

Obligatorio 3  
Puntos de Fekete  
Fundamentos de Optimización  
Facultad de Ingeniería  
2023

Francisco Santini  
4.894.396-1  
Ian Arazny  
5.147.233-7

<b>Propuesta 4: Puntos de Fekete Introducción</b>	<b>3</b>
<b>Punto 1: Cálculo y computación del gradiente</b>	<b>3</b>
Breve introducción al método de gradiente proyectado:	5
<b>Punto 2: Implementación del método de gradiente proyectado para cada punto</b>	<b>5</b>
<b>Punto 3: Implementación del método de gradiente proyectado para toda la configuración de puntos</b>	<b>6</b>
<b>Punto 4: Comparación de los métodos</b>	<b>6</b>

#### Propuesta 4: Puntos de Fekete Introducción

En esta propuesta buscaremos puntos en la esfera que estén “bien distribuidos”, de acuerdo a la denominada energía logarítmica.

Dados  $N$  puntos  $x_i$  en la esfera  $S^2 = \{x \in \mathbb{R}^3 : \|x\|_2 = 1\}$ , definimos la energía logarítmica del conjunto de puntos como

$$E(x_1, x_2, \dots, x_N) = - \sum_{i=1}^N \sum_{j=1}^N \log(\|x_i - x_j\|_2)$$

El objetivo de la propuesta será buscar una configuración de los  $N$  puntos que sea un mínimo local de  $E$ .

##### Punto 1: Cálculo y computación del gradiente

Hallar el gradiente de  $E$  respecto a un punto  $x_i$ . Implementarlo, e implementar también el gradiente respecto a toda la configuración de puntos  $x = (x_1, x_2, \dots, x_N) \in \mathbb{R}^{3N}$ .

Se halla el gradiente de la función  $E$ , para un punto  $x_i$  y para una configuración de puntos  $x$  de la siguiente manera:

Tomemos un punto específico  $x_i$  y denotemos sus coordenadas como  $x_{i_k}$ , donde  $k$  representa el índice de la coordenada correspondiente. Diferenciemos  $E$  con respecto a  $x_{i_k}$

$$\frac{\partial E}{\partial x_{i_k}} = - \sum_{j=1, j \neq i}^N \frac{\partial}{\partial x_{i_k}} (\log(\|x_i - x_j\|_2))$$

Usando la regla de la cadena para diferenciar el logaritmo y luego aplicaremos la regla de la cadena nuevamente para diferenciar la norma euclidiana al cuadrado. Comenzamos diferenciando el logaritmo:

$$\frac{\partial E}{\partial x_{i_k}} = - \sum_{j=1, j \neq i}^N \frac{1}{\|x_i - x_j\|_2} \frac{\partial}{\partial x_{i_k}} ((x_{i_1} - x_{j_1})^2 + (x_{i_2} - x_{j_2})^2 + \dots + (x_{i_N} - x_{j_N})^2)$$

Aplicando la regla de la cadena, obtenemos:

$$\frac{\partial E}{\partial x_{i_k}} = - \sum_{j=1, j \neq i}^N \frac{1}{\|x_i - x_j\|_2} \frac{\partial}{\partial x_{i_k}} \left( \sum_{l=1}^N (x_{i_l} - x_{j_l})^2 \right)$$

Simplificando la expresión interna, obtenemos:

$$\frac{\partial E}{\partial x_{i_k}} = - \sum_{j=1, j \neq i}^N \frac{1}{\|x_i - x_j\|_2} \frac{\partial}{\partial x_{i_k}} \left( \sum_{l=1}^N (x_{i_l} - x_{j_l})^2 \right)$$

Ahora, diferenciamos el término dentro de la suma con respecto a  $x_{i_k}$

$$\frac{\partial}{\partial x_{i_k}} \left( \sum_{l=1}^N (x_{i_l} - x_{j_l})^2 \right) = \frac{\partial}{\partial x_{i_k}} ((x_{i_1} - x_{j_1})^2 + (x_{i_2} - x_{j_2})^2 + \dots + (x_{i_N} - x_{j_N})^2) = 2(x_{i_k} - x_{j_k})$$

