes of better solutions. Starting right from the ‘setjy’ step all the way to transferring the flux scale of the primary calibrator to the secondary calibrator using the ‘fluxscale’ command. This now is essentially an iterative process wherein we keep repeating the process of flagging and recalibrating until we deem the solutions found fit to be applied to the target.

Once we complete this iterative process to a satisfactory level of solutions , we go ahead and apply our solutions to our target. An important note here is to use our secondary calibrator, which in the case of the ‘9622’ dataset is ‘0440-435’, as our gainfield. The command for the same is as follows:

```
applycal(vis='32_089_9622_GSB.ms', field='2',
         gaintable=['32_089_9622_GSB.fluxcale11',
                    '32_089_9622_GSB.K01',
                    '32_089_9622_GSB.B01'],
         gainfield=['0440-435', '', ''],
         interp=['nearest', '', ''],
         calwt=[False])
```

The reason we use the secondary calibrator here is to determine the appropriate complex gains for the target source by minimising differences through the atmosphere (neutral

and/or ionized) between the lines of sight to the phase calibrator and the target source as the primary calibrator is usually chosen to be far away from our target of interest.

Once we apply the solutions to the target , we plot the corrected amplitudes and phases against time , channel and baseline length (in the units of wavelength), to identify RFI and then flag them, similar to what we had done for the calibrators.

Once again we use 'rflag' here as now we can be much less conservative even with our target data. We try to fine-tune the parameters through multiple iterations and variations in the various parameter values. An example command in the process can be:
At the end of this process we obtain the following plots:

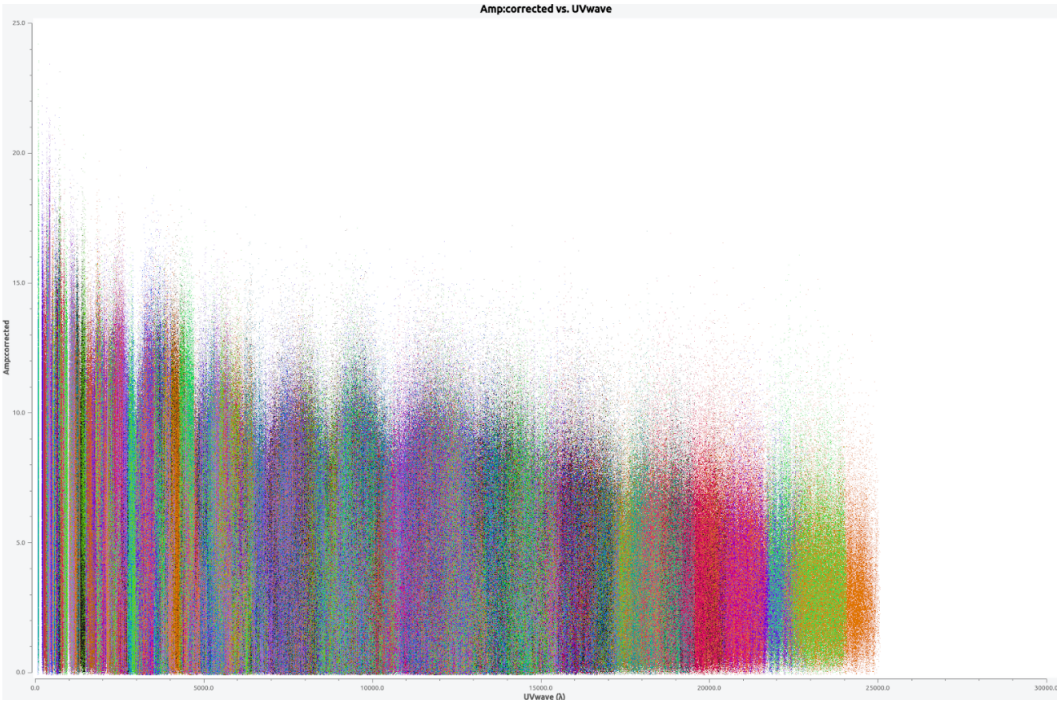


Figure 65: Amp:corrected vs UVwave (baseline length in units of wavelength)

