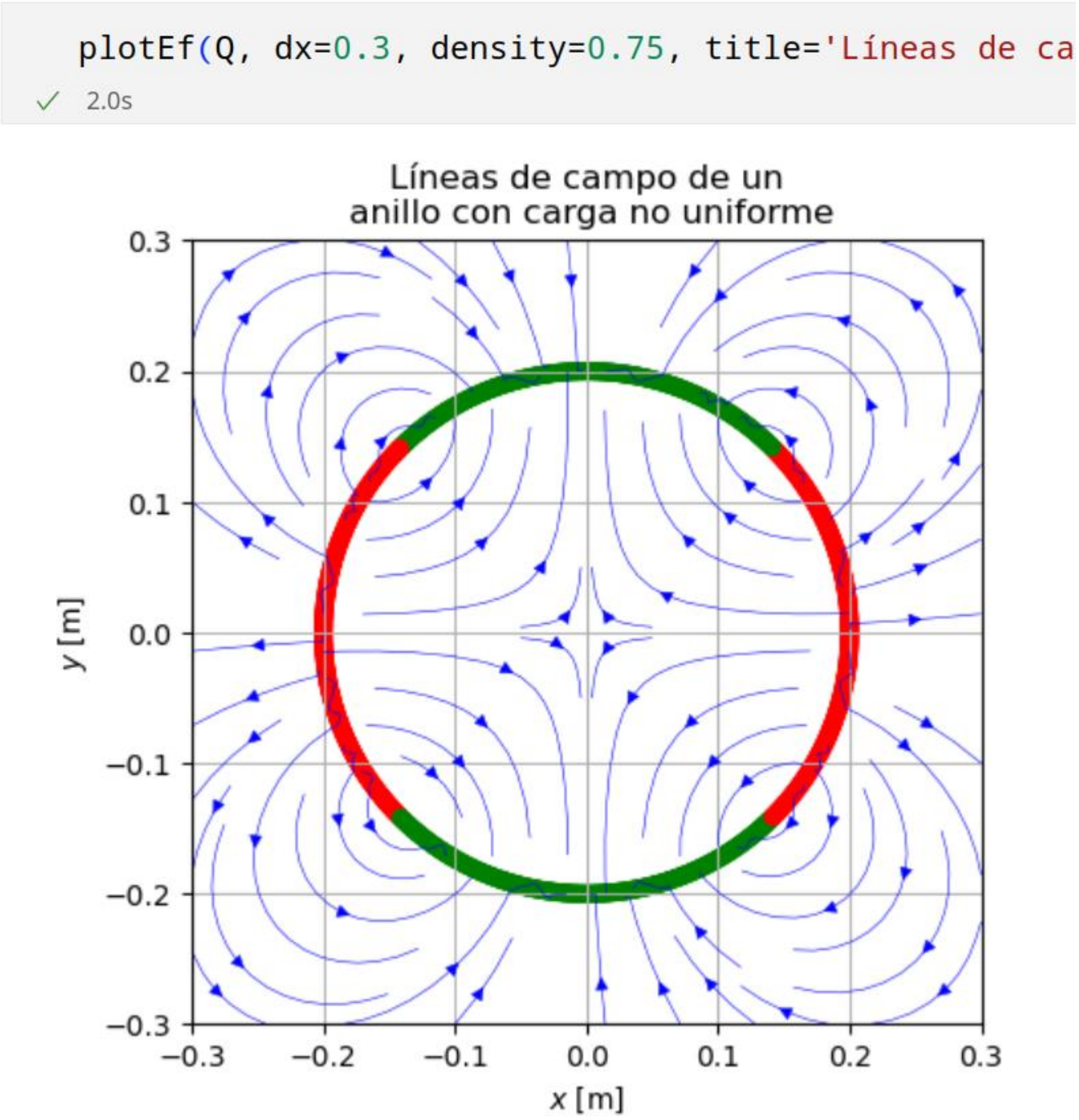
Ampliando el alcance de lo aprendido en electrostática usando Python

E. Palazzo – UTN Facultad Regional Avellaneda
epalazzo@fra.utn.edu.ar

V. A. Bettachini – UNLAM

M. A. Real – INCALIN UNSAM

https://frautn.github.io/F2-electromagnetismo/



UTN AVELLANEDA
UDB Física

Electromagnetismo usando Python

Material para realizar algunas prácticas computacionales que complementan el trabajo en clase de la unidad sobre electromagnetismo de la materia Física 2, en la Facultad Regional Avellaneda de la Universidad Tecnológica Nacional.

Actividades

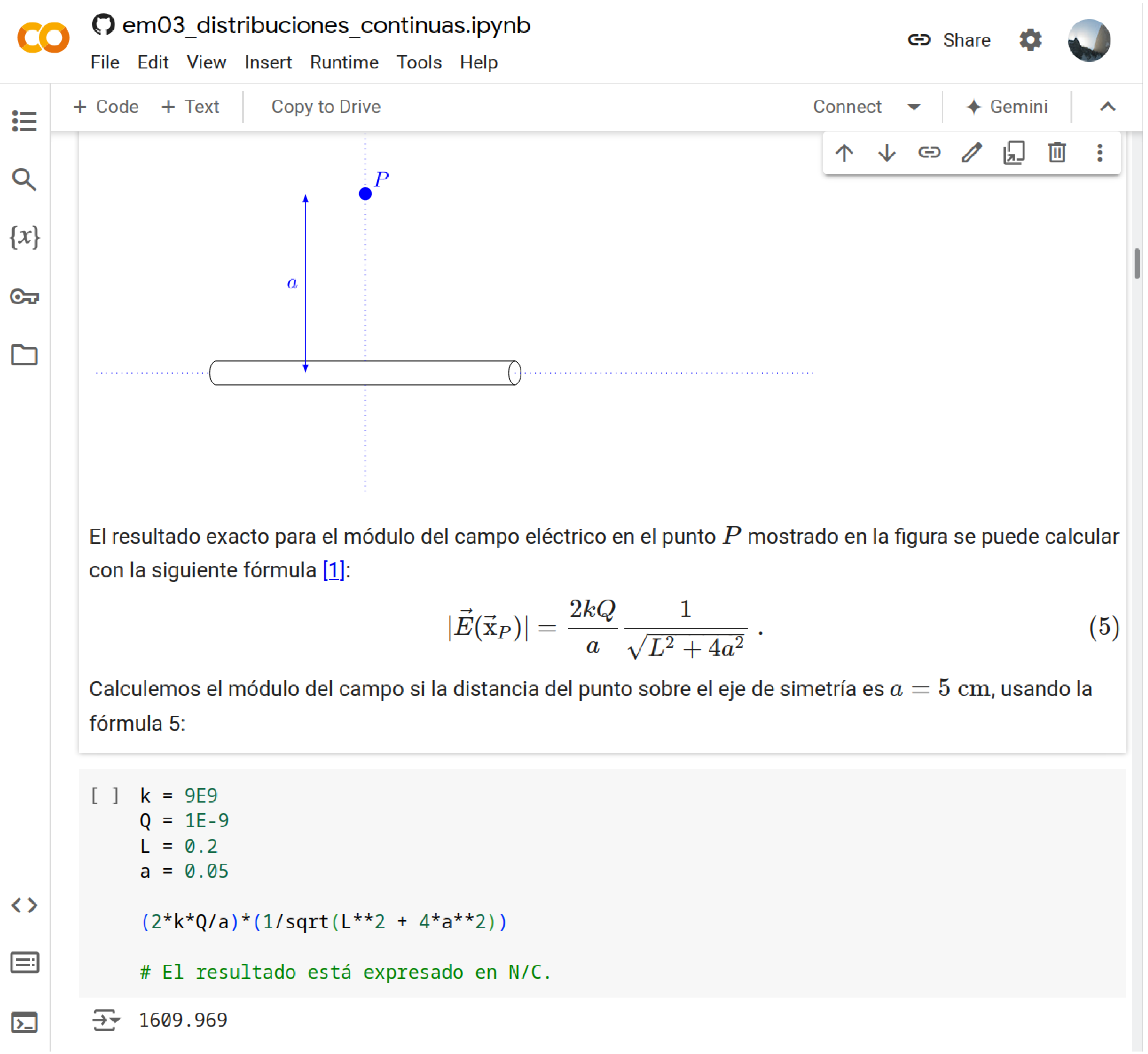
Módulo	Contenido	Enlace
1	Campo eléctrico de cargas puntuales	Abrir en Colab
2	Potencial eléctrico	Abrir en Colab
3	Distribuciones de carga contínuas - parte 1: cálculos numéricos sumando cargas puntuales	Abrir en Colab
4	Distribuciones de carga contínuas - parte 2: cálculos simbólicos	Abrir en Colab
5	Campos y equipotenciales en presencia de conductores	Abrir en Colab
6	Trabajo final de electrostática: plantilla para la entrega de los ejercicios propuestos	Abrir en Colab

