

Scattering en dos dimensiones: evaluación con modelos canónicos

Ramiro M. Irastorza¹

¹*Instituto de Física de Líquidos y Sistemas Biológicos CONICET - CCT La Plata.
La Plata, Buenos Aires, Argentina*

¹*Instituto de Ingeniería y Agronomía, UNAJ.
Florencio Varela, Buenos Aires, Argentina*

October 2, 2018

1 Introducción

En este artículo estudiaremos los modelos canónicos de scattering de microondas. Se explicará el uso y los algoritmos del software desarrollado en python para tal fin. En los casos donde sea posible se implementará la solución analítica y en los que no, una o varias soluciones numéricas. En particular la solución numérica que se aplicará es la de Richmond [1]. Este es un método de momentos con la suficiente aproximación como para ser utilizada en los algoritmos de reconstrucción.

La excitación que utilizaremos es una línea de corriente que sale del plano x-y (eje z). La transmisión generada será entonces Transversal Magnética (TM) y cada vez que aparezca E significa que es la componente en el eje z del campo eléctrico (E_z). Si la línea de corriente es de intensidad I , el campo incidente es conocido (ver pags. 224 y 225 del libro de Harrington [2]) y se puede calcular como:

$$E^{\text{inc}} = -\frac{k_b^2 I}{4\omega_0 \epsilon_b} H_0^{(2)}(k_b |\rho - \rho'|) \quad (1)$$

k_b y ϵ_b son el número de onda y la permitividad compleja del medio de acoplamiento¹. ρ y ρ' son la posición donde se mide el campo eléctrico y de la fuente, respectivamente. Para esto, podemos utilizar la función:

```
Ezinc[nx,ny] = EINC_LINESOURCE([xt,yt], [x[nx],y[ny]],ACOPLANTE)
```

¹En este texto utilizaremos ϵ para permitividad y ε para permitividad relativa, entonces $\varepsilon = \epsilon/\epsilon_0$

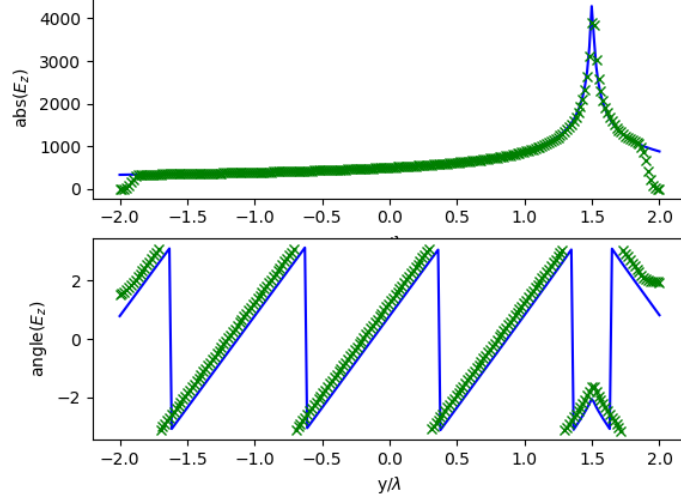


Figure 1: Resultados del script test0.py. Fijando $x = 0$ y variando y , módulo y fase. Transmisor cero ($T_x = 0$). En cruz el cálculo de FDTD (utilizando Meep).

aquí `ACOPLANTE` es un objeto que contiene los parámetros del medio acoplante. (x_t, y_t) son las coordenadas de la antena transmisora y (x, y) son las coordenadas donde se quiere sensar el campo (en este caso hemos discretizado las posiciones). Los valores por defecto del acoplante son el vacío. Figuras 1 y 2 se obtuvieron con el script `test0.py` utilizando el transmisor cero ($T_x = 0$). Se colocaron 8 antenas transmisoras en una circunferencia de radio $1,5\lambda$. Esta función está implementada en `funciones_canonicos.py`.

