

OBJETIVO DEL TRABAJO

El objetivo del trabajo es el de crear un programa en Matlab que, por un lado, represente gráficamente la superficie del potencial eléctrico y las líneas del campo eléctrico creado por dos cargas, y, por otro lado, que contenga una animación que muestre cómo se movería una carga de prueba bajo la acción de dicho campo eléctrico.

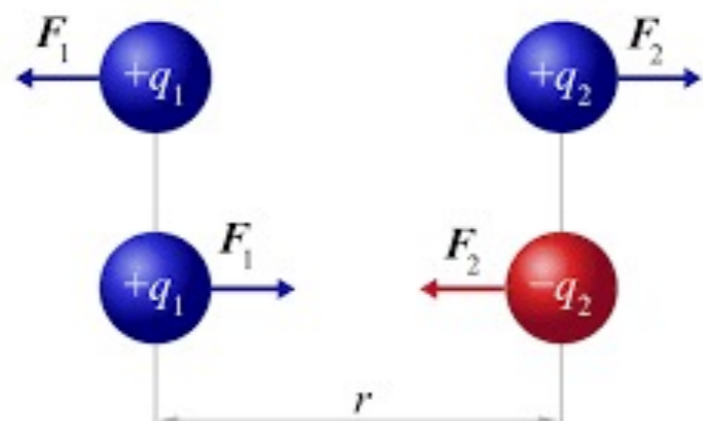
EXPLICACIÓN DEL FENÓMENO FÍSICO

Campo eléctrico y potencial eléctrico

Un **campo eléctrico**¹ es un campo de fuerzas creado por una partícula con carga eléctrica. El campo eléctrico tiene la misma dirección que la fuerza eléctrica que ejercería la carga sobre otra unidad de carga positiva (1). La fuerza se calcula mediante la Ley de Coulomb (2):

$$\vec{E} = \frac{\vec{F}}{q} = \frac{k_e q}{r^2} \vec{u} \quad (1)$$

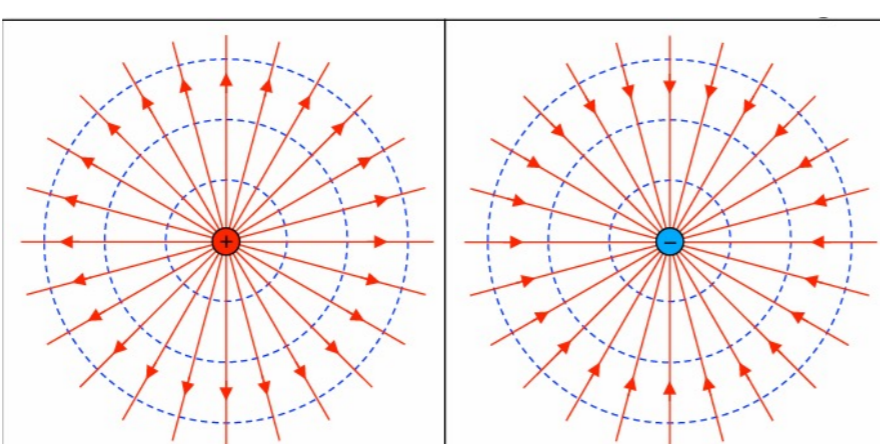
$$\vec{F} = \frac{k_e q_1 q_2}{r^2} \vec{u} \quad (2)$$



Este fenómeno físico cumple con el principio de superposición, por lo que, si hay varias cargas en el espacio, el campo eléctrico neto sería la suma del campo generado por cada carga.

Se define el potencial eléctrico (V) como la energía potencial por unidad de carga que hay en un punto. La expresión utilizada en el programa es para el potencial generado por una partícula puntual (Q).

$$V = \frac{U}{q_0} = k_e \frac{Q}{r} \quad (3)$$



En azul aparecen representadas las superficies equipotenciales, y en rojo las líneas del campo eléctrico

EXPLICACIÓN DE LA PARTE COMPUTACIONAL

Datos iniciales: magnitud de las dos cargas y sus posiciones.

• Representación del potencial eléctrico:

Utilizando la función *meshgrid* se crea una malla, y en cada uno de esos puntos se calculan las componentes x e y del campo eléctrico con la expresión 1, aplicando el principio de superposición.

Después, se calcula el potencial en cada punto debido a la carga 1 y 2 (expresión 3).

Finalmente, con la función *surf* se representa la superficie del potencial eléctrico.

• Representación de las líneas de campo eléctrico:

Se hace a partir de los datos calculados, mediante la función *streamslice*. Y, con una función auxiliar llamada *draw_charge*, se representan las cargas con sus signos.