Objetivo:

Asistir a los estudiantes durante la adquisición de los conceptos relacionados a Campo y Potencial Electrostático.

Metodología:

- Actividades divididas en módulos.
- Reemplazar el trabajo de pizarrón y papel por clases enteramente realizadas en Python.
- Reemplazar exposiciones con experimentación por parte de los estudiantes.



- **Cuadernos Jupyter**

  Teoría, comentarios y ejemplos intercalados con el código.

- **Google Colab**

  Accesible desde cualquier dispositivo, libre y gratuito.

  Solo se necesita un navegador.

em01_campo_electrico.ipynb

File Edit View Insert Runtime Tools Help

+ Code + Text Copy to Drive Connect Gemini Share

Física 2
2024
Electromagnetismo
Material para trabajo en clase

2024 Edgardo Palazzo (epalazzo@fra.utn.edu.ar)

Campo Eléctrico

Objetivos

- Familiarizarse con la metodología de trabajo a través de ejemplos sencillos.
- Repasar el cálculo de campos eléctricos y el análisis de su dirección y sentido.
- Visualizar líneas de campo eléctrico de distintas distribuciones de cargas puntuales.

Tabla de contenidos

em01_campo_electrico.ipynb

File Edit View Insert Runtime Tools Help

+ Code + Text Copy to Drive Connect Gemini Share

1. Campo de una carga

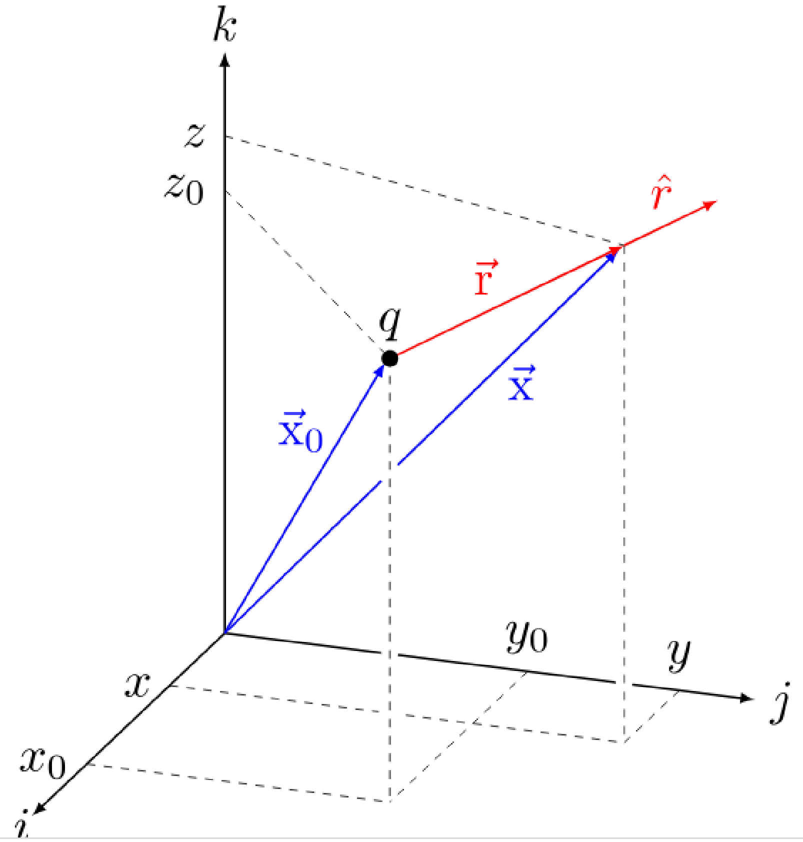
El campo eléctrico producido por una carga puntual q ubicada en la posición $\vec{x}_0$ puede escribirse como:

$$\vec{E}(\vec{x}) = \frac{1}{4\pi\varepsilon_0}\frac{q}{|\vec{r}|^2}\hat{\mathbf{r}}\,, \tag{1}$$

donde

- $|\vec{r}|$ es la distancia entre la posición de la partícula cargada ($\vec{x}_0$) y la posición donde se quiere obtener el vector de campo eléctrico ($\vec{x}$)
- $\hat{r}$ es el vector unitario en la dirección desde $\vec{x}_0$ hacia $\vec{x}$

según se muestra en la siguiente figura.



em01_campo_electrico.ipynb

File Edit View Insert Runtime Tools Help

Share

\+ Code + Text Copy to Drive Connect Gemini

$$\vec{E}(\vec{x}) = \frac{1}{4\pi\varepsilon_0}\frac{q}{|\vec{r}|^3}\vec{r}\,. \qquad (2)$$

Para escribir un código que calcule el campo eléctrico es conveniente utilizar una expresión con las coordenadas del punto campo y las coordenadas de la posición de la carga explícitas. El código que utilizaremos reproduce la siguiente expresión:

$$\vec{E}(x,y,z) = \frac{1}{4\pi\varepsilon_0}\frac{q}{((x-x_0)^2+(y-y_0)^2+(z-z_0)^2)^{3/2}}\Big((x-x_0)\hat{\mathbf{i}}+(y-y_0)\hat{\mathbf{j}}+(z-z_0)\hat{\mathbf{k}}\Big)\,. \qquad (3)$$

A continuación de define una función que calcula las componentes del campo eléctrico en la posición (x,y,z) cuando se tiene una carga de $1\ \mathrm{nC}$ en el origen:

```
[ ] def E1(x,y,z):
        """Calcula las componentes del campo eléctrico en N/C.
        Ingresar valores de x,y,z en metros y q en coulomb.
        """
        k = 9E9   #Constante de Coulomb en las unidades correspondientes.

        q = 1E-9
        x0 = 0
        y0 = 0
        z0 = 0

        # El denominador en la ec. 3:
        r32 = ((x - x0)**2 + (y - y0)**2 + (z - z0)**2)**(3/2)

        # Cada una de las componentes de la ec. 3:
        Ei = k * q * (x - x0) / r32
        Ej = k * q * (y - y0) / r32
        Ek = k * q * (z - z0) / r32

        return Ei, Ej, Ek
```

Si queremos el vector campo eléctrico en alguna posición, simplemente se ingresan los valores de x, y, z (el resultado está expresado en N/C):

```
[ ] E1(-1,0,0)

(-9.000, 0.000, 0.000)
```

Biblioteca con funciones específicas para este curso. (No es un curso de programación)

```
from frautnEM.puntuales import Ef
```

A continuación se define una configuración con 3 cargas en el plano xy: $q_1 = 5\,\mathrm{nC}$ en $(0,0,0)$, $q_2 = -3\,\mathrm{nC}$ en $(-2,0,0)$ y $q_3 = 7\,\mathrm{nC}$ en $(2,3,0)$, y se calcula el campo producido en distintas posiciones.