Sustituyendo este resultado en la expresión anterior, obtenemos:

$$\frac{\partial E}{\partial x_{i_k}} = - \sum_{j=1, j \neq i}^N \frac{1}{\|x_i - x_j\|_2} \cdot 2(x_{i_k} - x_{j_k})$$

Por lo tanto, la fórmula para el k-ésimo componente del gradiente de E respecto a  $x_i$ :

$$\nabla E / \nabla x_{i_k} = - \sum_{j=1, j \neq i}^N \frac{x_{i_k} - x_{j_k}}{\|x_i - x_j\|_2^2}$$

En particular, siendo  $x_i = (x_1, x_2, x_3)$  se obtiene:

$$\frac{\partial E}{\partial x_i} = \left( - \sum_{j=1, j \neq i}^3 \frac{x_{i_1} - x_{j_1}}{\|x_i - x_j\|_2^2}, - \sum_{j=1, j \neq i}^3 \frac{x_{i_2} - x_{j_2}}{\|x_i - x_j\|_2^2}, - \sum_{j=1, j \neq i}^3 \frac{x_{i_3} - x_{j_3}}{\|x_i - x_j\|_2^2} \right)$$

Generalizando, para toda la configuración de puntos  $x = (x_1, \dots, x_N)$ , obtenemos:

$$\frac{\partial E}{\partial x} = \left( \frac{\partial E}{\partial x_1}, \frac{\partial E}{\partial x_2}, \dots, \frac{\partial E}{\partial x_N} \right)$$

Para implementar la primer parte se calcula el gradiente de la función de energía E respecto a un punto específico  $x_i$ . Recorre todos los puntos  $x_j$ , en la

configuración  $x$ , excepto  $x_i$ , y se suma las contribuciones al gradiente en cada coordenada de  $x_i$  utilizando la fórmula mencionada anteriormente. Mientras para la implementación de la segunda parte es básicamente lo mismo solo que recorriendo toda la configuración de puntos aplicando la fórmula obtenida.

Breve introducción al método de gradiente proyectado:

El método del gradiente proyectado es un algoritmo de optimización para funciones sujetas a restricciones de dominio.

El objetivo del método del gradiente proyectado es minimizar una función objetivo sujeta a una restricción de dominio, que puede ser representada como una región factible en el espacio de búsqueda. Este método combina el uso del gradiente de la función objetivo y la proyección de puntos para garantizar que las soluciones se mantengan dentro de la región factible. En este caso se la función objetivo se trata de la energía logarítmica y la restricción de dominio es que pertenezca a la esfera.

Punto 2: Implementación del método de gradiente proyectado para cada punto

Implementar un método para hallar un mínimo local de  $E$ , de la siguiente forma: para cada uno de los  $N$  puntos (en forma consecutiva), actualizar la posición de  $x_i$  según un

método del gradiente proyectado, con un paso fijo, usando como condición de parada que la

norma de la diferencia con el iterado anterior sea menor que cierta tolerancia.

Repetir el ciclo hasta convergencia (tomar una condición de parada similar, pero a nivel de toda la configuración de puntos).

Se implementó el método del gradiente proyectado en la función  $E$  para hallar un mínimo local.

La función  $\text{proj}(x)$  realiza la proyección de un punto  $x$  en caso de que su norma no sea igual a 1. Esto garantiza que los puntos se mantengan dentro de la esfera unitaria. El algoritmo itera hasta que se cumple el criterio de convergencia establecido en  $\text{tol1}$ , que se basa en la diferencia entre la configuración actual de puntos y la anterior. Dentro del bucle principal, se actualiza la configuración de puntos. Se calcula el punto proyectado  $x_{\text{techo}}$  utilizando la función de proyección y  $\text{grad\_E\_puntual}$ , función la cual calcula el gradiente puntual de la función de energía logarítmica para un punto específico en la configuración de puntos, para luego realizar una actualización ponderada de los puntos  $x_2$  utilizando el gradiente y los parámetros  $\alpha$  y  $s$ .

El método registra el valor de la función de energía en cada iteración en la lista  $\text{es2}$  y realiza un seguimiento del número total de iteraciones ( $\text{it}$ ) y el número de veces que se revisa la configuración completa de puntos ( $\text{it2}$ ).