• Animación:

Datos: la masa, la carga de la partícula y su velocidad y posición iniciales.

Se representan las líneas de campo como fondo con la función *quiver*. Y, con la función *animatedline*, se crea la animación de la trayectoria. Sus puntos se actualizan con un bucle utilizando la integración por el método de Euler: para calcular la fuerza en la posición (r) se utiliza la función *E_point*. Con la fuerza se calcula la aceleración (segunda ley de Newton), después, la velocidad y, finalmente, se actualiza la posición (r). El bucle continúa hasta un tiempo t (que puede modificarse).

```
%DEFINICIÓN DEL ESPACIO:
a=20; %cambiando "a" cambias el zoom
n=7; %resolución de la malla
[x, y] = meshgrid(-a:a/n:a, -a: a/n:a);

%CAMPO ELÉCTRICO
% inicializo las componentes del
% campo eléctrico
Ex = zeros(size(x));
Ey = zeros(size(y));
% calculo la distancia entre cada
% carga y cada punto.

r1 = sqrt((x - x1).^2 + (y - y1).^2);
r2 = sqrt((x - x2).^2 + (y - y2).^2);

%el campo eléctrico en cada punto debido a q1
%cos_a=(x-x1)/r1
%Ex=kq*cos_a./r.^2
%voy aplicando principio de superposición
% a cada carga
%primero a la primera carga q1
% esto es E cos alpha
Ex = Ex +k* q1 * (x - x1) ./ ( r1.^3);
% esto es E sen alpha
Ey = Ey +k* q1 * (y - y1) ./ ( r1.^3);
%ahora, a cada componente le sumo el
% campo eléctrico en cada punto debido a q2
Ex = Ex + k*q2 * (x - x2) ./ ( r2.^3);
Ey = Ey + k*q2 * (y - y2) ./ ( r2.^3);

%POTENCIAL:
V1=k* q1./r1;
V2=k* q2./r2;
```