Estudiantes: solo deben familiarizarse con el formato de esta lista de cargas.

```
[ ] Q = [
        [  5E-9,  0, 0, 0],
        [ -3E-9, -2, 0, 0],
        [  7E-9,  2, 3, 0],
    ]

    # Vector de campo eléctrico en la posición (1,1,0) generado
    # por la distribución de cargas Q:

    Ef(1,1,0,Q)

    # (El resultado está expresado en N/C)
```

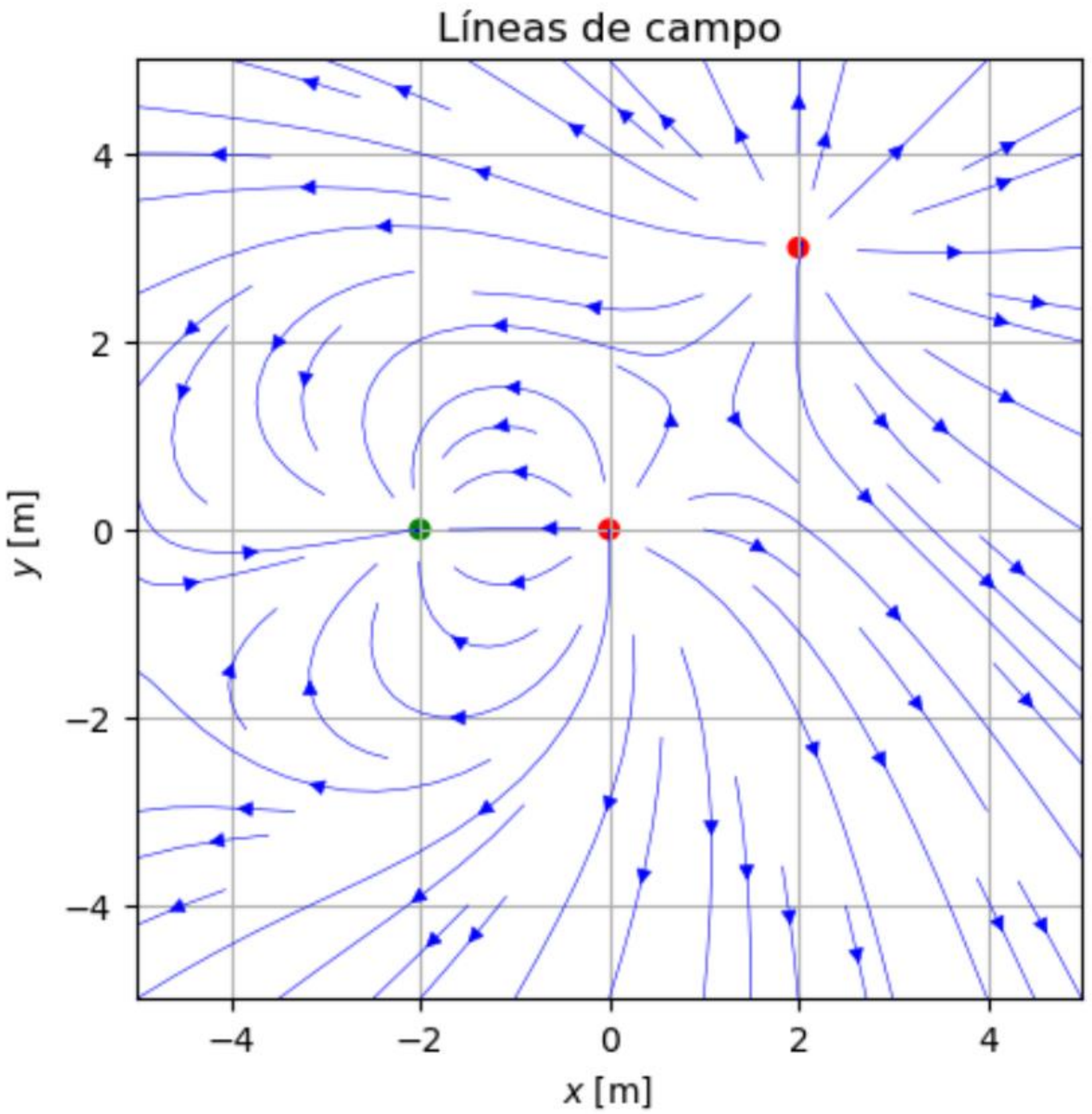
La función para calcular el campo eléctrico tiene como variables a x,y,z y la lista de cargas.

Las componentes del campo en la posición seleccionada.

```
(7.714, 3.786, 0.000)
```

Veamos las líneas de campo de esta distribución usando nuevamente la función *plotEf(Q)*.

```
[ ] plotEf(Q)
```



Flexibilidad

Con funciones muy sencillas de usar se trabajan infinidad de problemas.

Estudiantes: pueden experimentar modificando las cargas (valores, posiciones y cantidad de cargas).

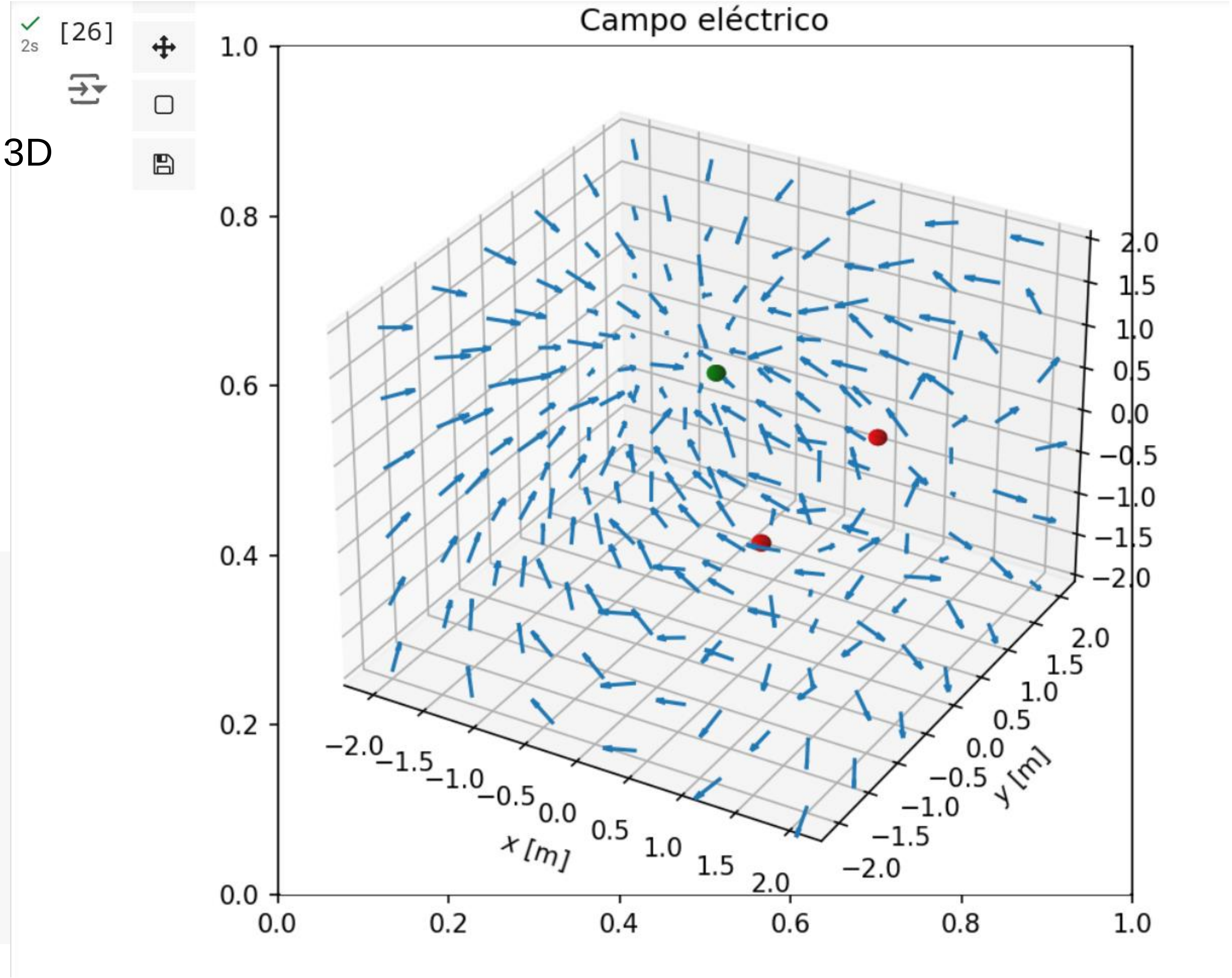
```
[ ]  Q = [
        [  5E-9,  0, 0, 0],
        [ -3E-9, -2, 0, 0],
        [  7E-9,  2, 3, 0],
     ]
```