Finalmente, el método devuelve la configuración final de puntos  $x_2$ , el número total de iteraciones, el número de iteraciones completas de revisión de la configuración

de puntos y un array que contiene los valores de la función de energía en cada iteración.

En resumen, el código implementa un método de gradiente proyectado para optimizar una función de energía logarítmica en una configuración de puntos en una esfera, garantizando que los puntos se mantengan dentro de la esfera durante la optimización.

Punto 3: Implementación del método de gradiente proyectado para toda la configuración de puntos

Implementar un método de gradiente proyectado, tomando como variable la configuración de puntos completa  $x \in R^{3N}$ , con condición de parada similar que el ítem anterior.

La implementación de este punto es exactamente igual a la anterior solamente que en este código se utiliza el gradiente de toda la configuración de puntos.

Punto 4: Comparación de los métodos

Comparar los métodos en tiempos de ejecución, y valores de la función objetivo alcanzados. Graficar la configuración de puntos resultante en  $R^3$ . Puede superponerlos a la esfera.

Unos comentarios previos sobre la variación de valores de  $\alpha$  y  $s$ , se observa que para el paso los conceptos generales se mantienen tal que  $\alpha$  controla la magnitud de los cambios realizados en cada iteración del algoritmo. Valores pequeños de  $\alpha$  resultan en una convergencia más lenta, pero pueden ayudar a evitar oscilaciones y saltos grandes. Por otro lado, valores grandes de  $\alpha$  pueden conducir a una convergencia más rápida, pero pueden ser propensos a oscilaciones o incluso divergir, por otro lado para valores pequeños de  $s$  indican que el algoritmo se mueve en pequeños pasos a lo largo del gradiente, lo que puede llevar a una convergencia más precisa pero lenta. Valores grandes de  $s$  permiten movimientos más rápidos, pero pueden hacer que el algoritmo oscile o se aleje del mínimo local.

Se observa que los valores obtenidos son para  $\alpha = 1$  y  $s = 1$

Se muestran los valor obtenidos en la tabla debajo:

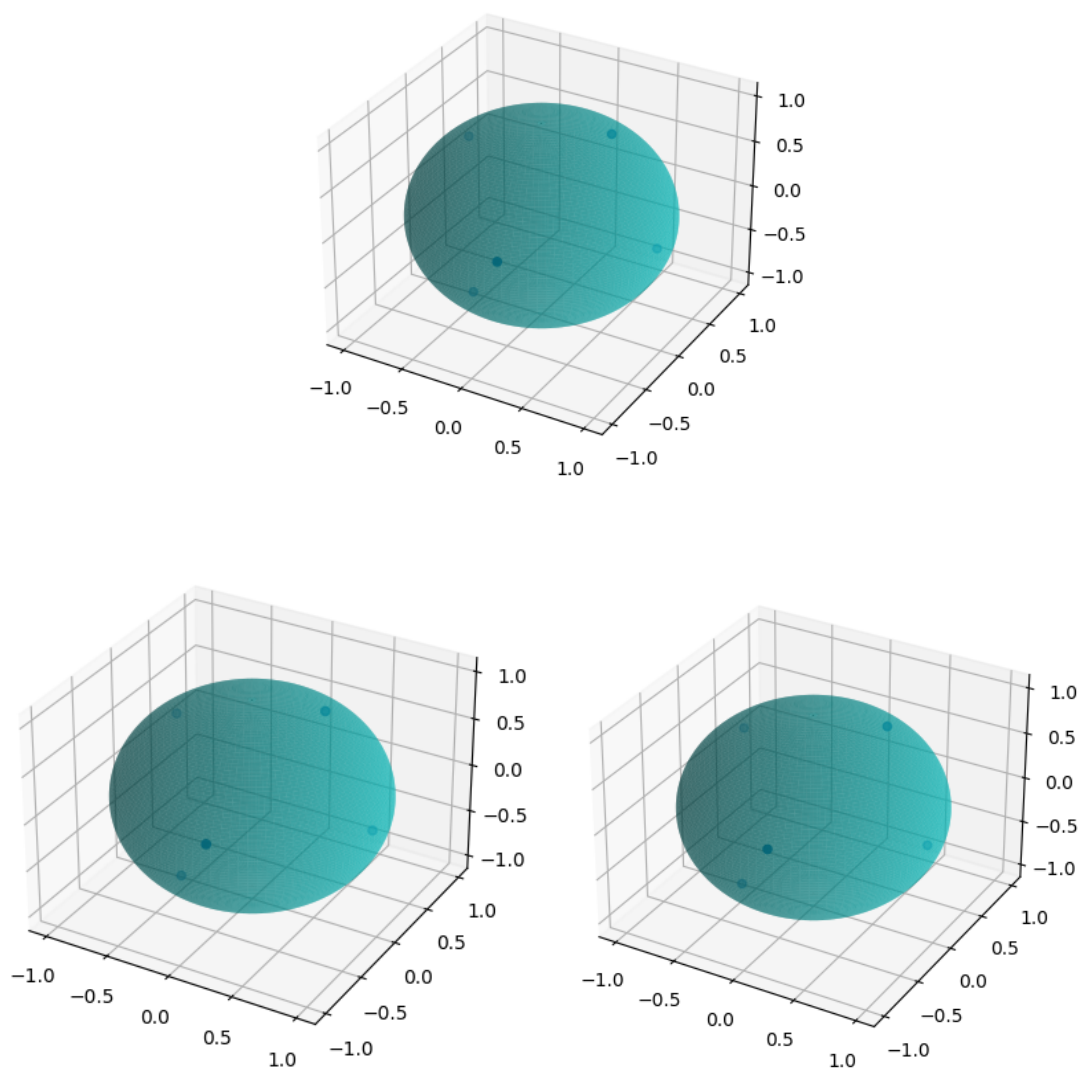
	N=5	
Método	2	3
Valor de la energía	-4.420507154229128	-4.420507153800402
Número iteraciones sobre cada punto	475	535
Número iteraciones sobre la configuración de puntos	95	107
Tiempo de ejecución(en segundos)	0.08234669599914923	0.0860644719996344
	N=16	
Método	2	3
Valor de la energía	-36.102794705759074	-36.102794690423615
Número iteraciones sobre cada punto	7008	10144
Número iteraciones sobre la configuración de puntos	438	634
Tiempo de ejecución(en segundos)	3.862492197999927	4.564817671000128

	N=100	
Método	2	3
Valor de la energía	-1083.3550121672783	-1083.3655834576637
Número iteraciones sobre cada punto	296300	383100
Número iteraciones sobre la configuración de puntos	2963	3831
Tiempo de ejecución(en segundos)	861.4168869019995	1118.852429787

