

Radio interferometría y síntesis de imágenes en astronomía

Departamento de Ingeniería en Informática

Proyecto - Parte 1

1 Objetivo

El objetivo de este laboratorio es diseñar e implementar la simulación radio-interferométrica, el gridding y la construcción de la *dirty image* de una colección de fuentes puntuales astronómicas.

2 Radio-interferometría astronómica

Un radio-interferómetro astronómico es un arreglo (*array*) de antenas que trabaja en conjunto para observar las ondas de radio que provienen de objetos en el cielo. Debido a que construir una antena de gran tamaño y de gran resolución es muy costoso e inviable, se prefiere usar un arreglo de antenas en donde la resolución θ está dada por la longitud de onda observada λ y la máxima separación entre las antenas D_{\max} . Esto es:

$$\theta = \frac{\lambda}{D_{\max}} = \frac{c}{\nu D_{\max}}, \quad (1)$$

donde c es la velocidad de la luz y ν es la frecuencia observada.

Para entender el funcionamiento de un radio-interferómetro es necesario simplificar el arreglo de antenas a un solo *baseline* (i, j) como se muestra en la Figura 1.

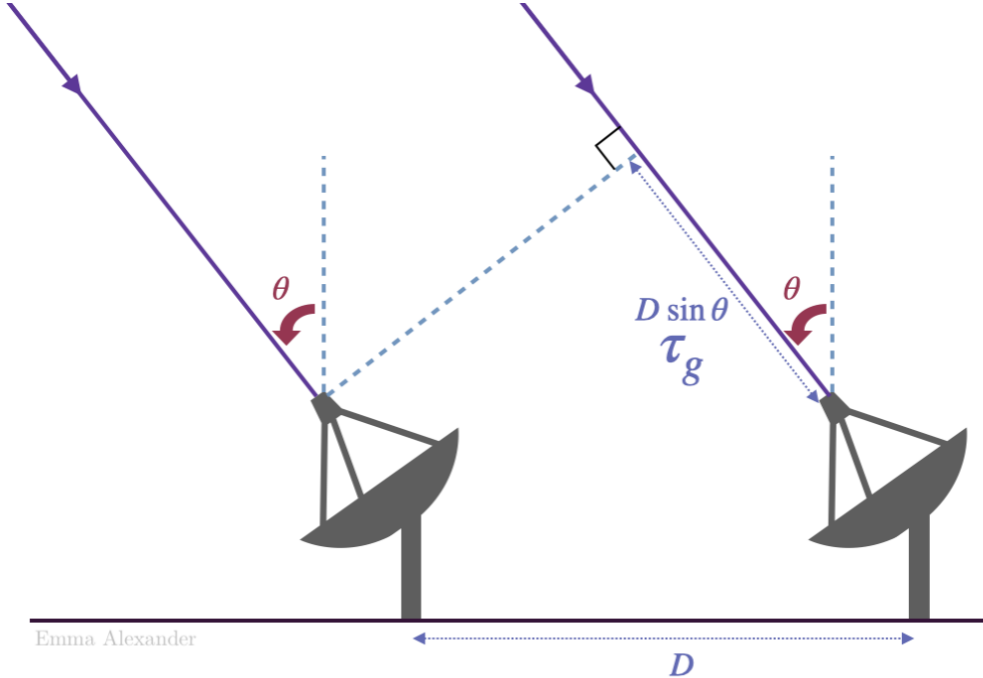


Figure 1: Par de antenas (*baseline*) a una distancia D observando el cielo. Créditos: Emma Alexander.