Salir del plano

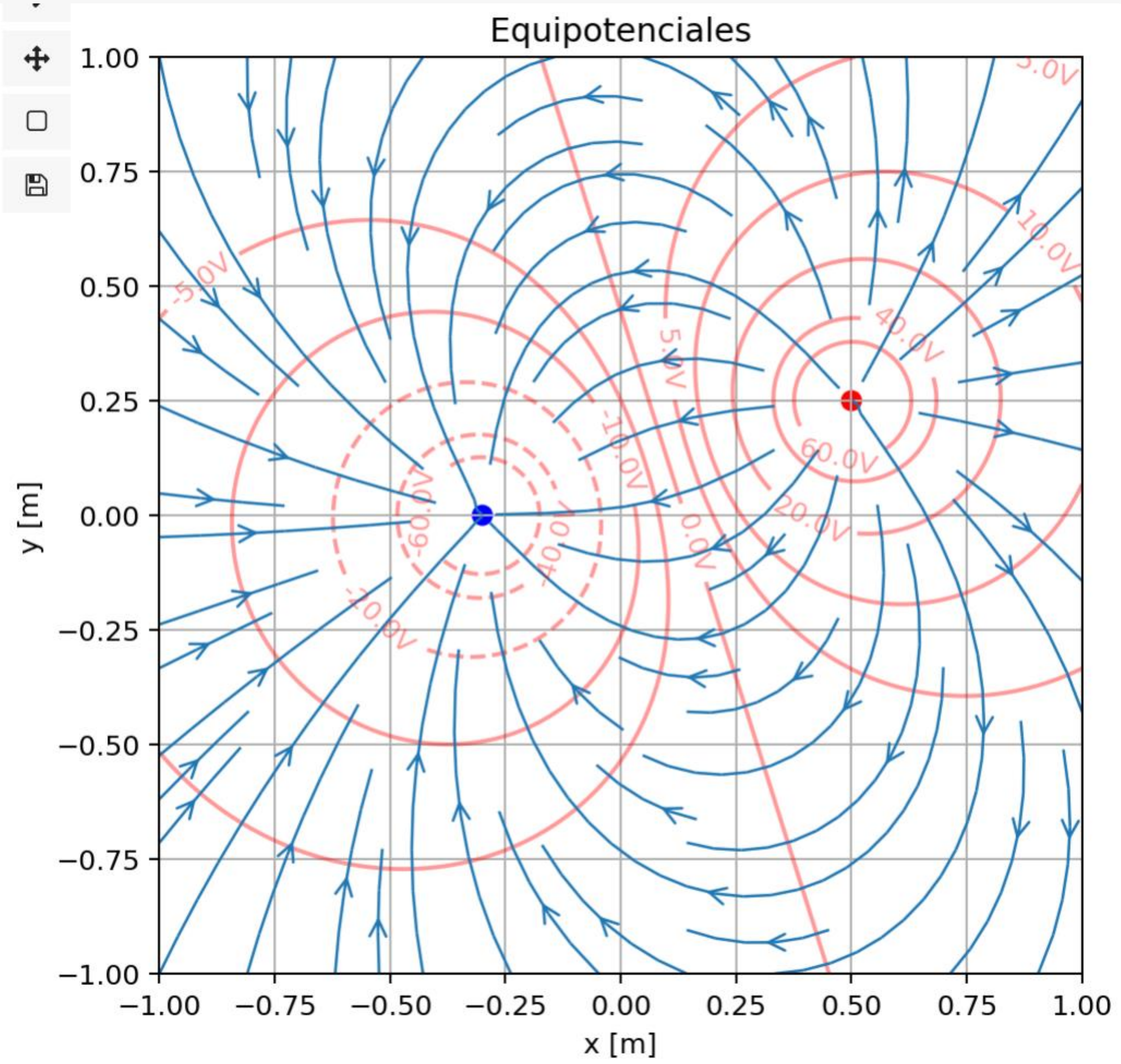
Visualizar los vectores en 3D con gráficos interactivos.

Las cargas pueden ubicarse en cualquier punto del espacio sin dificultad, algo muy complicado en el pizarrón o en el papel.

```
[26] Q = [
        [1E-9,1,1,0],
        [1E-9,1,-1,0],
       # [1E-9,-1,1,0],
       # [1E-9,-1,-1,0],
        [-3E-9,0,0,1],
    ]

    plotEfvector3d(Q, dx=2, w=6, figsize=(6,6))
```