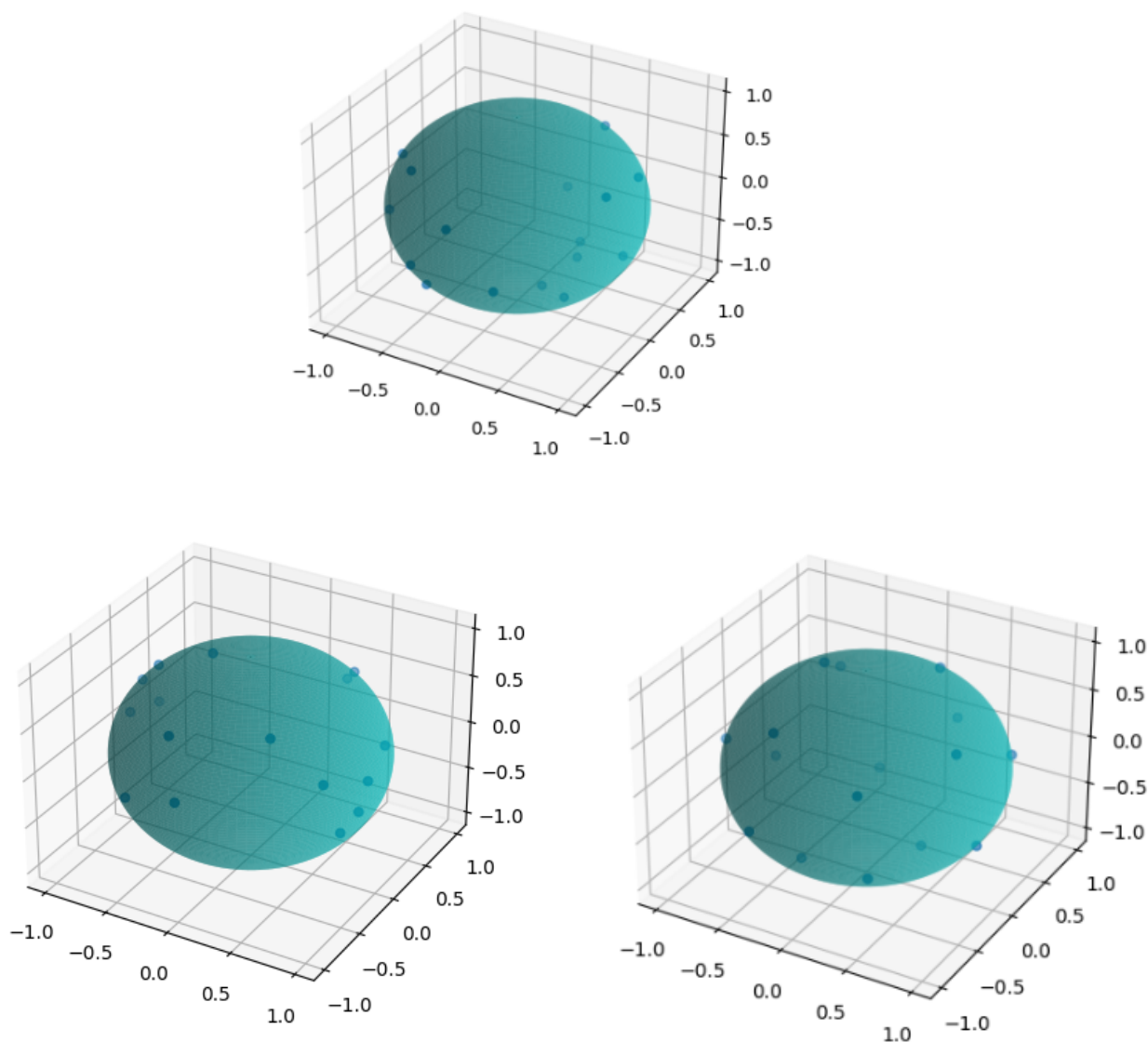


Debajo se grafica la configuración de puntos para los valores de  $N$  igual a 5, 16 y 100.