Código de la primera parte del programa

FIGURAS DEL PROGRAMA

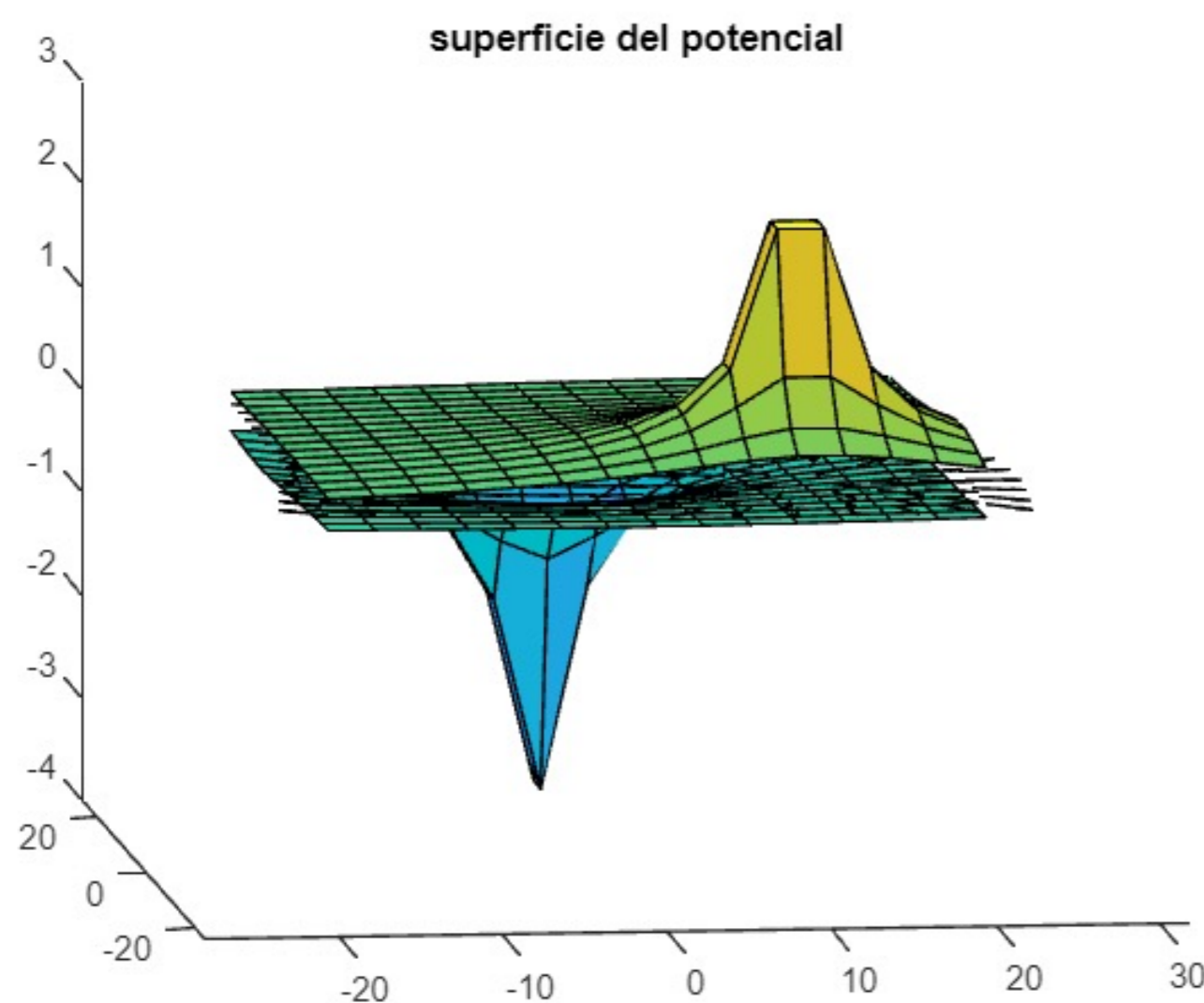


Figura 1. En la siguiente figura se representa la superficie del potencial eléctrico obtenida en la primera parte del programa.