Cada par de antenas observa *visibilidades* en un tiempo dado. Una visibilidad se puede describir como la Ecuación 2. Es decir, como el promedio temporal de la correlación entre los campos eléctricos de las ondas de radio que llegan a cada par de antenas.

$$V_{ij} = \langle E(r_i, t) E^*(r_j, t) \rangle_t \quad (2)$$

Más tarde, el teorema de Van Cittert-Zernike (ver Ecuación 3) demostraría que una visibilidad equivale a tomar la transformada de Fourier de la imagen del cielo $I(l, m)$. Donde j representa $\sqrt{-1}$, l y m los cosenos directores, y (u, v, w) representan la separación de los pares de antenas en un plano Cartesiano proyectado hacia la dirección de observación en unidades de numero longitudes de onda.

$$V(u, v, w) = \int \frac{I(l, m)}{\sqrt{1 - l^2 - m^2}} e^{2\pi j(ul + vm + w(\sqrt{1 - l^2 - m^2} - 1))} dl dm \quad (3)$$

3 Parte 1: Simulación de cobertura uv

En clases, hemos visto que cada antena contiene una posición (x, y, z) que puede encontrarse ya sea en un plano ecuatorial o en un plano de referencia terrestre. En caso, de ser el ultimo caso, se necesita convertir las coordenadas siguiendo los siguientes pasos:

1. Convertir desde (x, y, z) a (East, North, Up)
2. Convertir de coordenadas ENU locales a Azimuth/Elevación local $(\mathcal{A}, \mathcal{E})$.
3. Convertir de coordenadas locales $(\mathcal{A}, \mathcal{E})$ a ecuatoriales (H, δ) o equivalentemente (X, Y, Z) .

Luego, basta con aplicar dos rotaciones, para obtener las coordenadas del plano uv , también conocido como plano de Fourier. Esto es descrito en la Ecuación 4, donde H_0 y δ_0 se refiere a la hora ángulo y la declinación de la fuente, respectivamente.

$$\begin{pmatrix} u \\ v \\ w \end{pmatrix} = \begin{pmatrix} \sin H_0 & \cos H_0 & 0 \\ -\sin \delta_0 \cos H_0 & \sin \delta_0 \sin H_0 & \cos \delta_0 \\ \cos \delta_0 \cos H_0 & -\cos \delta_0 \sin H_0 & \sin \delta_0 \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} \quad (4)$$

Cada estudiante recibirá una configuración de antenas distinta de la cual se tendrá que generar posiciones uvw para un rango de horas ángulo, muestreadas regularmente cada cierto tiempo y en una declinación específica. Además, se debe realizar un muestreo regular en frecuencia. Para ello, solo es necesario simular las coordenadas uvw en metros, y luego convertir las coordenadas a numero de longitudes de onda. Esto se realiza usando la siguientes ecuaciones:

$$\begin{aligned} u_\lambda &= u_m \times \frac{1}{\lambda} = u_m \times \frac{\nu}{c}, \\ v_\lambda &= v_m \times \frac{1}{\lambda} = v_m \times \frac{\nu}{c}, \\ w_\lambda &= w_m \times \frac{1}{\lambda} = w_m \times \frac{\nu}{c}. \end{aligned}$$

Considere que en este caso, λ y ν son vectores de longitudes de onda y de frecuencia, respectivamente.

Finalmente, en caso de que la configuración de antenas contenga las posiciones de un plano de referencia terrestre, se proveerá un script Python para su transformación a coordenadas ecuatoriales.

3.1 Fuente puntual

Si asumimos la imagen $I(l, m)$ como una fuente puntual, entonces

$$I(l, m) = S_0 \delta(l - l_0, m - m_0), \quad (5)$$

donde S_0 es la amplitud de la fuente, y (l_0, m_0) es su posición. Si aplicamos la transformada de Fourier usando la Ecuación 3, se tiene que

$$\begin{aligned} V(u, v, w) &= \int A(l, m) \frac{S_0 \delta(l - l_0, m - m_0)}{\sqrt{1 - l^2 - m^2}} e^{2\pi j(ul + vm + w(\sqrt{1 - l^2 - m^2} - 1))} dl dm \\ &= A(l_0, m_0) \frac{S_0}{\sqrt{1 - l_0^2 - m_0^2}} e^{2\pi j(ul_0 + vm_0 + w(\sqrt{1 - l_0^2 - m_0^2} - 1))}. \end{aligned}$$

Donde $A(l, m)$ es el *primary beam* o respuesta de la antena. En este trabajo el primary beam será modelado como una Gaussiana circular. Finalmente, esta última ecuación indica la transformada de Fourier analítica de una fuente puntual, y por lo tanto es posible obtener los valores de las visibilidades directamente usando las coordenadas uvw . Además para cada visibilidad, consideraremos una desviación estándar σ de valor 1. Entonces si definimos el peso de una visibilidad tal que $\omega = \frac{1}{\sigma^2}$, tenemos para cada una de las visibilidades un peso $\omega = 1$.