2 Línea de corriente excitando cilindro dieléctrico

Desarrollaremos un cilindro centrado en el origen de cierta permitividad relativa ($\varepsilon = \varepsilon_r - j\frac{\sigma}{\epsilon_0\omega}$) para una frecuencia fija (f_0).

2.1 Teoría

Para la solución analítica utilizamos los resultados de Arslanagic y otros [3]. La función que utilizaremos aquí es:

`EZ_CILINDRO_LINESOURCE_MATRIZ(eps,CIL,ACOP,TRANS,Tx,deltaX)`

aquí `eps` es un archivo de extensión `.h5` con el mapa de permitividad (generado con Meep, ver más adelante). `CIL`, `ACOP`, y `TRANS` son clases que contienen varias características de los medios que mencionan sus nombres.

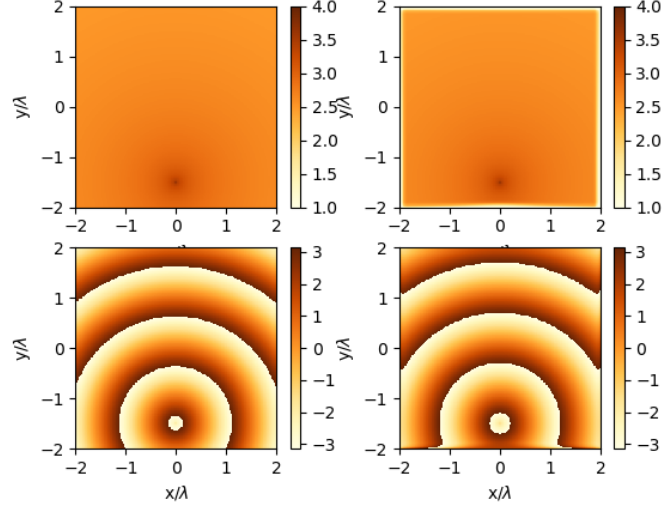


Figure 2: Resultados del script test0.py. Mapa de logaritmo en base diez de la intensidad de campo (módulo) y fase en todo el dominio. Transmisor cero ($T_x = 0$).

Tx es el transmisor y deltaX la resolución espacial del mapa eps.

Las Figs. 3-5 muestran los resultados teóricos y numéricos del scattering producido por un cilindro en (0,0) con las siguientes propiedades: $\lambda = c/1.0$ GHz, $r = \lambda/2$, $\varepsilon_r = 2.2$, y $\sigma = 0.02$ Sm⁻¹. El medio de acoplamiento tiene las siguientes propiedades: $\varepsilon_{rb} = 1.0$, y $\sigma_b = 0.01$ Sm⁻¹. El arreglo de antenas tiene un radio de $\rho_s = 1.5\lambda$ y son ocho antenas ($S = 8$).

2.2 Simulación

Discretización con método de los momentos de Richmond [1].

La Ec. (9) del trabajo de Richmond [1] brinda la ecuación integral que permite calcular el campo total en un sistema que incide un campo E_{inc} :

$$E(x, y) + \frac{jk_b}{4} \iint (\varepsilon - \varepsilon_b) E(x', y') H_0^{(2)}(k_b |\rho - \rho'|) dx' dy' = E^{\text{inc}}(x, y) \quad (2)$$

Si se divide en celdas el dominio y se supone constante en cada celda la permitividad y el campo eléctrico entonces se puede escribir la siguiente ecuación:

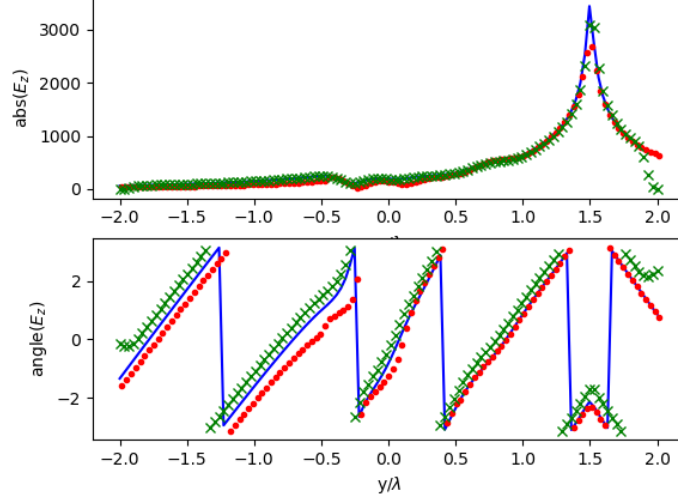


Figure 3: Resultados del script test1.py. Fijando $x = 0$ y variando y , módulo y fase. Transmisor cero ($T_x = 0$). En cruz el cálculo de FDTD (utilizando Meep).

$$E_m + \frac{jk_b}{4} \sum_{n=1}^N (\varepsilon_{r_n} - \varepsilon_b) E_n \iint_{\text{celda } n} H_0^{(2)}(k_b |\rho - \rho'|) dx' dy' = E_m^{\text{inc}} \quad (3)$$

La solución de la integral doble de la función de Hankel de una celda circular de radio “a” tiene solución analítica y esta solución se utiliza para obtener los coeficientes C_{mn} de tal manera de escribir la Ec. (3) como:

$$\sum_{n=1}^N C_{mn} E_n = E_m^{\text{inc}} \text{ con } m = 1, \dots, N \quad (4)$$

y así construir un sistema lineal de la forma $Ax = B$ donde las incógnitas son los valores de los campos eléctricos en cada celda. Todo esto está implementado en la función:

```
Ezinc[nx,ny] = numRichmond(eps, CIL,ACOP,TRANS,Tx,deltaX)
```

Discretización con método diferencias finitas en dominio de tiempo (de sus siglas en inglés FDTD) [4].

En este caso, la solución está dada por el método FDTD implementado en el software Meep. Este software también construye la matriz que representa la propiedad dieléctrica (particularmente la permitividad ε_r) del

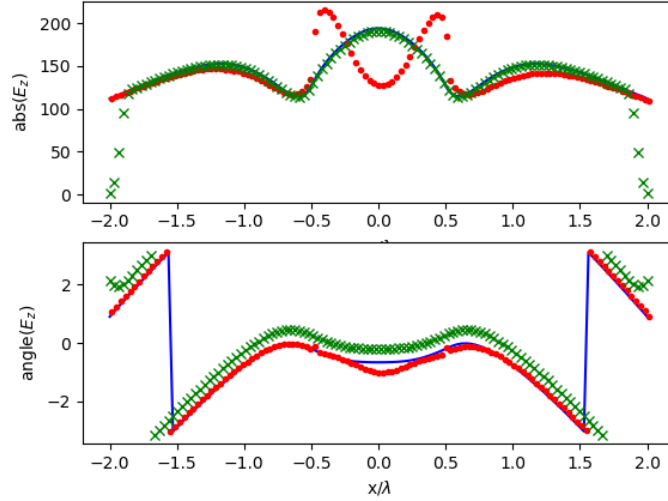


Figure 4: Resultados del script test1.py. Fijando $y = 0$ y variando x , módulo y fase. Transmisor cero ($T_x = 0$). En cruz el cálculo de FDTD (utilizando Meep).

medio en un archivo del tipo `nombre-eps-000000.00.h5`, donde “nombre” es el nombre del archivo puesto por el usuario. Se desarrolló una función que construye un archivo `.ctl`:

```
flag = generaCTL()
```

a esta función se le deberá indicar el “nombre” del archivo (entre otras entradas) y luego se deberá ejecutar en consola con Meep de la siguiente forma:

```
>> meep no-bone?=false nombre.ctl
```

2.3 Cilindro multi-capa

2.4 Elipse

2.5 Elipse multi-capa

3 Método de inversión basado en aproximaciones tipo Born

3.1 Aproximación de la integral

Todo lo desarrollado aquí está basado en el libro de Pastorino [5] (págs. 108-117). Antes de comenzar con este método vamos a estudiar la aproximación