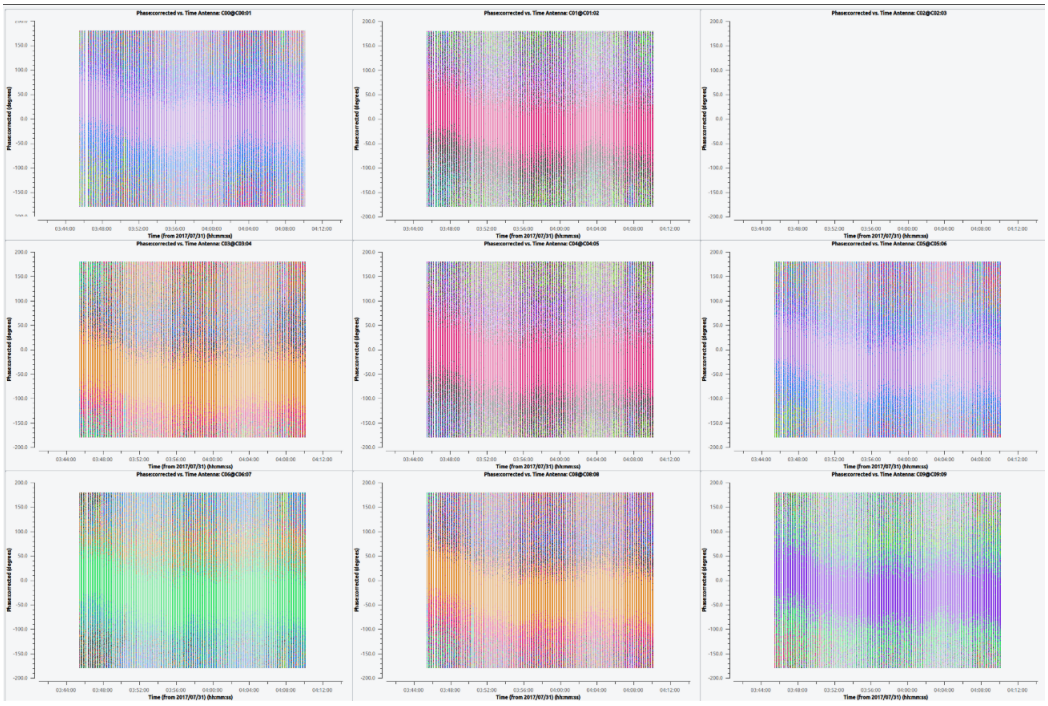


Figure 66: Phase:corrected vs time corresponding to the target scan

The amp:corrected vs UVwave plot aligns with our theory which tells us that for a resolved source the amplitude received by a baseline decreases as the baseline length increases, or in other words, as the resolution gets finer. Additionally, a clear pattern is also visible in the phase:corrected vs time plot when colorising by baseline.

We, thus, having properly calibrated our target data, move forward to imaging.

5 Imaging

Imaging in CASA is the process of converting raw visibility data from radio interferometers into meaningful sky images. This involves transforming measurements from the spatial frequency (u-v) domain to the image plane using Fourier transforms. The primary task used for imaging in CASA is tclean, which reconstructs the sky brightness distribution while accounting for the instrument's response (i.e., the synthesized beam). The imaging process can also include deconvolution techniques like CLEAN to remove artifacts introduced by the telescope's limited u-v plane sampling. CASA allows for flexible control over parameters like image size, resolution, weighting, and spectral setup, enabling astronomers to produce scientifically accurate and high-fidelity images of celestial sources.

5.1 Tclean

Tclean is the primary imaging task in CASA used to reconstruct images from calibrated visibility data obtained by radio interferometers. It performs the inverse Fourier transform of the measured visibilities to create an initial "dirty" image, and then applies deconvolution algorithms most commonly the CLEAN algorithm to remove artifacts caused by the incomplete sampling of the u-v plane. tclean offers advanced options for imaging, including multi-frequency synthesis, multi-scale CLEAN, wide-field imaging, and interactive cleaning. It replaces the older clean task and is designed to be more flexible, robust, and user-friendly, supporting modern imaging needs in radio astronomy [2].

Tclean has various parameters with the basic parameters in the tclean task defining the fundamental setup of the imaging process. The vis parameter specifies the input Measurement Set (MS) that contains the calibrated visibility data to be imaged. The imagename sets the name prefix for all output image products generated by the task. imsize determines the dimensions of the output image in pixels, typically provided as a two-element list (e.g., [1024, 1024]) but can be a single element list as well (eg.,[4000]), while cell defines the angular size of each image pixel, usually in arcseconds (e.g., '0.5arcsec') [3].