En este punto, se tienen los datos que la mayoría de los observatorios entregan a los radio astrónomos: Un dataset con *visibilidades continuas*, muestreadas de forma regular en el tiempo y en frecuencia.

4 Parte 2: Gridding

Hasta ahora, es sabido que las visibilidades generadas están muestreadas de forma regular en el tiempo y frecuencia, pero **no de forma regular en el espacio uvw de Fourier**. Si la imagen y su transformada de Fourier estuviesen muestrados en una grilla regular. Es decir, las visibilidades y los píxeles estuvieran equidistantemente espaciados en las direcciones u y v , y x e y , respectivamente, la distancia entre los píxeles de la imagen $I(x, y)$ sería $-\Delta x$ y Δy en unidades de radianes. Entonces, la distancia en los puntos en su transformada $V(u, v)$ sería:

$$\Delta u_\lambda = \frac{1}{N \Delta x_{\text{rad}}} \quad \Delta v_\lambda = \frac{1}{N \Delta y_{\text{rad}}} \quad (6)$$

Debido a que las coordenadas uv generadas no están equispaciadas, no es posible obtener la transformada rápida inversa de Fourier y obtener una imagen. La Figura 2 muestra un zoom de un plano uv junto con una grilla regular. Por lo tanto los puntos uv no pueden ser usados directamente para aplicar la inversa de Fourier y obtener la imagen sucia.

Note de aquí en adelante omitiremos la coordenada w ya que la consideraremos cercana a 0 para todas las muestras.

El proceso de *gridding* consiste en construir datos regularmente espaciados en el plano Fourier, a partir de datos astronómicos radio-interferométricos. Hay muchas formas de hacer esto, pero la más sencilla es sumar todos los valores de las visibilidades que se encuentran en una vecindad de cada punto de la grilla regular.

A continuación, se explica con detalle esta operación.

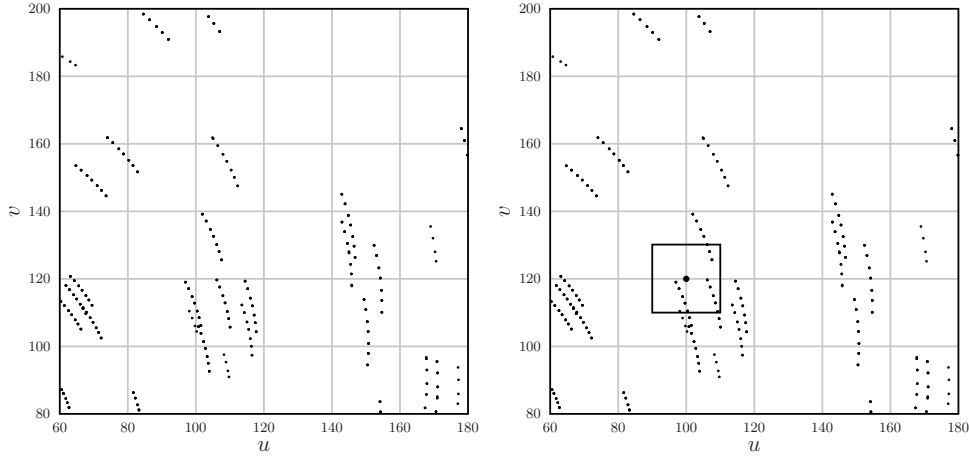


Figure 2: Zoom en el plano uv .

