

**UNIVERSIDAD COMPLUTENSE DE MADRID**  
**FACULTAD DE CIENCIAS FÍSICAS**

DEPARTAMENTO DE ARQUITECTURA DE COMPUTADORES Y AUTOMÁTICA



**TRABAJO DE FIN DE GRADO**

Código de TFG: ACA-01

Control de sistemas multiagente y sus aplicaciones

Control of multi-agent systems and their applications

Supervisor/es: Lía García Pérez / Juan Jiménez

**Xabier Blázquez González**

Grado en Física

Curso académico 2022-23

Convocatoria ordinaria de junio

## **Resumen:**

Los problemas de búsqueda de fuente son aquellos en los que un grupo de robots autónomos debe localizar y seguir la fuente de una señal desconocida. En este trabajo se ha propuesto una formación circular simétrica de agentes en 2D y en 3D para conseguirlo, ya que facilita la estimación del gradiente y de la matriz Hessiana en el centro de la formación mediante una media ponderada de la señal medida por cada agente. Se han simulado numéricamente diferentes algoritmos basados en consenso: ascenso del gradiente con y sin búsqueda de línea, ascenso tipo Newton-Raphson con y sin búsqueda de línea, gradiente conjugado y método Quasi-Newton. El rendimiento de cada algoritmo en convergencia y en complejidad temporal y espacial se ha analizado en 2D en este artículo, siendo el caso 3D parecido a éste. Además, se ha estudiado cómo funcionan estos algoritmos bajo mediciones ruidosas y errores de formación.

## **Abstract:**

Source seeking problems are those in which a group of autonomous robots must locate and follow the source of an unknown signal. In this paper a symmetric circular formation of agents has been proposed in 2D and in 3D in order to achieve this, as it facilitates the estimation of the gradient and the Hessian at its center via a weighted average of the signal measurement by each agent. Different consensus-based algorithms have been numerically simulated: gradient-ascent with and without line search, Newton-Raphson-like ascent with and without line search, conjugate gradient and Quasi-Newton method. The performance of each algorithm in convergence and in time and spatial complexity has been analyzed in 2D in this article, with the 3D case being similar to this one. In addition, how these algorithms perform under noisy measurements and formation errors has been studied.

# Índice

<b>1. Introducción</b>	<b>1</b>
<b>2. Objetivos</b>	<b>1</b>
<b>3. Metodología</b>	<b>2</b>
3.1. Problema en 2D . . . . .	2
3.1.1. Aproximación del gradiente . . . . .	2
3.1.2. Aproximación de la matriz Hessiana . . . . .	3
3.2. Problema en 3D . . . . .	3
3.2.1. Aproximación del gradiente . . . . .	4
3.2.2. Aproximación de la matriz Hessiana . . . . .	4
3.3. Características de las simulaciones numéricas . . . . .	5
<b>4. Resultados</b>	<b>5</b>
4.1. Método del gradiente ascendente . . . . .	5
4.2. Método de Newton-Raphson . . . . .	7
4.3. Métodos con búsqueda de línea . . . . .	9
4.3.1. Método del gradiente ascendente con búsqueda de línea	11
4.3.2. Método de Newton-Raphson con búsqueda de línea . .	12
4.3.3. Método del gradiente conjugado con búsqueda de línea	13
4.3.4. Método Quasi-Newton con búsqueda de línea . . . . .	15
<b>5. Conclusiones</b>	<b>16</b>
<b>6. Bibliografía</b>	<b>19</b>

## 1. Introducción

La búsqueda de la fuente de una señal es un problema relevante para varias aplicaciones complejas. Algunos ejemplos son: la detección de fuentes de contaminación bioquímica como petróleo o radiación nuclear en el océano [1] y la detección de filtración de sustancias químicas nocivas en el aire [2]. En muchos artículos se intenta abordar el problema con un solo agente, pero estas situaciones se resuelven con mayor eficacia usando grupos de robots, ya que la influencia del ruido en las medidas disminuye como se comenta en [3] y se puede explorar un área más extensa en menor tiempo. Se utilizan vehículos autónomos equipados con un sensor que a partir de las medidas tomadas del campo escalar, son capaces de localizar la fuente y dirigirse hacia ese punto. Debido al contexto del problema, hay varias maneras de hacer esto.

En [4], se utiliza el método de gradiente estocástico. Este consiste en calcular el gradiente utilizando un conjunto aleatorio de las medidas obtenidas en cada iteración. Esto resulta especialmente útil cuando se estudia un campo no convexo o cuando el campo tiene un ruido considerablemente grande que afecta notablemente a las medidas, ya que la aleatoriedad puede ayudar a evitar mínimos o máximos locales. No obstante, en este algoritmo los coeficientes se deben ajustar para cada situación manualmente por motivos de convergencia. Además, en el estudio de las funciones convexas resultan ser menos eficientes, puesto que al utilizar medidas aleatorias, se necesitan más iteraciones para llegar a la fuente.

En [1], en cambio, se utiliza la estadística bayesiana y la entropía Shannon para hallar la fuente de contaminación en el océano. Suponen que las mediciones se hacen en un fluido turbulento, por lo que utiliza la distribución Poisson a priori y a partir de las mediciones de cada agente, se calcula la distribución a posteriori. A partir de la distribución obtenida, se calcula la entropía de ésta y se procede a mover los robots en la dirección en la que el valor de la entropía sea mínima. Este método es útil en situaciones donde el campo escalar que se está analizando se conozca previamente. En el caso de que no sea así, no es posible implementar una distribución de probabilidad acorde al campo y es probable que no se consiga la convergencia.