The pixel size is usually taken to be 4-5 times smaller than the resolution for better understanding of the shapes and sizes of structures. The field parameter allows selection of one or more fields from the MS to be included in the imaging process, and spw specifies which spectral window(s) and channel ranges should be imaged. Finally, the phasecenter sets the central sky position of the output image, which can be specified either by coordinate or by field name/index. These parameters form the core of any imaging configuration in CASA's tclean [4].

The tclean task in CASA offers a wide range of parameters that allow precise control over the imaging process. Under spectral settings, the specmode parameter defines the imaging mode, with options such as 'mfs' for multi-frequency synthesis, 'cube' for spectral cube imaging, or 'cont' for continuum imaging. When imaging in cube mode, nchan sets the number of output image channels, start determines the starting channel or frequency, and width defines the width of each channel, either in channel numbers or frequency units.

In terms of gridding and weighting, the gridder parameter selects the gridding algorithm

used during image formation, with options like ‘standard’, ‘mosaic’, or ‘awproject’, depending on the observation type and array configuration. The weighting parameter specifies the visibility weighting scheme, such as ‘natural’, ‘uniform’, or ‘briggs’, and the robust value adjusts the trade-off between resolution and sensitivity when using Briggs weighting [4].

For deconvolution and cleaning, the deconvolver parameter sets the CLEAN algorithm to use, such as ‘clark’, ‘multiscale’, or ‘mtmfs’, tailored to different imaging scenarios. The niter parameter defines the maximum number of CLEAN iterations, while threshold sets the stopping point in terms of flux level. The gain parameter determines the fraction of peak flux removed in each iteration, and cycleniter and cyclefactor control the major cycle loop behavior and when major cycles are triggered. Under masks and region control, the mask parameter allows the use of user-defined or auto-generated CLEAN masks, and usemask determines how the mask is applied during the deconvolution process—either manually or through automated methods like ‘auto-multithresh’.

Finally, for output control, the interactive parameter enables interactive CLEANing if set to True, allowing real-time user guidance during deconvolution. The savemodel option decides whether and how the model image is saved back to the measurement set, with choices such as ‘none’, ‘modelcolumn’, or ‘virtual’. These parameters together provide comprehensive flexibility for crafting high-quality radio images in CASA. We use these specific parameters to create the image. Once the task ends successfully, you can view the image using imview. The main image you would like to see is the one with the extension image.tt0. Adjust the data range to display it in a reasonable colour scale to show the peaks as well as the noise in the image. Also inspect the rest of the images created by tclean - the .psf, .mask, .model, .residual [5].

5.2 Self-calibration

This is essentially an iterative process to obtain better and better images. The model from the first tclean is used to calibrate the data and the corrected data are then imaged to make a better model and the process is repeated. The order of the tasks is tclean, gaincal, applycal, tclean. A reasonable choice is to do 5 phase only and two amplitude and phase self-calibrations. We start from a longer “solint” (solution interval), for e. g. “8min” and gradually lower it to “1min” during the phase only iterations. For “a&p” self-calibration, again choose a longer solint such as “2min”.

We keep solnorm=False in phase only iterations and solnorm=True in “a&p” self-calibration. As the iterations advance, the model sky is expected to get better so in the task tclean, lower the threshold and increase niter.In the next iteration we will use a larger niter and lower the threshold. In the tclean messages do check the total cleaned flux and the number of iterations needed to reach that.We repeat this until we stop seeing improvement in the image sensitivity [1].

5.3 Imaging process

Before we start imaging the data , we first take a look at the dirty image by keeping niter = 0 , i.e. , no deconvolution is done. This gives us a feel about the artefacts and the PSF corresponding to our data.