Los valores Δu y Δv se pueden considerar como la distancia entre dos celdas de la grilla, o la tasa de muestreo (sampling rate). Cada celda (cuadrado) en la Figura 2 tiene un centro (u_i, v_j) . Suponga además que la grilla regular es de tamaño $N \times N$. Luego, los centros de las celdas son:

$$(u_i, v_j) = (i\Delta u, j\Delta v) \quad i, j = -(N/2 - 1), -(N/2 - 2), \dots, -1, 0, 1, \dots, N/2 - 1$$

Esta relación implica que:

1. Los puntos (u_i, v_j) son coordenadas equiespaciadas en el plano uv y no corresponden a los índices de la matriz.
2. El plano uv está centrado en la grilla.
3. El eje u comienza en valores positivos a la izquierda de la grilla y va hacia los negativos hacia la derecha
4. El eje v comienza con valores negativos arriba de la grilla y va hacia los positivos hacia abajo.

El proceso de gridding consiste en acumular todos los valores de los puntos uv que caen en una celda, en su centro. En realidad hay muchas formas de realizar esta suma, pero lo más simple es la suma.

En la Figura 2 las posiciones en las intersecciones de las líneas son las posiciones en la grilla regular, y las posiciones marcadas con puntos son posiciones de las visibilidades continuas. Entonces, a partir de los valores en los puntos deseamos estimar el valor en las intersecciones. En la Figura 2 se muestra la vecindad del punto de la grilla regular $(100, 120)$. Todos los puntos uv que están dentro de esa vecindad cuadrada deben sumarse para formar el valor del punto en la grilla. Note que para este ejemplo $\Delta u_\lambda = \Delta v_\lambda = 20$.

Recordar que antes de comenzar el gridding, las coordenadas se deben transformar número de longitudes de onda. De ahora en adelante, se asume que ese proceso ya se realizó.

Sea V_G una matriz compleja de $N \times N$ y sea W_G una matriz real de $N \times N$ que acumula los pesos correspondientes. Para determinar a que posición de la matriz corresponde una visibilidad (u_k, v_k) basta calcular:

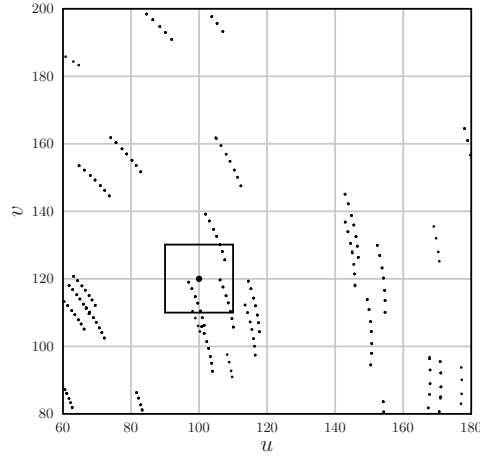


Figure 3: Ejemplo de vecindad que define los puntos que se suman para el punto de la grilla regular.

$$i_k = \left\lfloor \frac{u_{\lambda k}}{\Delta u_\lambda} \right\rfloor + \frac{N}{2}, \quad j_k = \left\lfloor \frac{v_{\lambda k}}{\Delta v_\lambda} \right\rfloor + \frac{N}{2}$$

donde el operador es el redondeo al entero más cercano. Por ejemplo, supongamos que $N = 8$, $\Delta u_\lambda = 0.5$, y que $u_{\lambda k} = -0.24$. Entonces $i_k = 4$. Si $u_{\lambda k} = 0.3$, $i = 5$.

Una vez determinado a que entrada de la matriz pertenece la visibilidad, acumulamos

$$\begin{aligned} V_G[i_k, j_k] &= V_G[i_k, j_k] + \omega_k \times V(u_k, v_k) \\ W_G[i_k, j_k] &= W_G[i_k, j_k] + \omega_k \end{aligned}$$

Al finalizar de gridear todas la visibilidades, se realiza la normalización con los pesos acumulados:

$$V_G[i, j] = V_G[i, j] / W_G[i, j]$$

5 Entregables

- Un jupyter notebook o Collab notebook
- Archivos python adicionales en caso de usar

Fecha de entrega (hard-deadline):
30 de octubre, 2024 antes de las 23:59 hrs.