## 2. Objetivos

El problema en el que se va a enfocar esta memoria es la búsqueda de una fuente que se halla en un campo escalar representado por una función convexa. Para ello, se hace uso de la formación circular simétrica en 2D y 3D para poder estimar el gradiente y el Hessiano con sencillez, tal como se describe en [5].

En este trabajo se analizará la convergencia, la complejidad temporal y espacial de los siguientes métodos: el método del gradiente ascendente con y sin búsqueda de línea inexacta, Newton-Raphson con y sin búsqueda de línea inexacta y los métodos de Quasi-Newton y gradiente conjugado. Se intentará optimizar la búsqueda de la fuente y se compararán los resultados obtenidos para determinar qué método se debería implementar en la búsqueda de la fuente de campos escalares. También se realizará este análisis para el caso en el que haya errores tanto en las medidas como en la formación simétrica.

Por último, cabe destacar que en este trabajo no se considerará la dinámica de los robots, la cual viene descrita en [6] para 2D, sino que se centrará en la optimización del cálculo de la fuente.

### 3. Metodología

Se pretende emplear una formación de agentes para buscar el máximo de un campo escalar convexo. Cada vehículo autónomo tiene un sensor que es capaz de medir el campo escalar  $\sigma(r)$  en el punto  $r$ . Además, comparte la información obtenida con los demás agentes para así poder calcular el gradiente  $\nabla\sigma(c)$  y el Hessiano  $H_{\sigma(c)}$  en el centro de la formación. El máximo del campo escalar se halla en la posición  $r^*$ , donde el gradiente es nulo,  $\nabla\sigma(r^*) = 0$ , mientras que en todos los demás puntos,  $\nabla\sigma(r) \neq 0$ . Asimismo, la función que representa el campo escalar  $\sigma(r) : \mathbb{R}^p \rightarrow \mathbb{R}$  en el punto  $r$ , donde  $p = 2$  para 2D y  $p = 3$  para 3D, debe ser al menos tres veces continuamente diferenciable y sus derivadas parciales hasta orden tres deben estar globalmente acotadas [5].

Si empleamos el desarrollo de Taylor de  $\sigma(r)$  en torno a un punto  $c$ , el error en la aproximación a primer y segundo orden, estará acotado,

$$|\sigma(r) - \sigma(c) - \nabla\sigma(c)^T(r - c)| = \gamma^\nabla(r, c) \leq L\|r - c\|^2 \quad (1)$$

$$|\gamma^\nabla(r, c) - \frac{1}{2}(r - c)^T H_{\sigma(c)}(r - c)| = \gamma^H(r, c) \leq M\|r - c\|^3 \quad (2)$$

donde  $L$  y  $M$  son escalares, y  $\gamma^\nabla(r, c)$  y  $\gamma^H(r, c)$  son, respectivamente, los restos de los desarrollos a primer y segundo orden.

#### 3.1. Problema en 2D

En 2D habrá una formación de  $N$  robots distribuidos uniformemente en una formación circular tal que la posición de cada uno vendrá dada por

$$r_i = c + DR_{\phi_i}e \quad (3)$$

donde  $r_i \in \mathbb{R}^2$  es la posición del agente  $i$ ,  $c \in \mathbb{R}^2$  es el centro de la formación,  $\phi_i = \phi_0 + 2\pi i/N = \omega_0 t + 2\pi i/N$  es el ángulo de rotación ( $\omega_0$  es la velocidad angular),  $R_{\phi_i} = \begin{bmatrix} \cos\phi_i & -\sin\phi_i \\ \sin\phi_i & \cos\phi_i \end{bmatrix}$  es la matriz de rotación asociada a  $\phi_i$ , y  $e = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$ .

##### 3.1.1. Aproximación del gradiente

Para un equipo de  $N \geq 3$  robots distribuidos uniformemente en un círculo con centro  $c$  se hace la aproximación

$$\hat{\nabla}\sigma(c) = \frac{2}{ND^2} \sum_{i=1}^N \sigma(r_i)(r_i - c). \quad (4)$$

A partir de la ecuación (1) y de la simetría de la formación, se concluye en [7] que

$$\|\hat{\nabla}\sigma(c) - \nabla\sigma(c)\| \leq LD \quad (5)$$

donde  $L$  es un escalar,  $D$  es el radio de la formación y  $\nabla\sigma(c)$  es el gradiente exacto. Como se puede observar, el error en la aproximación decrece linealmente con el radio  $D$ .

Además, para  $N \geq 4$  agentes el gradiente de una señal cuadrática se puede calcular exactamente con la ecuación (3), esto es,  $\hat{\nabla}\sigma(c) = \nabla\sigma(c)$ , como se demuestra en [5]. Esto es relevante en la búsqueda de la fuente, ya que cerca de ella, la señal cuadrática es una buena aproximación de la real.

### 3.1.2. Aproximación de la matriz Hessiana

Los robots se disponen de la manera descrita en la ecuación (3), pero se coloca otro vehículo en el centro  $c \in \mathbb{R}^2$ .