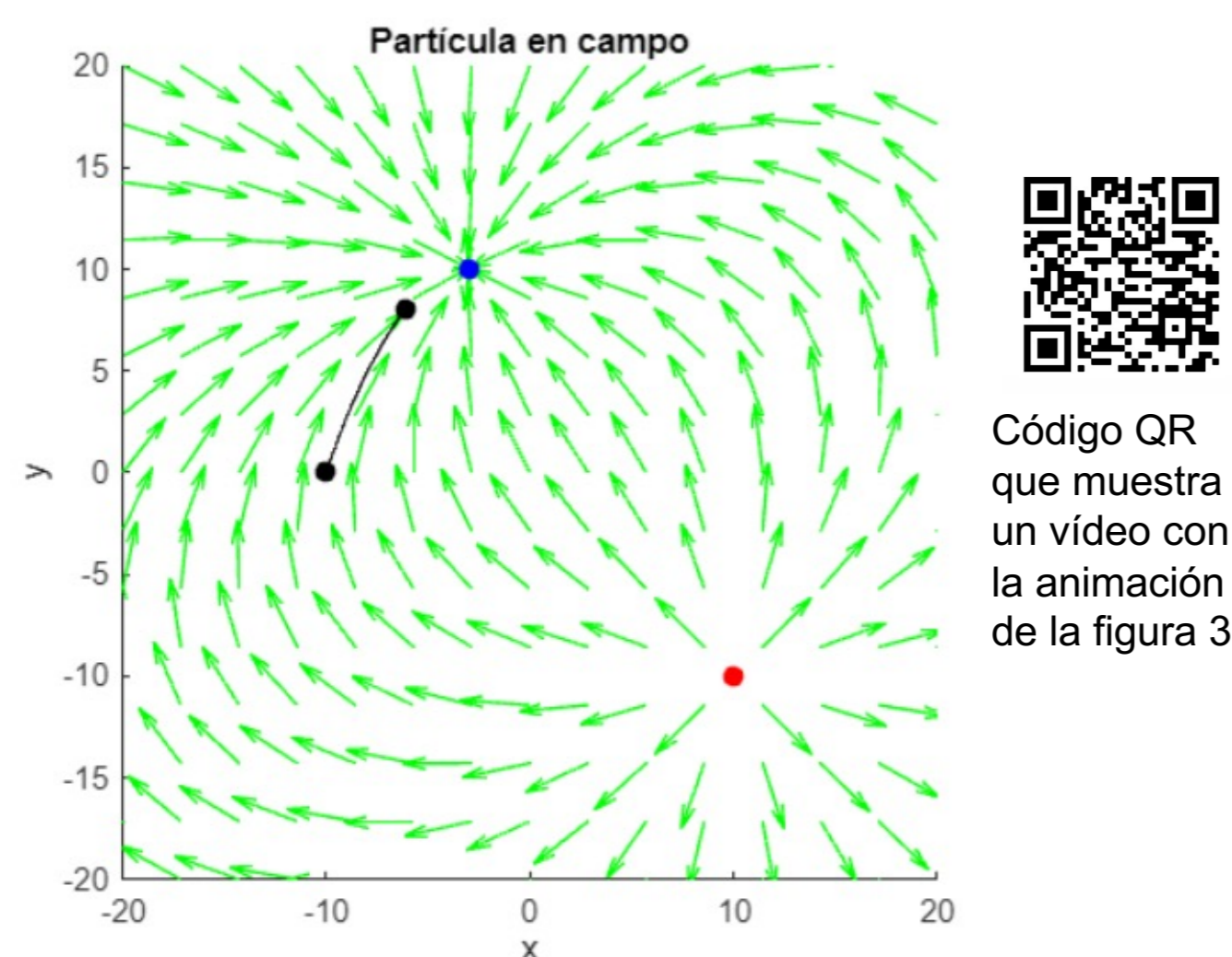


Figura 3. Trayectoria de la carga de prueba bajo la acción del campo eléctrico.

EXPLICACIÓN DE LAS FIGURAS

- En la primera figura se observa cómo el potencial aumenta en valor absoluto al acercarse a las cargas.
- También se observa cómo tiende a infinito cuando es nula la distancia a las cargas (aunque no se ve directamente debido a que la resolución de la malla no es infinita).
- Al tratarse el ejemplo propuesto de un dipolo, se observa que todas las líneas de campo que salen de la carga positiva entran en la negativa, en la segunda figura.
- En la tercera figura, se representa la trayectoria que sigue una carga positiva en el campo eléctrico durante 20 segundos.
- En la quinta figura, se muestra el bucle utilizado en la tercera parte del programa