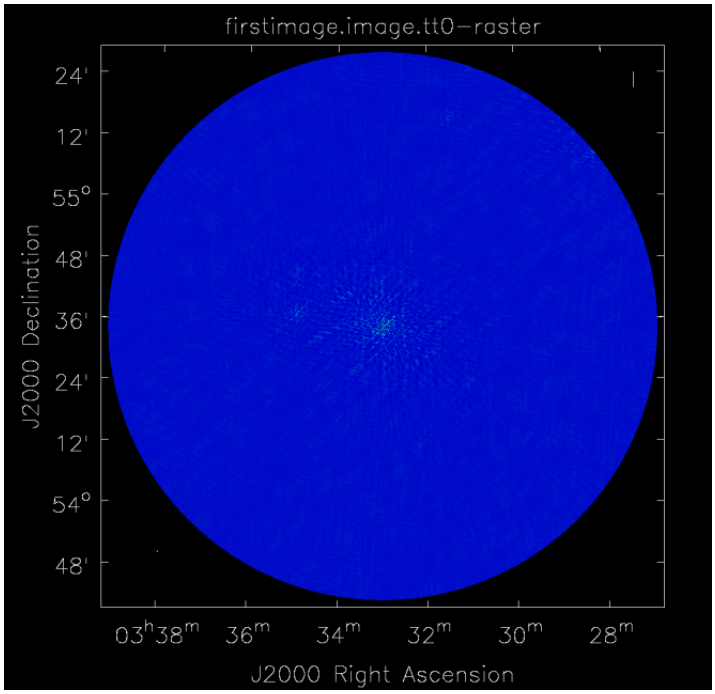


Figure 67: Dirty image with no deconvolution

In this image we can see a lot of artefacts and patterns near the point sources. By deconvolving , we hope to remove such artefacts and get a cleaner image.
Next , we run tclean keeping interactive = True , to allow us to mask regions after each major cycle. Here , as it is our first image , we keep niter = 2000 , a comparatively low value.

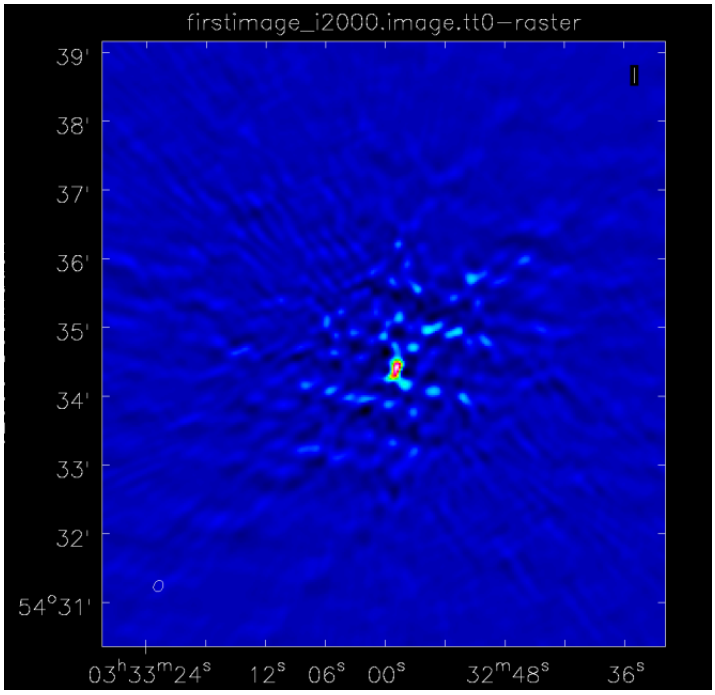


Figure 68: Image obtained after running tclean for the first time

Zooming in at the source, we see that much of the previously present patterns and artefacts are now gone. However, there still exists a spread near the source and to clear this, we now utilise self calibration, with a high enough solint (in our case '8min').
Once again, after applying this calibration, we go ahead and run tclean, this time with a higher number of iterations, niter = 4000, keeping interactive = True, to effectively mask the regions.

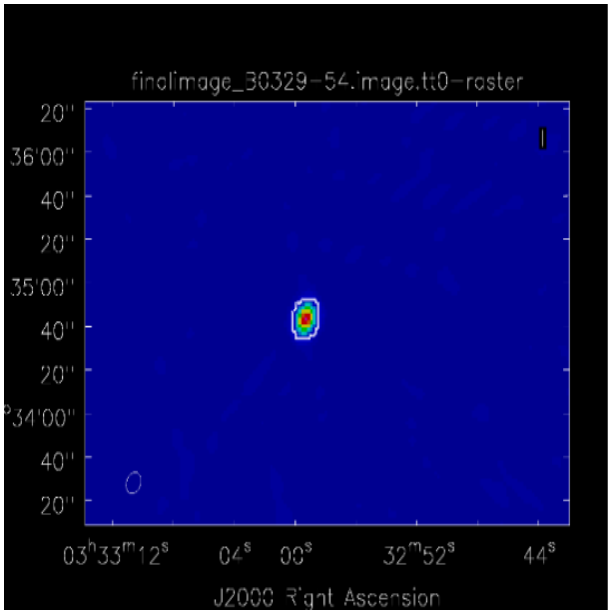


Figure 69: Image obtained after running tclean with niter = 4000

We see that we get a far better image, with the point source resembling the shape of

our primary beam as well. However, the residual of the process shows that some sources were left out. Additionally, in the image itself we can still spot some leftover artefacts. The white contour around the image shows the masking that we did during the imaging process.