En [7] se demuestra que para una cantidad de vehículos autónomos  $N \geq 5$  (sin incluir el robot situado en el centro) se cumple

$$K_{\sigma(c)} = \frac{16}{ND^4} \sum_{i=1}^N (\sigma(r_i) - \sigma(c))(r_i - c)(r_i - c)^T \quad (6)$$

$$\|K_{\sigma(c)} - (3H_{\sigma(c)} + R_{\frac{\pi}{2}}H_{\sigma(c)}R_{\frac{\pi}{2}}^T)\|_F \leq 16MD \quad (7)$$

donde  $M$  es un escalar,  $D$  es el radio de la formación y  $\|\cdot\|_F$  es la norma de Frobenius. A partir de las ecuaciones (2), (6) y la simetría de la formación, como se puede comprobar en [5], se obtiene

$$3\hat{H}_{\sigma(c)} + R_{\frac{\pi}{2}}\hat{H}_{\sigma(c)}R_{\frac{\pi}{2}}^T = K_{\sigma(c)} \quad (8)$$

donde  $K_{\sigma(c)} = 8 \begin{bmatrix} \frac{3}{8}H_{11} + \frac{1}{8}H_{22} & \frac{1}{4}H_{12} \\ \frac{1}{4}H_{12} & \frac{1}{8}H_{11} + \frac{3}{8}H_{22} \end{bmatrix}$  y  $\hat{H}_{\sigma(c)} = \begin{bmatrix} H_{11} & H_{12} \\ H_{12} & H_{22} \end{bmatrix}$ . Por lo tanto, una vez calculado  $K_{\sigma(c)}$ , se puede conseguir fácilmente  $\hat{H}_{\sigma(c)}$ .

El error asociado a esta aproximación, está acotado y viene dado por la expresión

$$\Psi^H(D, c) = \frac{16}{ND^4} \sum_{i=1}^N (r_i - c)\gamma^H(r_i, c)(r_i - c)^T \leq 16MD. \quad (9)$$

Sin embargo, se puede obtener la matriz Hessiana exacta en el centro de la formación analizando una señal cuadrática cuando se tiene  $N \geq 5$  agentes (sin incluir el vehículo en el centro) [5], lo cual será útil para cuando la formación se halle cerca del máximo.

### 3.2. Problema en 3D

Para la aproximación del gradiente se configurará un equipo de  $N = 2n, n \in \mathbb{N}$  vehículos. Formarán una configuración simétrica de dos círculos paralelos cuyos centros están alineados con el eje  $z$ , que tendrán una distancia de  $D\sin\theta_F = D\sqrt{2/3}$  respecto al centro de la formación  $c \in \mathbb{R}^3$ . La mitad de los robots se colocarán en el círculo superior y la otra mitad en el inferior. Matemáticamente, las posiciones en coordenadas esféricas son

$$r_i = c + D \begin{bmatrix} \sin\theta_i \cos\phi_i & \sin\theta_i \sin\phi_i & \cos\theta_i \end{bmatrix}^T \quad (10)$$

donde  $r_i \in \mathbb{R}^3$  es la posición del agente  $i$ ,  $\phi_i = \omega_0 t + 2\pi i/N$  es el ángulo azimutal,  $D$  el radio y  $\theta_i$  viene dada por

$$\theta_i = \begin{cases} \theta_F, & \text{si } i = 2m - 1 \\ \pi - \theta_F, & \text{si } i = 2m \end{cases} \quad (11)$$

donde  $\theta_F$  es el ángulo polar dado por  $\sin\theta_F = \sqrt{2/3}$  y  $m = 1, \dots, n$ .

### 3.2.1. Aproximación del gradiente

Para un equipo de  $N = 2n$  robots con  $n \geq 2$  si se define que el gradiente es

$$\hat{\nabla}\sigma(c) = \frac{3}{ND^2} \sum_{i=1}^N \sigma(r_i)(r_i - c). \quad (12)$$

Entonces se demuestra en [7] que

$$\|\hat{\nabla}\sigma(c) - \nabla\sigma(c)\| \leq 3LD \quad (13)$$

donde  $L$  es un escalar y  $D$  es el radio de la formación.

Para  $n \geq 3$  vehículos el gradiente de una señal cuadrática se puede calcular exactamente con la ecuación (11), los cálculos se detallan en [7].

### 3.2.2. Aproximación de la matriz Hessiana

La formación de los agentes será la misma que la presentada en las ecuaciones (9) y (10), pero al equipo de  $N = 2n$ , se añadirán tres más. Uno de ellos estará en el centro  $c$ , y los otros dos estarán alineados en el eje  $z$  en las posiciones  $\theta_{N+1} = 0$  y  $\theta_{N+2} = \pi$ .

Debido a la simetría de la distribución, si  $n \geq 4$ , entonces

$$K_{\sigma(c)} = \frac{18}{ND^4} \sum_{i=1}^{N+2} (\sigma(r_i) - \sigma(c))(r_i - c)(r_i - c)^T \quad (14)$$

satisface la siguiente expresión en el que los valores  $*$  son los términos simétricos de la matriz,

$$\|K_{\sigma(c)} - \begin{bmatrix} d_1 & * & * \\ H_{12} & d_2 & * \\ 2H_{13} & 2H_{23} & d_3 \end{bmatrix}\|_F \leq \frac{18(N+2)DM}{N} \quad (15)$$

con

$$\begin{aligned} d_1 &= 3H_{11}/2 + H_{22}/2 + H_{33} \\ d_2 &= H_{11}/2 + 3H_{22}/2 + H_{33} \\ d_3 &= H_{11} + H_{22} + H_{33} + 18H_{33}/N \end{aligned} \quad (16)$$

y donde la matriz Hessiana en el centro de la formación viene dada por  $H_{\sigma(c)} = \begin{bmatrix} H_{11} & * & * \\ H_{12} & H_{22} & * \\ H_{13} & H_{23} & H_{33} \end{bmatrix}$ .