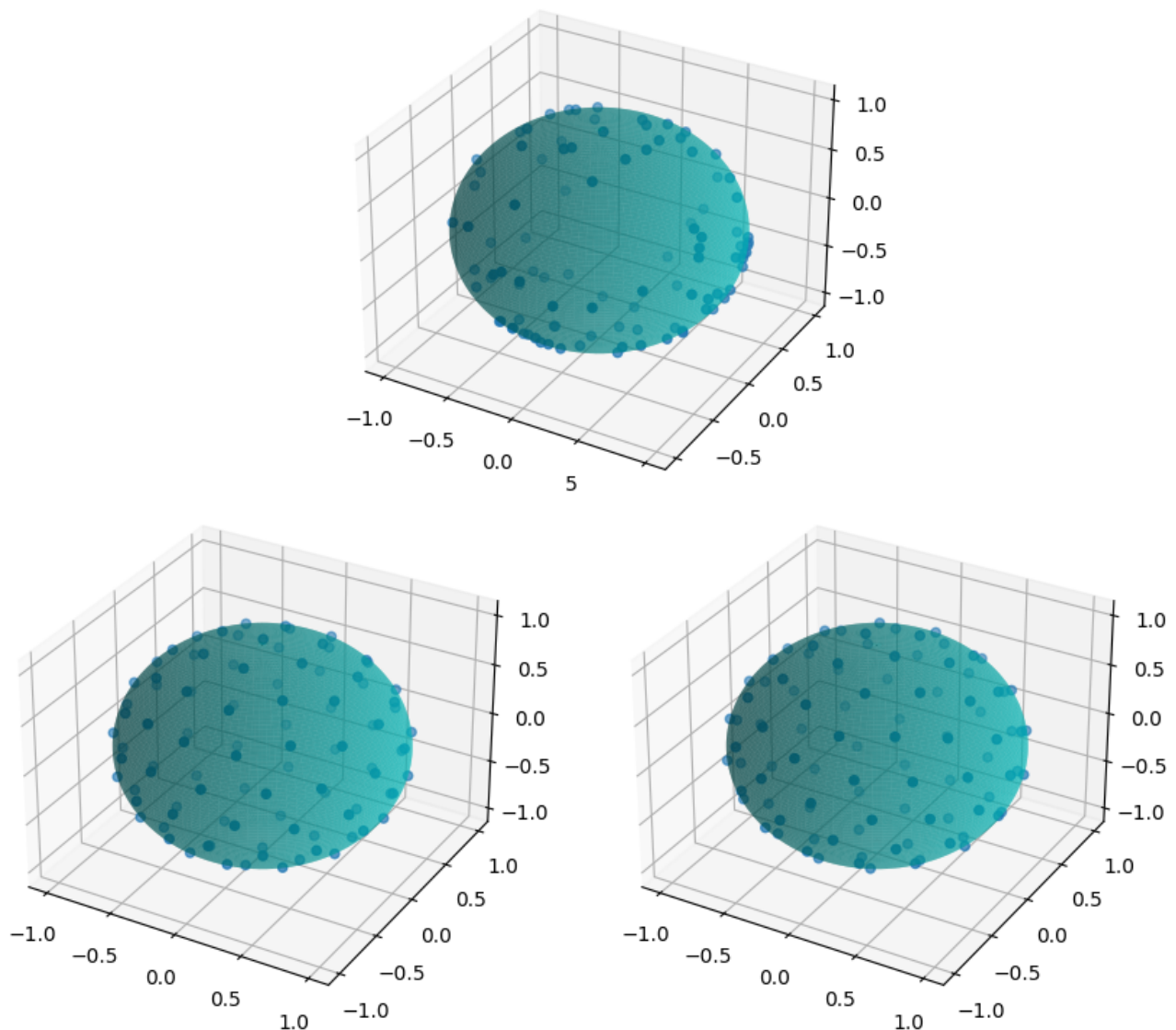
### Configuración inicial $N=5$



Configuración para 5 puntos: utilizando los métodos 2 y 3, a la izquierda y derecha respectivamente

Configuración inicial N=16

Configuración para 16 puntos: utilizando los  
métodos 2 y 3, a la izquierda y derecha respectivamente

Configuración inicial N=100:

Configuración para 100 puntos: utilizando los métodos 2 y 3, a la izquierda y derecha respectivamente.

Observemos los puntos a comparar separando por criterios a partir de los datos obtenidos:

**Valor de la energía:** En general, ambos métodos alcanzan valores de energía muy similares, con diferencias muy pequeñas en la escala de los resultados. Esto indica que ambos métodos son efectivos para minimizar la energía y obtener configuraciones de puntos cercanas al mínimo global.

**Número de iteraciones sobre cada punto:** En la mayoría de los casos, el código del punto dos (utilizando `grad_E_puntual`) requiere menos iteraciones sobre cada punto en comparación con el código del punto tres (utilizando `grad_E`). Esta diferencia se vuelve más pronunciada a medida que aumenta el valor de  $N$ . Esto sugiere que el primer código converge más rápidamente a nivel de cada punto individual.

**Número de iteraciones sobre la configuración de puntos:** En la mayoría de los casos, el código del punto dos también requiere menos iteraciones sobre la configuración de puntos en comparación con el código del punto tres. Al igual que el punto anterior, esta diferencia se acentúa a medida que aumenta el valor de  $N$ . Esto indica que el primer código converge más rápidamente a nivel global, alcanzando una configuración óptima en menos iteraciones.

**Tiempo de ejecución:** El código del punto dos generalmente presenta un tiempo de ejecución más rápido que el segundo código, especialmente a medida que aumenta el valor de  $N$ .

### **Conclusiones:**

El código del punto dos muestra un mejor desempeño en términos de velocidad de convergencia tanto a nivel de cada punto como a nivel global, así como un tiempo de ejecución más rápido en la mayoría de los casos.