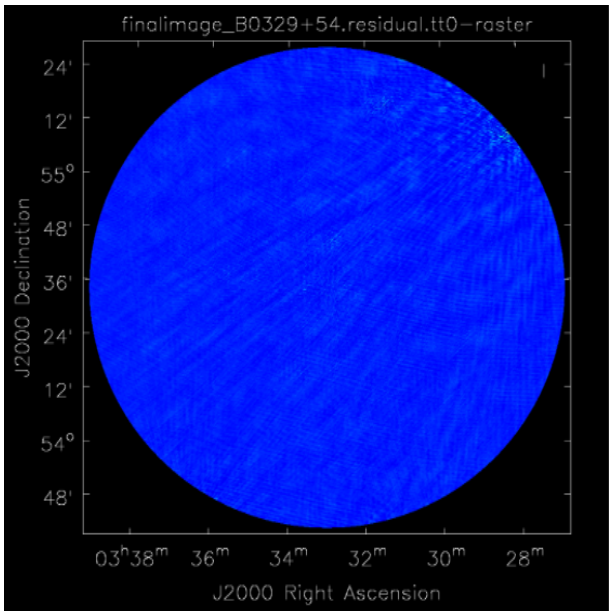


Figure 70: Residual image showing masking done

At this point, we iterate over the process of self calibrating and then running tclean, till we get a clean image with low residual image rms values.
On the completion of this process, we receive the following final image of B0329+54.

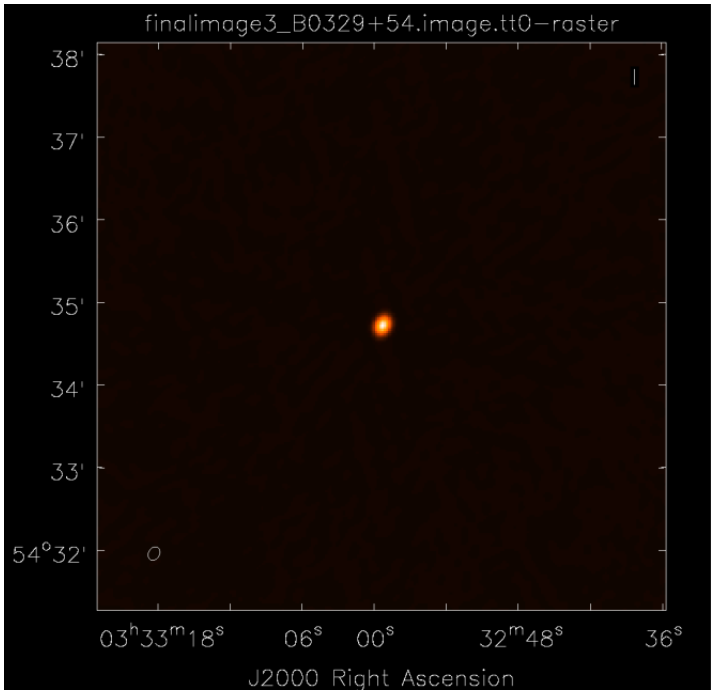


Figure 71: Final image obtained of B0329+54

References

[1] GMRT Tutorials Team. *Continuum Band 4 Tutorial*. Accessed: 2025-07-30. 2023. URL: https://gmrt-tutorials.readthedocs.io/en/latest/continuumband4.html.

[2] NRAO. *VLA CASA Flagging (CASA 6.5.4)*. Accessed: 2025-07-30. 2023. URL: https://casaguides.nrao.edu/index.php/VLA_CASA_Flagging-CASA6.5.4.

[3] *CASA task: tclean*. Accessed: July-30,2025. 2025. URL: https://casadocs.readthedocs.io/en/stable/api/tt/casatasks.imaging.tclean.html.

[4] F. Sun. *Introduction to CASA Imaging*. https://magnif.as.arizona.edu/~fsun/home/data/slides/02b_Intro_CASA_Imaging_AA-20221005.pdf. Accessed: 2025-07-30. 2022.

[5] ALMA Allegro Team. *Imaging in CASA – Allegro Workshop 2024*. https://www.alma-allegro.nl/wp-content/uploads/2024/10/Imaging_Allegro2024_p2.pdf. Accessed: 2025-07-30. 2024.