De manera similar a como se ha hecho en el resto de los casos, se puede demostrar que se puede obtener la aproximación de la matriz Hessiana  $\hat{H}_{\sigma(c)}$  con

$$K_{\sigma(c)} = \begin{bmatrix} \frac{3}{2}\hat{H}_{11} + \frac{1}{2}\hat{H}_{22} + \hat{H}_{33} & * & * \\ \hat{H}_{12} & \frac{1}{2}\hat{H}_{11} + \frac{3}{2}\hat{H}_{22} + \hat{H}_{33} & * \\ 2\hat{H}_{13} & 2\hat{H}_{23} & \hat{H}_{11} + \hat{H}_{22} + \hat{H}_{33} + \frac{18}{N}\hat{H}_{33} \end{bmatrix}. \quad (17)$$

Al calcular  $K_{\sigma(c)}$ , se puede obtener  $\hat{H}_{\sigma(c)}$  mediante un simple sistema de ecuaciones. Asimismo, la matriz Hessiana exacta de una señal cuadrática se obtiene cuando se tiene  $n \geq 4$  robots [5].

### 3.3. Características de las simulaciones numéricas

Se van a realizar simulaciones numéricas a partir de la aproximación del gradiente y de la matriz Hessiana calculadas anteriormente, para así poder determinar la convergencia y complejidad temporal de los distintos algoritmos propuestos. Se utilizará el siguiente campo escalar

$$\sigma(r) = 100e^{-r^T S r} \quad (18)$$

donde  $S = 10^{-4} \begin{bmatrix} 2 & 1 \\ 1 & 1 \end{bmatrix}$  para 2D y  $S = 10^{-4} \begin{bmatrix} 100 & 1 & 1 \\ 1 & 1 & 0 \\ 1 & 0 & 10 \end{bmatrix}$  para 3D. El máximo de este campo escalar se halla en el origen; por lo tanto, el centro de la formación deberá dirigirse a este punto. Se ha utilizado específicamente este campo para así poder comparar los resultados obtenidos con los del artículo [5]. Asimismo, para el problema en 2D y 3D se utilizarán el número de agentes  $N$  mínimos para que se pueda implementar el algoritmo correspondiente, esto es, el número de robots  $N$  que hace falta para calcular exactamente en el centro de la formación el gradiente o el Hessiano de una señal cuadrática. Por lo que para los algoritmos donde solo sea necesario estimar el gradiente en el centro  $\hat{\nabla}\sigma_c$  se tomarán  $N = 4$  en 2D y  $N = 6$  en 3D, mientras que para el caso de las estimaciones de la matriz Hessiana, se tomarán  $N = 5$  en 2D y  $N = 8$  en 3D. En todos los algoritmos el centro de formación de los robots se colocará inicialmente en la posición  $c_0 = [20 \ 20]^T$  en 2D y  $c_0 = [20 \ 20 \ 20]^T$  en 3D, para así poder compararlos equitativamente. Además, por sencillez y para que el radio de la formación no jugara un papel relevante en la convergencia cuando el ruido no está presente, se ha tomado  $D = 1m$  en estos casos.

En las aplicaciones de estos métodos en la vida real pueden producirse problemas debido al ruido en las mediciones del campo escalar o a la perturbación de la posición de los vehículos respecto a la formación simétrica deseada. Es por esto por lo que además de las simulaciones sin ruido, también se calculará cómo afectan estos factores a la búsqueda de la fuente. Se aplicará solamente al problema en 2D, ya que en 3D será un caso similar. Se asumirá que las medidas del campo escalar estudiado vienen afectadas por un ruido blanco Gaussiano independiente en cada robot, el cual tiene una función de densidad que responde a una distribución normal con media nula:

$$y(r_i) = \sigma(r_i) + v_i(r_i), \quad v_i(r_i) \sim \mathcal{N}(0, v^2) \quad (19)$$

donde  $v^2$  es la varianza del ruido. En lo que corresponde a la perturbación en la posición, también vendrá dada por una distribución normal con media nula:

$$r_i^{real} = r_i + \delta_i(r_i), \quad \delta_i(r_i) = \begin{bmatrix} \delta_{xi} \\ \delta_{yi} \end{bmatrix} \quad (20)$$

donde  $\delta_{xi}, \delta_{yi} \sim \mathcal{N}(0, \delta^2)$ . En todas las simulaciones numéricas calculadas se ha supuesto  $v = 2$  y  $\delta = 0,1m$ . Cabe destacar que en estas situaciones se ha decidido tomar  $D = 5m$  para todos los algoritmos presentados en este trabajo, debido a la repercusión que este valor tiene en la desviación estándar de cada método. Esto se analizará con más detalle en las siguientes secciones.

En la representación de la trayectoria de los vehículos también se ha representado la trayectoria del centro de la formación con puntos rojos y el punto inicial  $c_0$  con un círculo azul.