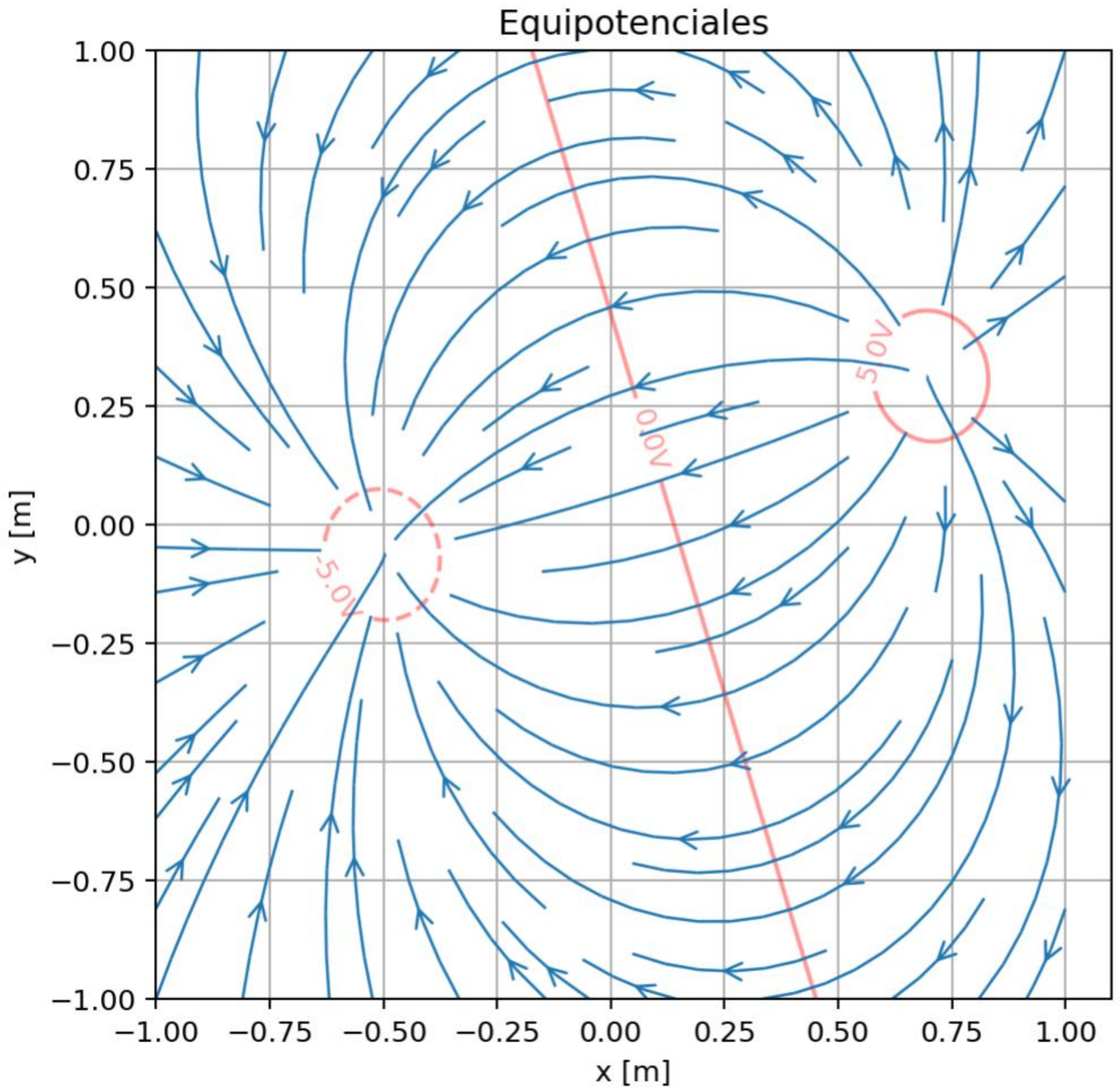


```
[3] Q = [
        [-1E-9,-0.3,0,0],
        [1E-9,0.5,0.25,0],
    ]

    niveles = [-60, -40, -20, -10, -5, 0, 5, 10, 20, 40, 60]

    equipotencialesPuntuales(Q, niveles=niveles, EF=True, z = 0)
```



Superficies equipotenciales

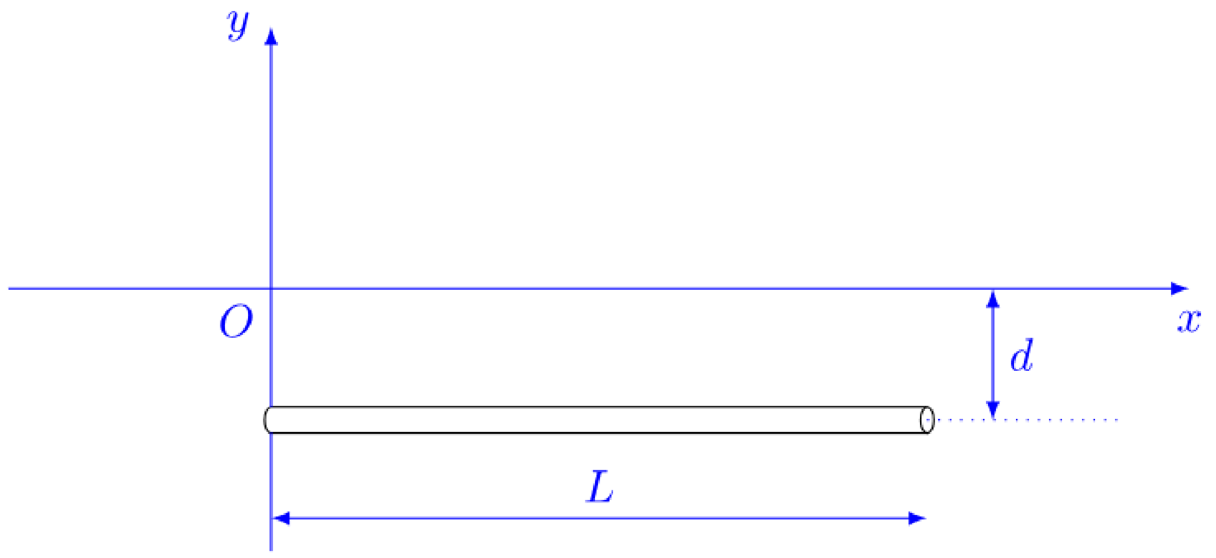
z = 0.7



Trabajo final

Ejercicio 1:

Considerar un hilo de longitud $L = 15\,\mathrm{cm}$, con carga total de $40\,\mathrm{nC}$ distribuida uniformemente, ubicado como se muestra en la figura, siendo $d = 3\,\mathrm{cm}$.



a) Encontrar una posición aproximada donde ubicar una carga puntual de $2\,\mu\mathrm{C}$, para que el campo eléctrico total en el origen sea horizontal y apunte hacia la izquierda ($-x$).

```
# 1) Escribir la lista de pequeñas cargas que conforman el hilo.

# Si fueran 4 cargas, el formato es el siguiente:

# Q = [
#     [q1, x1, y1, z1],
#     [q2, x2, y2, z2],
#     [q3, x3, y3, z3],
#     [q4, x4, y4, z4],
# ]
```

```
Ef(0,0,0,Q1)
```