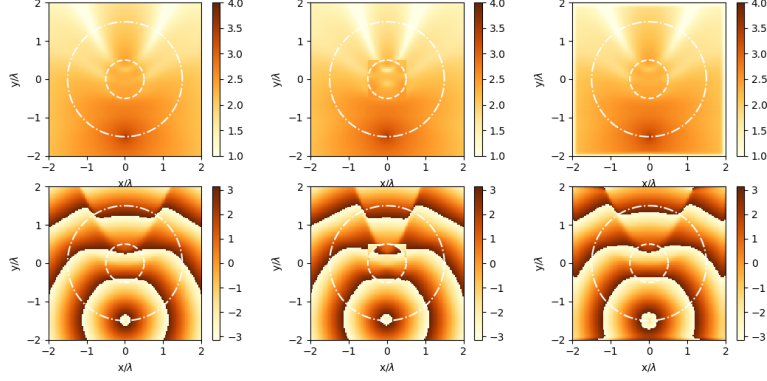


Figure 5: Resultados del script `test1.py`. Mapa de logaritmo en base diez de la intensidad de campo (módulo) y fase en todo el dominio. Transmisor cero ($T_x = 0$). Columna 1, 2, y 3 corresponden a la teoría, el método numérico de Richmond, y diferencias finitas en dominio de tiempo, respectivamente.

de Born de primer orden, y algunas formas de resolver numéricamente las integrales involucradas. Según esta aproximación, la Ec. (3.3.8) de [5] queda de la siguiente forma:

$$E(x, y) = E^{\text{inc}}(x, y) + \frac{jk_b}{4} \iint (\varepsilon_r - \varepsilon_b) E^{\text{inc}}(x', y') H_0^{(2)}(k_b |\rho - \rho'|) dx' dy' \quad (5)$$

Para una celda n , considerando la permitividad constante ($(\varepsilon_r - \varepsilon_b) = \text{constante en la celda } n$), entonces la parte derecha (integral) de la Ec. 5 queda como:

$$b_{mn} = \frac{jk_b}{4} \iint_{\text{celda } n} E^{\text{inc}}(x', y') H_0^{(2)}(k_b |\rho - \rho'|) dx' dy' \quad (6)$$

donde:

$$|\rho - \rho'| = \sqrt{(x' - x_m)^2 + (y' - y_m)^2}. \quad (7)$$

La expresión Ec. 6 es similar a la Ec. 3. En la Fig. 6 comparamos cuatro maneras de aproximarla, todo está implementado en el script `test2.py`:

1. Suponiendo que la celda es circular y el campo es constante en la celda (esto tiene solución analítica)
2. La celda es cuadrada (tal como lo hacemos en FDTD) y el campo es constante en la celda
3. Celda cuadrada con campo variable pero solución teórica incidente de línea de corriente $E^{\text{inc}}(x', y')$ (aproximación de Born)

4. El campo $E(x', y')$ (solución teórica de [3]). Notar aquí que el campo es el total, por lo tanto no es una aproximación del tipo Born.

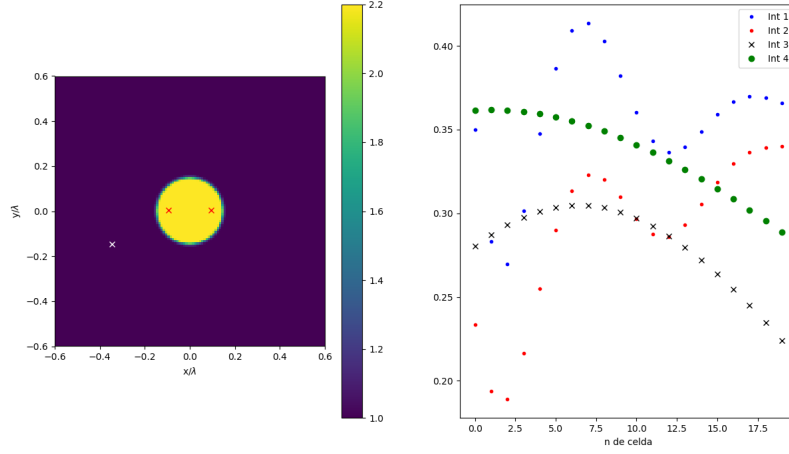


Figure 6: Integral b_{mn} .