## 4. Resultados

### 4.1. Método del gradiente ascendente

A partir de las aproximaciones del gradiente en el centro obtenidas para 2D en la ecuación (4) y para 3D en la ecuación (12), se halla la fuente del campo escalar calculando iterativamente la



siguiente posición del centro de la formación a partir de la expresión

$$c_{k+1} = c_k + \alpha \hat{\nabla} \sigma(c_k). \quad (21)$$

Se puede ver que desde el punto  $c_k$  se determina el siguiente punto  $c_{k+1}$  a partir de la dirección (estimada) de máximo ascenso, que viene dada por el valor del gradiente  $\hat{\nabla} \sigma(c_k)$ . Por otro lado, la convergencia de este método está determinada por el valor  $\alpha$ . Cuanto más grande sea este valor, antes convergerá el método; no obstante, si resulta ser demasiado grande, el método podría oscilar en torno a la fuente, haciéndolo más ineficiente e incluso llegando a divergir. Con el objetivo de evitar este problema, se ha tomado  $\alpha = 0,1$  en todos los casos.

Para poder aplicar el método del gradiente ascendente, es necesario que la función estudiada sea convexa y diferenciable, ya que en caso de que no lo sea, no se podría computar satisfactoriamente su gradiente y cabe la posibilidad de que converja a máximos locales. Sin embargo, al elegir el campo escalar dado por la ecuación (18), el único problema que puede surgir es la lenta convergencia del método.

En las Figuras 1(a) y 2(a) se representa la trayectoria de los robots en 2D y 3D, respectivamente. En las Figuras 1(b) y 2(b), en cambio, la distancia del centro de la formación a la fuente frente a las iteraciones.

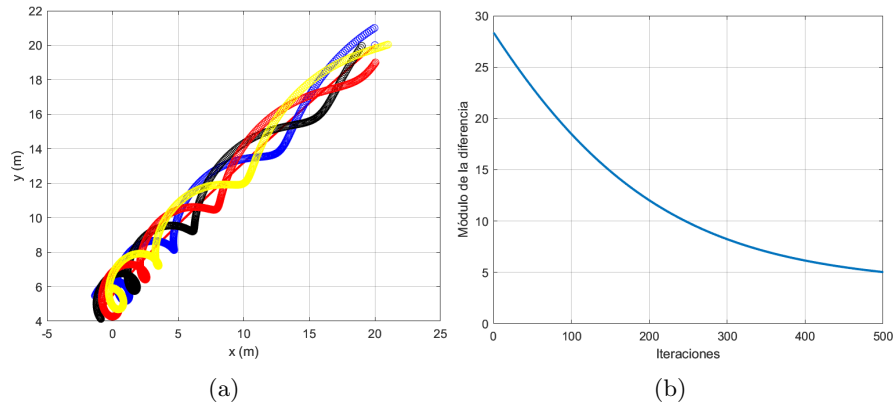


Figura 1: (a) Trayectoria de los robots en 2D con el método del gradiente ascendente. (b) Evolución de la distancia entre el centro de la formación y la fuente con el método del gradiente ascendente en 2D.

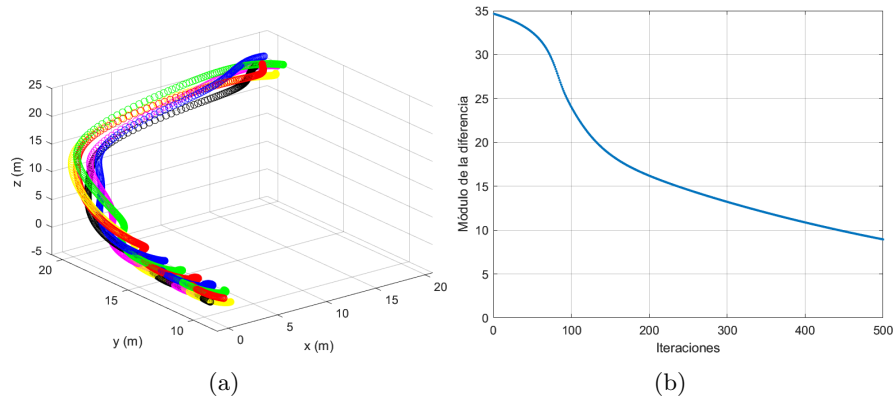


Figura 2: (a) Trayectoria de los robots en 3D con el método del gradiente ascendente. (b) Evolución de la distancia entre el centro de la formación y la fuente con el método del gradiente ascendente en 3D.

La complejidad temporal por iteración es  $\mathcal{O}(n)$ , ya que para computar el gradiente en  $c$ ,  $\hat{\nabla}\sigma(c)$ , el número de operaciones crece linealmente con el número de agentes  $N$ , como se puede ver en las ecuaciones (4) y (12).

Se ha podido calcular directamente el escalar  $L$  que determina cuánto varía la estimación del gradiente en el centro de su valor real. Para obtener estos valores, se ha calculado el gradiente exacto en cada punto donde se ha hallado el centro de la formación  $c_k$  utilizando funciones internas de Matlab, mientras que para las aproximaciones del gradiente se ha hecho uso de las ecuaciones (4) y (12). Una vez calculados estos valores, con las ecuaciones (5) y (13), se halla el valor  $L$  para cada iteración y se elige el valor máximo entre estos, ya que debe ser una cota superior. Los escalares obtenidos son  $L_{2D} = 2,63 \cdot 10^{-4}$  y  $L_{3D} = 9,1 \cdot 10^{-3}$ , por lo que se puede concluir que las estimaciones calculadas son satisfactorias. Además, al aumentar la cantidad de agentes  $N$ , se puede observar que el número de iteraciones que necesita el método para converger no varía drásticamente, puesto que la bondad de la estimación del gradiente no mejora de manera significativa.