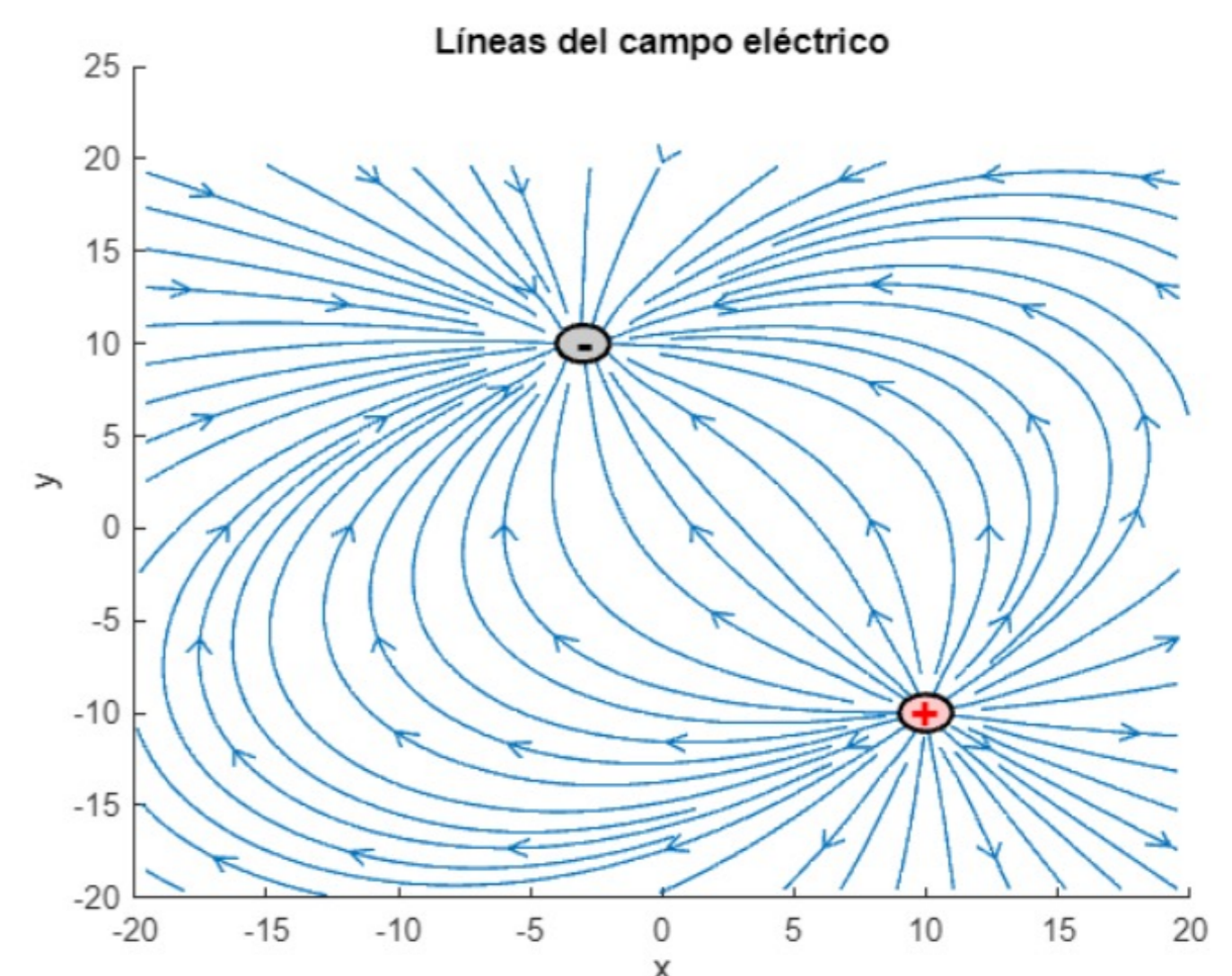


Figura 2. Arriba se representan las líneas del campo eléctrico generado por el campo eléctrico

```
% == Partícula ==
particle_plot = plot(pos(1), pos(2), 'ko', 'MarkerFaceColor', 'k');
plot(pos(1), pos(2), 'ko', 'MarkerFaceColor', 'k');
trail = animatedline('Color', 'k');

% Función para el campo de una carga puntual
E_point = @(q, r0, r) k*q * (r - r0) / norm(r - r0)^3;

% Simulación
for i = 1:frames
% Calcular campo en posición actual
if norm(pos - q1_pos) < 0.5 || norm(pos - q2_pos) < 0.5
break; % Evitar colisión con las cargas para que
% el ordenador no cometa errores de redondeo
% [al dividir por números muy pequeños]
end

E1 = E_point(q1, q1_pos, pos);
E2 = E_point(q2, q2_pos, pos);
E = E1 + E2;

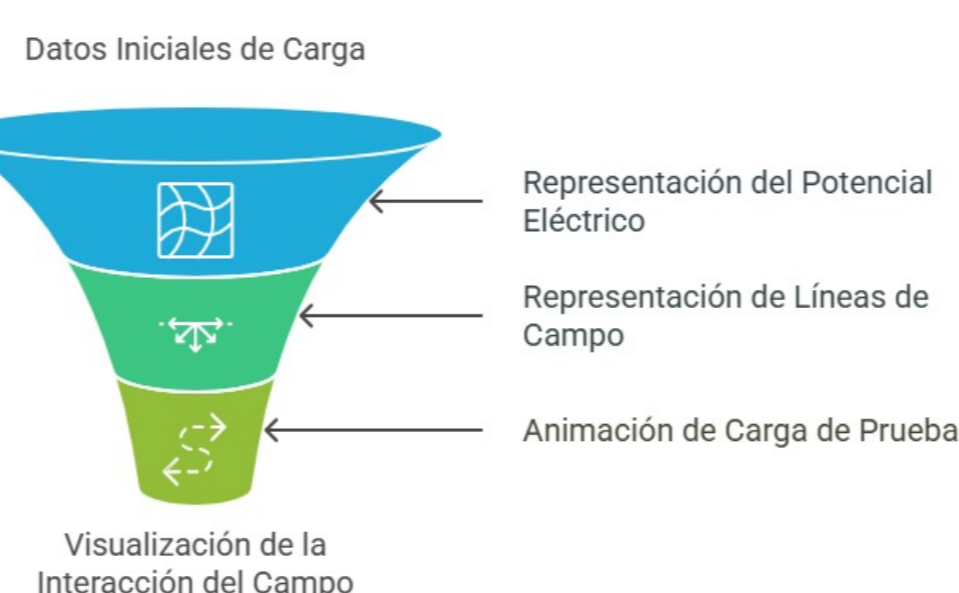
F = q_test * E;
a = F / m;

% Integración simple (Euler)
v = v + a * dt;
pos = pos + v * dt;

% Actualizar gráfico
set(particle_plot, 'XData', pos(1), 'YData', pos(2));
% plot(pos(1), pos(2), 'ko', 'MarkerFaceColor', 'k');
addpoints(trail, pos(1), pos(2));
drawnow;
end
```

Figura 5. Fragmento de la tercera parte del programa.

Visualización de la Interacción del Campo Eléctrico



REFERENCIAS

- [1] Rodríguez de Fonseca B. *CE1. Interacción eléctrica, campo eléctrico, leyes de Coulomb y de Gauss*[presentación de clase]. Madrid: Universidad Complutense de Madrid; 2025.
- [2] Jiménez Castellanos JF. *Laboratorio de Computación Científica* [apuntes de clase en línea]. Madrid: Universidad Complutense de Madrid; 2020.

AGRADECIMIENTOS

Este trabajo forma parte del proyecto INNOVA "Integración de la computación científica en la enseñanza de la Física (INCENFIS) (ref 344)