3.2 Estimación del campo disperso

Aquí comparamos el campo disperso (E_s) medido a partir de una simulación en Meep y el calculado por medio de la Ec. (5.12.4) de [5]. Repetimos aquí la expresión:

$$[\mathbf{B}]\tau = \mathbf{e}^s. \quad (8)$$

Tomamos aquí como ejemplo la misma elipse del libro pero desplazada y con la misma cantidad de transmisores ($S = 8$) y $M = 25$ receptores en el mismo sector angular dispuesto en el libro. Para verificar que el cómputo está bien hecho, en el programa `test5.py` calculamos $[\mathbf{B}]\tau$ como si conociéramos τ . En la Fig. 7 se muestra un ejemplo para $\varepsilon_r = 1,2$ y se comparan E_s (calculado con Meep) y \mathbf{e}^s .

3.3 Problema inverso

Para realizar el problema inverso se debe invertir la matriz \mathbf{B} la Ec. 8, pero esta matriz no es cuadrada y por eso utilizamos la descomposición SVD comentada en pag. 87 de [5] para calcular una pseudo inversa (o matriz de Moore-Penrose). Primero, para que el cómputo no sea tan lento, almacenamos los coeficientes de la matriz \mathbf{B} que son iguales para cualquier dispersor que se quiera reconstruir porque no depende de éste. El script

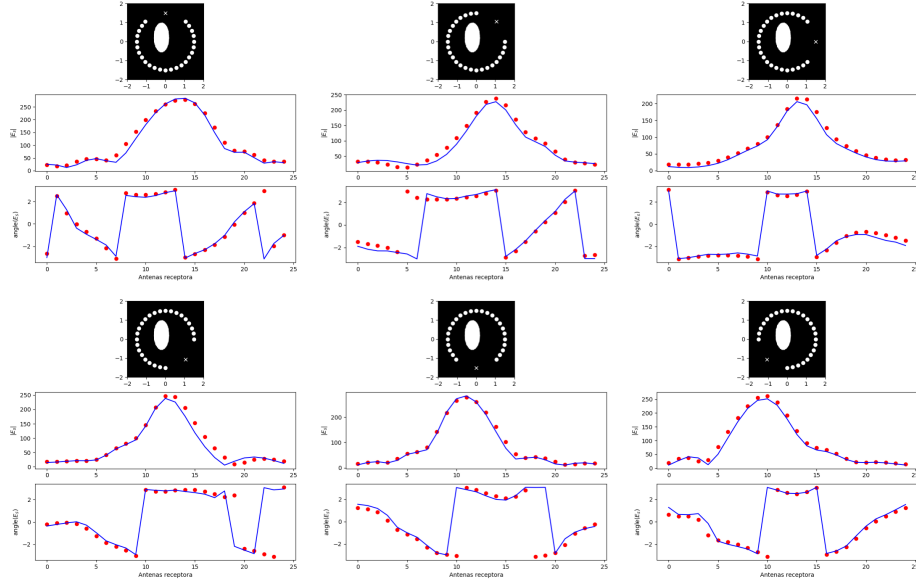


Figure 7: Ejemplo de algunas antenas utilizando $\varepsilon_r = 1,2$. Se grafica módulo y fase del campo disperso. En las figuras en blanco y negro se muestra la geometría y el arreglo de antenas receptoras (en círculo blanco) y transmisora (cruz blanca).