En lo que respecta al análisis del ruido y de la perturbación de los robots respecto a su posición definida, se concluye que cuanto mayor sea el radio  $D$  y el número de vehículos autónomos  $N$ , menor será la desviación estándar, como se puede comprobar aplicando los cálculos hechos en [7] en dos dimensiones:

$$s_{\nabla} = \frac{2v}{N^{1/2}D}. \quad (22)$$

Debido a esto, se ha decidido aumentar el radio a  $D = 5m$ . Se han hecho 100 simulaciones para así calcular la media y la desviación estándar, obteniéndose así la Figura 3. A pesar de estas circunstancias, la formación es capaz de converger.

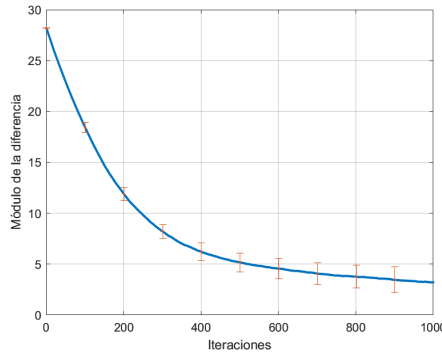


Figura 3: Media y desviación estándar después de 100 simulaciones con  $v = 2$  y  $\delta = 0,1m$  de la distancia entre el centro de la formación y la fuente en 2D con el método del gradiente ascendente.

## 4.2. Método de Newton-Raphson

En este caso se hace uso tanto de la estimación del gradiente  $\hat{\nabla}\sigma(c)$  como de la estimación de la matriz Hessiana  $\hat{H}_{\sigma(c)}$  para la búsqueda de la fuente:

$$c_{k+1} = c_k - \alpha \hat{H}_{\sigma(c_k)}^{-1} \hat{\nabla}\sigma(c_k). \quad (23)$$

En el caso de que  $\hat{H}_{\sigma(c)}$  se aproxime correctamente, el método es más efectivo que el del gradiente, puesto que converge a la fuente cuadráticamente en vez de linealmente. No obstante, hay que tener en cuenta que el cálculo del Hessiano como su inversión es computacionalmente costoso e incluso en algunos casos, podría no estar siquiera definido. Asimismo, en puntos lejanos al centro de la fuente es probable que la matriz Hessiana sea positiva semidefinida, alejando la formación del

punto al que queremos que llegue. Por ello, en caso de que la matriz no esté bien condicionada o que sea positiva semidefinida, se aplica el método del gradiente ascendente hasta poder aproximar el Hessiano satisfactoriamente.

Para implementar esta técnica se ha de hacer la aproximación de la matriz  $K_{\sigma(c)}$  para dos y tres dimensiones, en las que el número de operaciones es proporcional a  $N$  y  $N + 2$ , respectivamente. Después, se halla la matriz Hessiana a partir de la ecuación (9) en 2D y de la ecuación (17) en 3D, lo cual requiere de calcular la solución de un sistema de ecuaciones lineales. Además, se debe obtener la inversa de esta matriz, lo cual puede resultar computacionalmente costoso dependiendo del problema. En este caso, al ser las matrices de dimensiones  $2 \times 2$  y  $3 \times 3$ , se puede hallar la solución exacta con  $\mathcal{O}(n^3)$  operaciones, donde  $n$  es la dimensión de la matriz cuadrada. Por lo tanto, no ha habido grandes problemas al aplicar esta técnica para la búsqueda de fuente.

Como en el caso del método del gradiente ascendente, la convergencia depende del tamaño de paso  $\alpha$ . Para hacer una comparación justa entre las distintas simulaciones este valor seguirá siendo  $\alpha = 0,1$ . Los cálculos numéricos paran cuando el centro de la formación se encuentre a una distancia menor de  $0,001m$  del centro de la fuente. En las Figuras 4(a) y 5(a) se muestra la trayectoria de los vehículos autónomos en 2D y 3D, mientras que en las Figuras 4(b) y 5(b), se representa la evolución de la distancia entre el centro y la fuente.

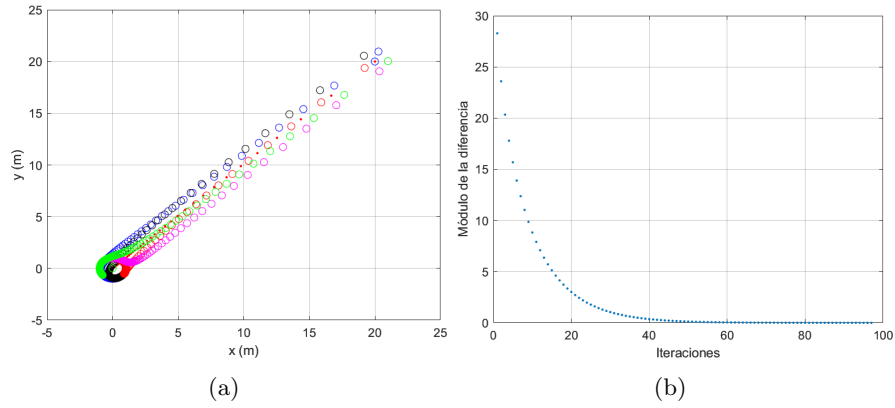


Figura 4: (a) Trayectoria de los robots en 2D con el método de Newton-Raphson. (b) Evolución de la distancia entre el centro de la formación y la fuente con el método de Newton-Raphson en 2D.