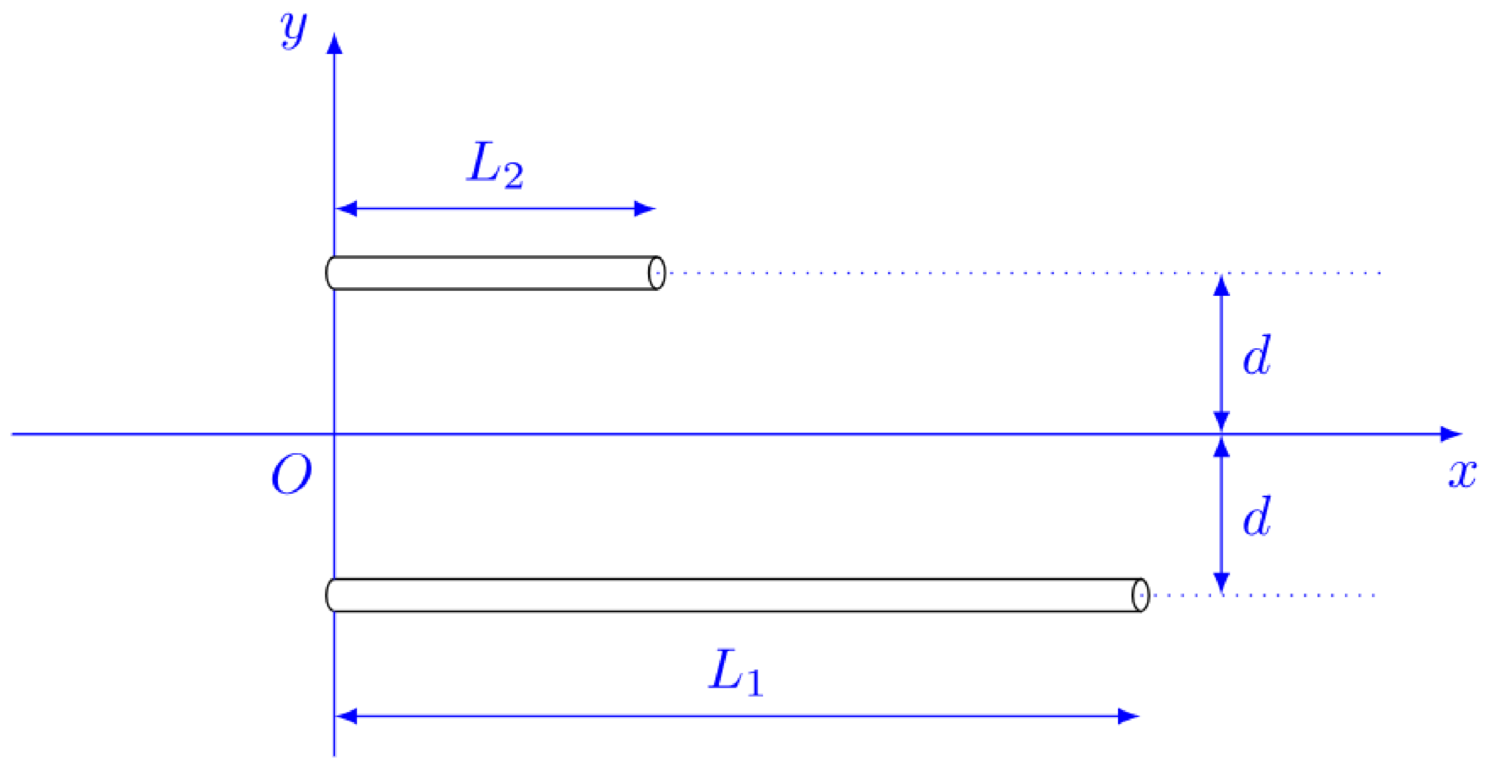
```
(-64344.604, 78446.599, 0.000)
```

```
Q2 = [
    [2E-6, 0, 0.47901, 0],
]

# x, y, z de la posición donde se desea calcular el campo:
x = 0
y = 0
z = 0

Ei1, Ej1, Ek1 = Ef(x,y,z,Q1)
Ei2, Ej2, Ek2 = Ef(x,y,z,Q2)
Ei1 + Ei2, Ej1 + Ej2, Ek1 + Ek2
```

```
(-64344.604, -1.666, 0.000)
```

Ejercicio 2:

En la figura se muestran el alambre 1 de longitud $L_1 = 15\,\mathrm{cm}$ y el alambre 2 de longitud $L_2 = 6\,\mathrm{cm}$, donde $d = 3\,\mathrm{cm}$.



Ambos alambres están cargados uniformemente. La carga total del alambre 1 es $400\,\mathrm{nC}$ y la del alambre 2 es $-400\,\mathrm{nC}$.

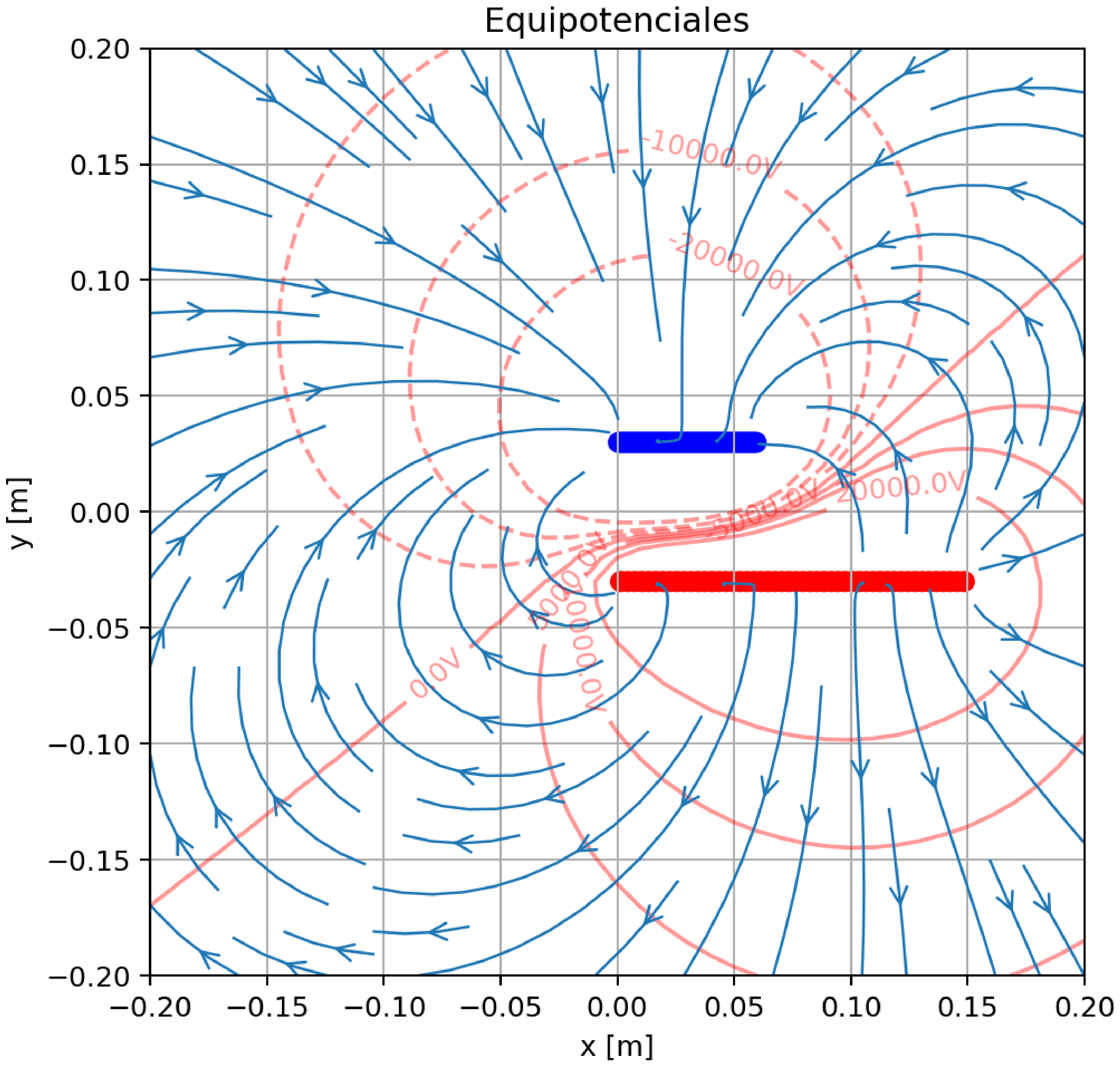
a) Graficar las equipotenciales de esta distribución en el plano xy. Incluir las líneas de campo en la misma figura.

```python
# Recomendaciones:
# 1. Crear las listas de cargas para cada segmento y sumarlas,
#    o armar directamente una única lista con las cargas de
ambos segmentos.

Qtotal = Q1 + Q2

# 2. Hacer un gráfico preliminar para tener una idea de los
valores de las equipotenciales:

equipotencialesPuntuales(Qtotal)

# 3. Se eligen cuáles equipotenciales mostrar en una lista.
Ejemplo:

levels = [-200, -100, 0 , 100, 200, 3000]

# 4. Graficar las equipotenciales con las líneas de campo,
ajustando el valor de "dim"
#    para obtener una buena visualización.

equipotencialesPuntuales(Qtotal, EF=True, levels = levels,
dim=20)
```