`test4_fabricaB.py` construye la matriz \mathbf{B} para cada transmisor T_{xi} denominándolo `B_Txi.npy` y luego los utilizo mediante el script `test6.py` para estimar la imagen. En la Fig. 8 se pueden ver algunos resultados variando la permitividad de la elipse. Mientras sea más diferente el dispersor peor es la estimación. El script `test6.py`, utiliza la función `numpy.linalg.pinv` que tiene como argumento la matriz \mathbf{B} y el menor valor singular a partir del cual se trunca la descomposición. Recordemos que en este caso la matriz \mathbf{B} es de 200×841 , el número de filas es 200 que surge de 8 (transmisores) $\times 25$ (receptores) y el número de columnas surge de 29×29 (número de celdas que divido el dominio de interés, es decir, la zona de búsqueda). Es importante destacar que aquí no se sumó ruido.

Si le sumamos ruido normal con media 0 y varianza para obtener una relación señal a ruido (SNR) de 10, 20, y 30 dB se obtienen los resultados utilizando la elipse con permitividad relativa 1.2 de la Fig. 9. Está implementado en el script `test7.py`.

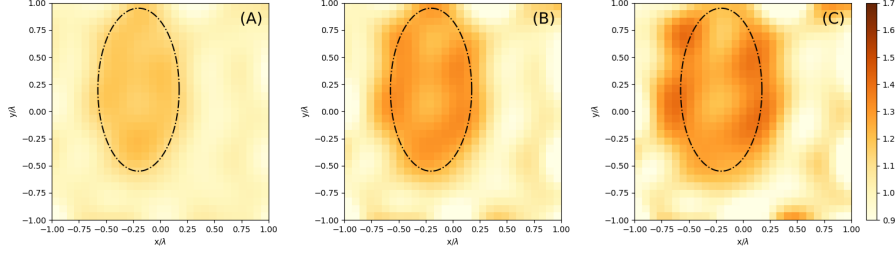


Figure 8: Ejemplo de estimación utilizando pseudo inversa calculada con descomposición SVD en el sistema de la Fig.7 cambiando la permitividad de la elipse. (A) $\varepsilon_r = 1,2$, (B) $\varepsilon_r = 1,4$, y (C) $\varepsilon_r = 1,6$. En línea punteada negra se grafica la elipse.

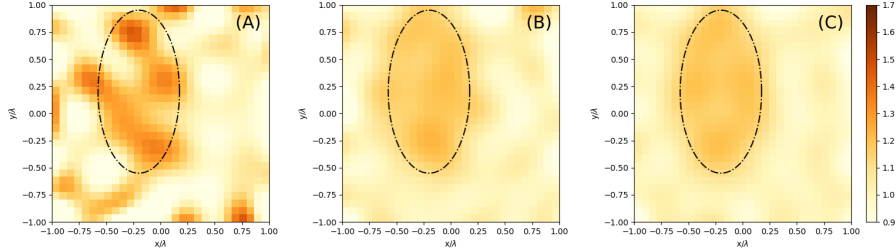


Figure 9: Ejemplo de estimación utilizando pseudo inversa calculada con descomposición SVD en el sistema de la Fig.7 cambiando SNR: (A) 10, (B) 20, y (C) 30 dB. La elipse tiene permitividad relativa $\varepsilon_r = 1,2$.

References

- [1] Jack Richmond. Scattering by a dielectric cylinder of arbitrary cross section shape. *IEEE Transactions on Antennas and Propagation*, 13(3):334–341, 1965.
- [2] Roger F Harrington. Time-harmonic electromagnetic fields. 1961.
- [3] Samel Arslanagic and Olav Breinbjerg. Electric-line-source illumination of a circular cylinder of lossless double-negative material: an investigation of near field, directivity, and radiation resistance. *IEEE Antennas and Propagation Magazine*, 48(3):38–54, 2006.
- [4] Ardavan F Oskooi, David Roundy, Mihai Ibanescu, Peter Bermel, John D Joannopoulos, and Steven G Johnson. MEEP: A flexible free-software package for electromagnetic simulations by the FDTD method. *Computer Physics Communications*, 181(3):687–702, 2010.

- [5] Matteo Pastorino. *Microwave imaging*, volume 208. John Wiley & Sons, 2010.