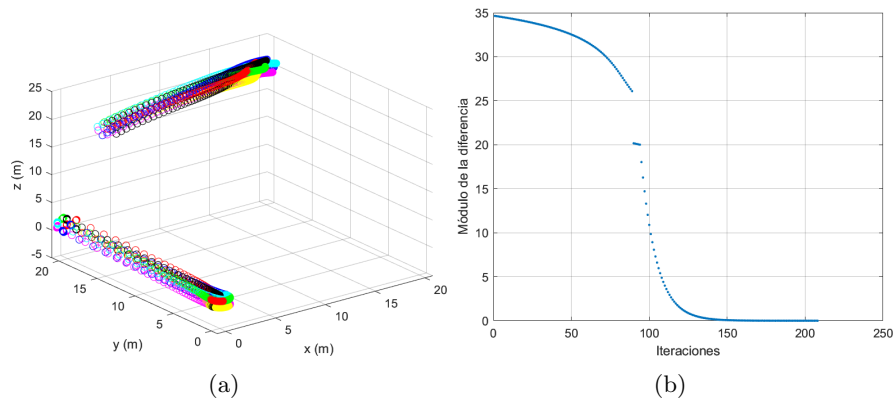


Figura 5: (a) Trayectoria de los robots en 3D con el método de Newton-Raphson. (b) Evolución de la distancia entre el centro de la formación y la fuente con el método de Newton-Raphson en 3D.

Se puede observar que en 3D se utiliza el método del gradiente ascendente hasta llegar a un punto en el que se puede aplicar el método de Newton-Raphson debido a las razones explicadas anteriormente. Esto se refleja claramente en la Figura 5(a) donde el uso del Hessiano provoca que haya un cambio brusco en la posición de los robots entre una iteración y otra. Esta transición ocurre en la iteración 95, como se puede apreciar en la Figura 5(b). Este no ha sido el caso en 2D, donde la convergencia ha sido mucho más rápida que en el método anterior.

Por otro lado, de manera similar al método del gradiente, se ha podido calcular el escalar  $M$  de las ecuaciones (7) y (15) para comprobar si las aproximaciones hechas son suficientemente cercanas a los valores reales. Se ha obtenido  $M_{2D} = 1,4 \cdot 10^{-5}$  y  $M_{3D} = 1,3 \cdot 10^{-3}$ , así que las aproximaciones son buenas y se ha podido comprobar que el número  $N$  de agentes no tiene mucho impacto en la búsqueda de la fuente.

Al hacer los cálculos numéricos con el ruido en las medidas y la perturbación en las posiciones de los agentes, se ha vuelto a determinar  $D = 5m$  para mitigar la desviación estándar:

$$s_H = \frac{16v}{ND^2}(N^2 + N)^{1/2} \simeq \frac{16v}{D^2}. \quad (24)$$

Se vuelven a hacer 100 simulaciones con  $v = 2$  y  $\delta = 0,1m$  en 2D y se consigue la Figura 6.

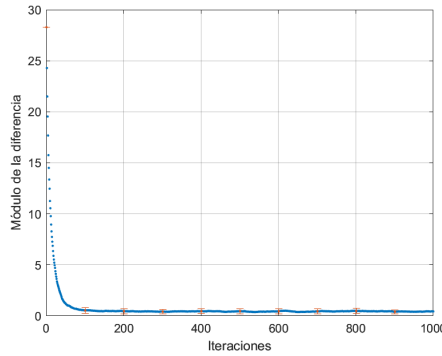


Figura 6: Media y desviación estándar después de 100 simulaciones con  $v = 2$  y  $\delta = 0,1m$  de la distancia entre el centro de la formación y la fuente en 2D con el método de Newton-Raphson.

### 4.3. Métodos con búsqueda de línea

Los métodos con búsqueda de línea se basan en calcular iterativamente el valor ideal del tamaño de paso  $\alpha_k$  a partir de la dirección de ascenso calculada  $p_k$  para acelerar la convergencia. Se podría obtener el valor de  $\alpha_k$  de manera exacta resolviendo la siguiente ecuación:

$$\alpha_k = \arg \max_{\alpha} \sigma(c_k + \alpha_k p_k). \quad (25)$$

Se ha determinado en [8] que calcular el tamaño de paso de forma exacta puede resultar muy costoso computacionalmente y esto resulta en que el algoritmo tardaría un tiempo considerable en converger. En consecuencia, se ha decidido utilizar una búsqueda de línea inexacta, que consiste en hallar un  $\alpha_k$  que satisface ciertas condiciones. Para ello, primero se halla un intervalo que contenga valores del tamaño de paso ideales (se empieza por  $\alpha_0 = 1$ ), y a partir de interpolación se calcula un valor aceptable.

Entre las condiciones más populares se encuentran las condiciones Goldstein y las condiciones Wolfe [8]. Las condiciones Goldstein pueden no estar bien adecuadas para el método de Newton-Raphson, porque es necesario que la aproximación de la matriz Hessiana no sea positiva semidefinida

en cada iteración y estas condiciones no lo pueden asegurar. Por ello, se ha hecho uso de las condiciones Wolfe fuertes.

En primer lugar, se debe satisfacer la condición de suficiente ascenso, también llamada condición Armijo:

$$\sigma(c_k + \alpha_k p_k) \geq \sigma(c_k) + c_1 \alpha_k \nabla \sigma(c_k)^T p_k \quad (26)$$

en la que  $c_1 \in (0, 1)$  es una constante. En funciones unidimensionales se puede interpretar como crear una recta con pendiente positiva desde el punto  $\sigma(c_k)$  para asegurar que  $\sigma(c_k + \alpha_k p_k)$  se encuentre por encima de esta línea.