Python

```python
Qtotal = Q1 + Q2
niveles = [-20000, -10000, -5000, 0, 5000, 10000, 20000]
equipotencialesPuntuales(Qtotal, EF=True, niveles = niveles,
dim=0.2)
```

✓ 0.3s Python

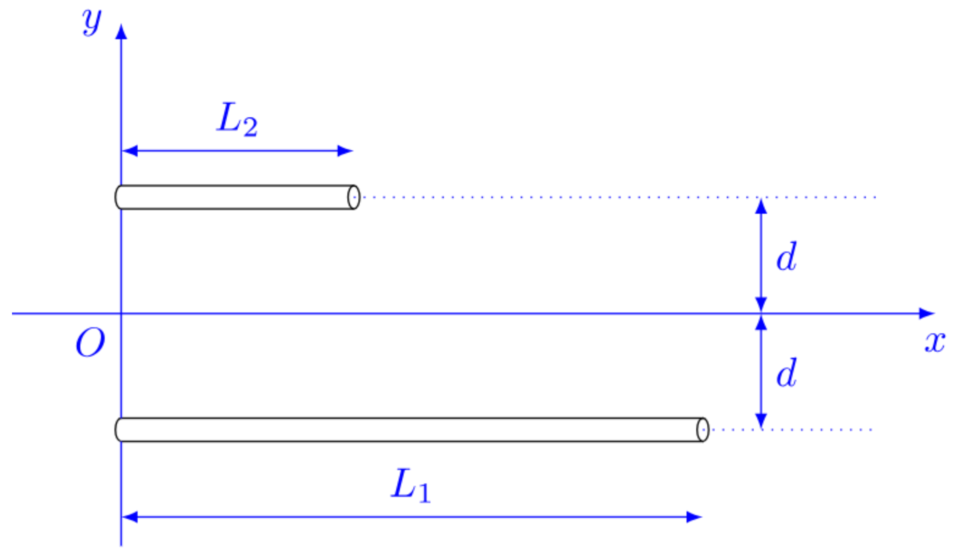


c) Encontrar alguna posición en el eje x tal que el potencial sea aproximadamente cero.

```python
# Aprovechar el gráfico de las equipotenciales para
# ubicar ese punto.

x = 0.066888
y = 0
z = 0
V(x,y,z,Qtotal)
```

✓ 0.0s

```
-8.251
```

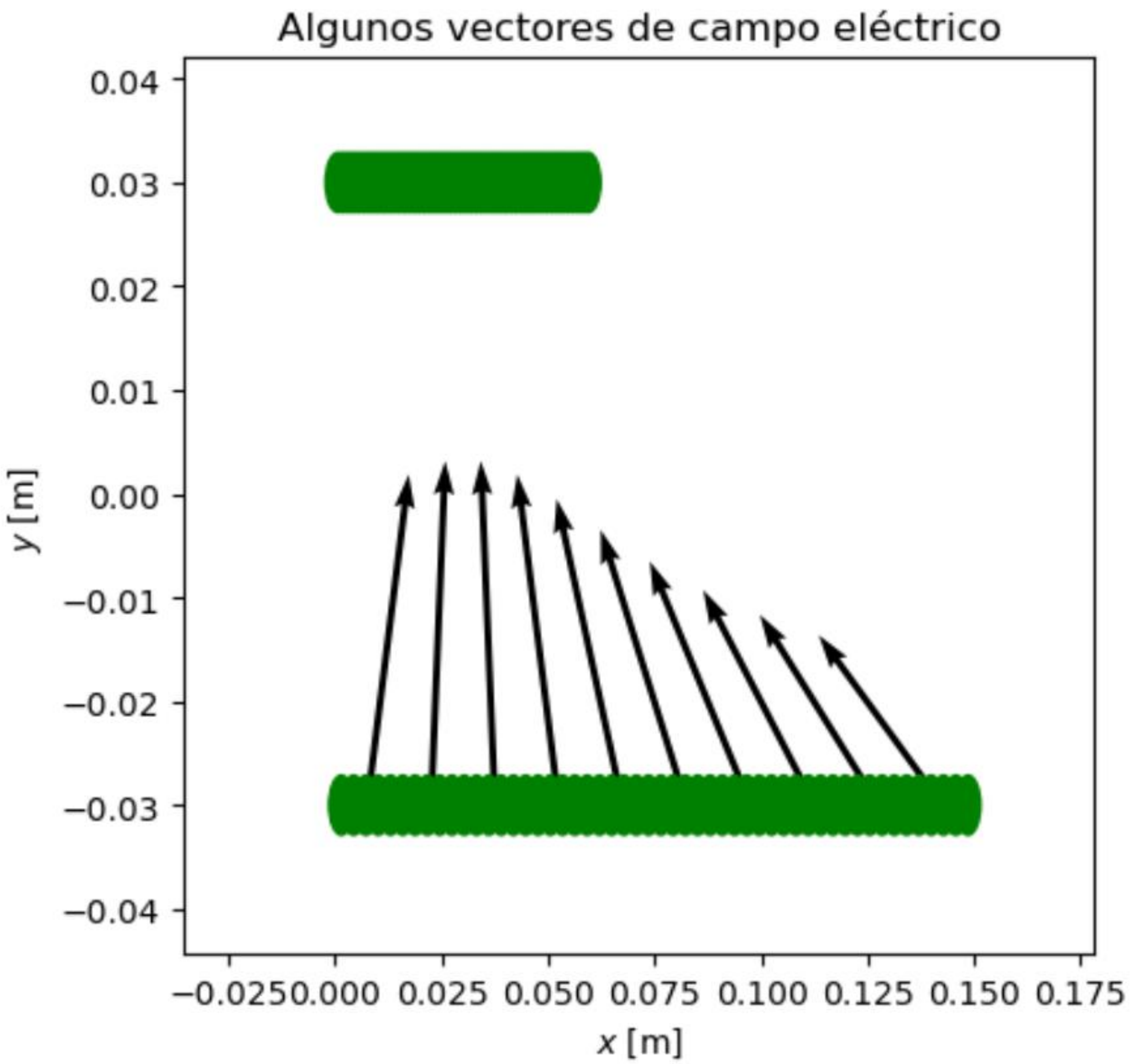


d) Calcular la fuerza neta que el alambre 2 ejerce sobre el alambre 1.

```
Fi = 0
Fj = 0
Fk = 0
for q1 in Q1:
    x = q1[1]
    y = q1[2]
    z = q1[3]
    Ei, Ej, Ek = Ef(x,y,z,Q2)
    Fi = Fi + q1[0] * Ei
    Fj = Fj + q1[0] * Ej
    Fk = Fk + q1[0] * Ek
Fi, Fj, Fk
```

```
(-0.069, 0.209, 0.000)
```

```
plotEfVector(Qtotal2, X, scale=20)
```



Suponiendo que el alambre 1 (el inferior) se libera, y teniendo en cuenta la fuerza total y estos vectores del campo eléctrico producidos por el alambre 2 en las posiciones del alambre 1, describir el movimiento inicial del alambre inferior.

Respuesta:

Escribir la respuesta en este lugar.

Si soltaramos el alambre de abajo lo que analiticamente sucederia seria que se desplazaria levemente hacia la izquierda y el extremo izquierdo se levantaria con mas fuerza respecto del extremo derecho, lo que provocaria un momento en el cuerpo (se inclina en el aire y *puede llegar* a girar)...

Respuesta:

la parte derecha de la barra se inclinara levemente hacia abajo mientras que la de la derecha subira con fuerza ya que en la parte izquierda los vectores de campo electrico son mas fuertes, tambien se desplaza un poco hacia la izquierda.

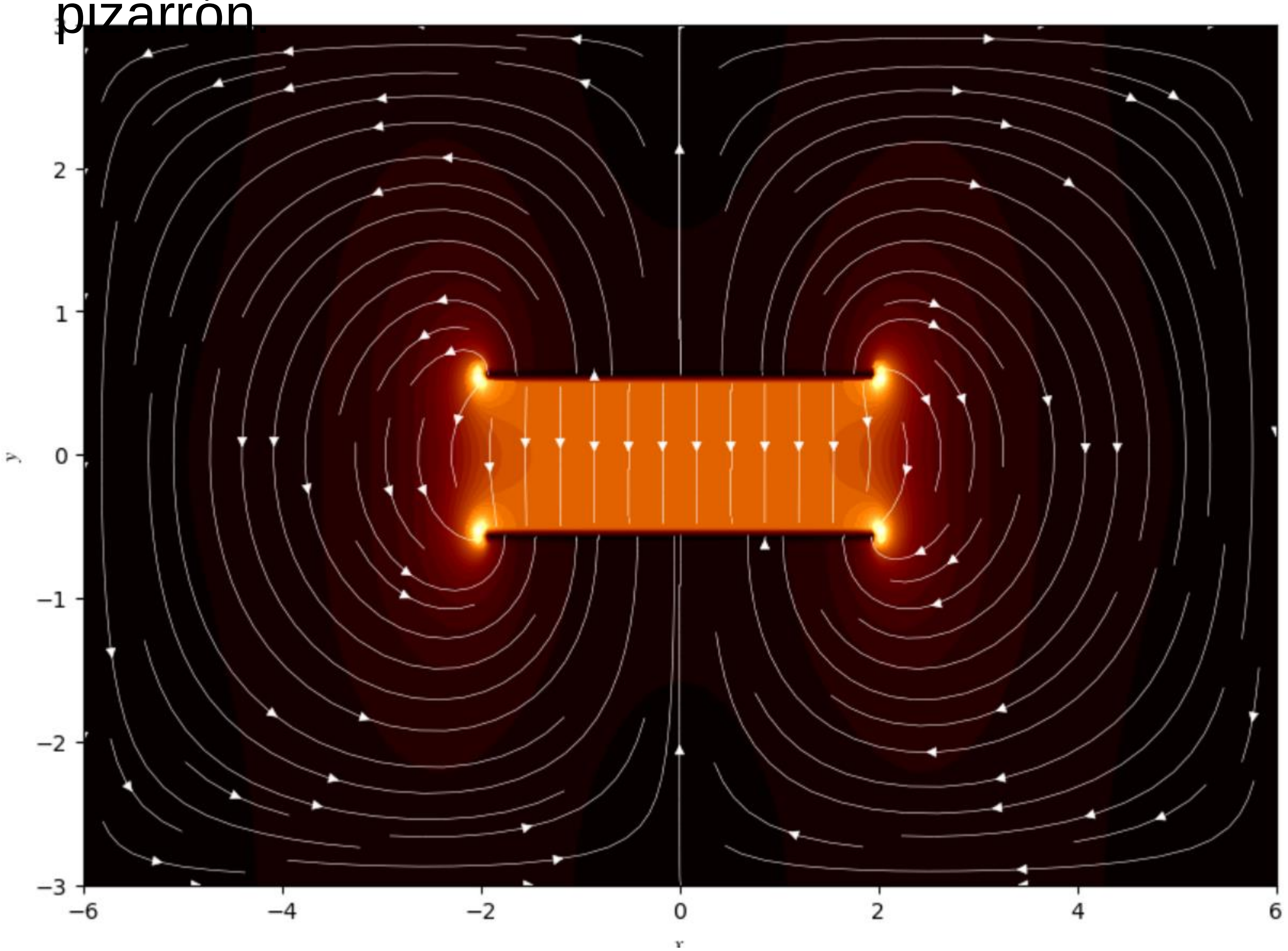
Respuesta:

Al soltar el hilo el lado izquierdo del mismo va a tender a acercarse mas rapido que el lado derecho por lo que rotará hasta ponerse en forma vertical

Más posibilidades

Cálculos simbólicos sin esfuerzo.

Generar problemas con conductores que no se pueden resolver en el pizarrón.



```
Eii = Ei.subs([(k, 9E9), (l, 5E-9), (a, -0.1), (b, 0.1)])
Ejj = Ej.subs([(k, 9E9), (l, 5E-9), (a, -0.1), (b, 0.1)])
Ekk = Ek.subs([(k, 9E9), (l, 5E-9), (a, -0.1), (b, 0.1)])
display(Eii)
display(Ejj)
display(Ekk)
```

$$E_i = -\frac{45.0}{\sqrt{x^2 + 0.2x + y^2 + z^2 + 0.01}} + \frac{45.0}{\sqrt{x^2 - 0.2x + y^2 + z^2 + 0.01}}$$

$$E_j = \frac{45.0y\,(0.1 - x)}{y^2\sqrt{y^2 + z^2 + (0.1 - x)^2} + z^2\sqrt{y^2 + z^2 + (0.1 - x)^2}} - \frac{45.0y\,(-x - 0.1)}{y^2\sqrt{y^2 + z^2 + (-x - 0.1)^2} + z^2\sqrt{y^2 + z^2 + (-x - 0.1)^2}}$$

$$E_k = \frac{45.0z\,(0.1 - x)}{y^2\sqrt{y^2 + z^2 + (0.1 - x)^2} + z^2\sqrt{y^2 + z^2 + (0.1 - x)^2}} - \frac{45.0z\,(-x - 0.1)}{y^2\sqrt{y^2 + z^2 + (-x - 0.1)^2} + z^2\sqrt{y^2 + z^2 + (-x - 0.1)^2}}$$

Y ahora podemos evaluar en la posición que nos interese. Recordar que estos r
válidos fuera del eje x, con $y > 0$ y $z > 0$.

Comentarios finales.

- **Análisis de múltiples problemas de campos y potenciales para mejorar su comprensión.**
  Muy fácil de adaptar según necesidades y dudas de los estudiantes.

- **Distribuciones contínuas**
  Posibilidad de realizar las integrales por completo utilizando código.
  Con poco esfuerzo se pueden estudiar aproximaciones de problemas sin solución analítica y con densidades no uniformes.

- **Resolver problemas libres y ver resultados rápidamente mediante la experimentación.**
  Posibilidad de plantear problemas de diseño complejos y que los cálculos no sean un limitante.

- **Superficies equipotenciales.**
  Algo que en el pizarrón es muy complicado, aquí se vuelve sencillo de estudiar.

- **Aprovechar el tiempo para analizar y no para calcular.**

- **Discusiones sobre aproximaciones.**

- **Trabajar fuera del plano.**
  La extensión de los problemas a 3D en el pizarrón o en el papel es muy limitada. En estas actividades, salir del plano no agrega dificultad ni tiempo.

- **Este es un primer intento.**
  El curso de mecánica en Python de UNLAM ya lleva 6 semestres y sigue sufriendo modificaciones.

- **Código abierto y reutilizable.**

GRACIAS por su atención.

Próximamente: Magnetostática.