Cuanto mayor sea la constante  $c_1$ , más difícil será cumplir esta condición, por lo que normalmente se elige un valor  $c_1$  cercano a 0. En los siguientes casos se ha determinado  $c_1 = 10^{-4}$ .

No obstante, utilizando solo esta condición se pueden elegir valores de  $\alpha_k$  que no sean suficientemente grandes como para que haya un progreso sustancial en la búsqueda del máximo. Es por esto por lo que se implementa también la condición de curvatura. Esta requiere que

$$\nabla \sigma(c_k + \alpha_k p_k)^T p_k \leq c_2 \nabla \sigma(c_k)^T p_k \quad (27)$$

donde  $c_2 \in (c_1, 1)$ . Con esta condición se logra que la derivada en el punto  $\sigma(c_k + \alpha_k p_k)$  sea menor que en el punto donde se inicia el estudio  $\sigma(c_k)$  multiplicado por la constante  $c_2$ . De esta manera, si  $\nabla \sigma(c_k + \alpha_k p_k)^T p_k$  resulta ser muy positiva, tiene sentido seguir buscando un valor  $\alpha_k$  mejor en esa dirección de ascenso  $p_k$ , ya que la función ascenderá significativamente. En cambio, si la derivada es apenas positiva o incluso negativa, se detiene el proceso de búsqueda de  $\alpha_k$ , ya que la función no ascenderá en esa dirección.

Las dos condiciones hasta ahora descritas combinadas forman las condiciones Wolfe. Sin embargo, la condición de curvatura descrita previamente puede provocar que acabemos en un punto lejano al máximo al que se quiere llegar. Para lidiar con este problema, se modifica esta condición de la siguiente manera:

$$|\nabla \sigma(c_k + \alpha_k p_k)^T p_k| \leq c_2 |\nabla \sigma(c_k)^T p_k|. \quad (28)$$

Con este requisito se asegura que no se estudien puntos que se puedan encontrar alejados de los máximos, ya que no permite que la derivada en estos puntos lleguen a ser muy negativos o positivos. A esta condición junto la condición Armijo se le denomina las condiciones Wolfe fuertes. Una de las principales ventajas de este método es que al reducir el valor de la constante  $c_2$ , la búsqueda de  $\alpha_k$  es de mayor calidad, al forzar que el punto  $\sigma(c_k + \alpha_k p_k)$  esté más cerca del mínimo. Según [8], se suele usar  $c_2 = 0,1$  para el método del gradiente conjugado y  $c_2 = 0,9$  para el método Quasi-Newton.

Se ha decidido implementar estas condiciones en la búsqueda del tamaño de paso  $\alpha_k$  óptimo en cada iteración para acelerar la convergencia en los métodos del gradiente ascendente y Newton-Raphson. También se ha utilizado para calcular  $\alpha_k$  para los métodos del gradiente conjugado y de Quasi-Newton, que se describirán más adelante (estos dos últimos métodos se han estudiado siempre incluyendo búsqueda de línea).

Una de las principales desventajas de aplicar esta técnica para calcular el tamaño de paso adecuado es que el ruido en la medida del campo escalar y en la posición de los robots afecta enormemente a la convergencia, puesto que las condiciones antes explicadas varían de valor en cada iteración para un mismo punto. El principal problema radica en el error que se produce en el

cálculo del gradiente en el centro de la formación, ya que este valor se utiliza en las dos condiciones y además, es necesario para determinar el siguiente centro de la formación  $c_{k+1}$  para todos los procedimientos que aparecen en este trabajo.

La solución propuesta a este dilema ha sido realizar varias aproximaciones del gradiente sin mover el centro de la formación de sitio. Al ser la media del ruido de la medida del campo escalar y la perturbación de la posición de los vehículos autónomos igual a 0, si se hace la media de las estimaciones, es probable que el gradiente estimado se acerque lo suficiente al gradiente exacto como para que no surjan problemas serios al aplicar estos métodos.

En [9] se hace uso de un método similar. Aseguran mejorar la precisión y eficiencia de la búsqueda al mover la formación entera a dos posiciones distintas relativamente cercanas y hacer una media entre los gradientes medidos en cada posición. En el caso que se estudia en este trabajo, no hará falta mover la formación entera en torno a otro punto para calcular la nueva aproximación del gradiente, ya que al estar los robots en constante movimiento rotatorio alrededor del centro, se pueden calcular nuevas estimaciones constantemente.

Los cálculos con este algoritmo se han realizado en dos dimensiones solamente, ya que para tres dimensiones funciona de la misma manera. Asimismo, también se han hecho cálculos numéricos para determinar cómo afectaría el ruido en las medidas del campo escalar y la posición de los agentes en la búsqueda de la fuente, como se hizo en las secciones 4.1. y 4.2. Aunque al implementar las condiciones de Wolfe fuertes a éstas, se ha reducido el número de simulaciones a 5 para reducir el tiempo de ejecución.

#### **4.3.1. Método del gradiente ascendente con búsqueda de línea**

En la sección 4.1. se ha utilizado la técnica del gradiente ascendente con un tamaño del paso constante. Con la ayuda de las condiciones Wolfe fuertes, se espera mejorar el rendimiento de este método.

Para una mayor eficacia del algoritmo se ha considerado  $c_2 = 0, 1$ . El resto de las variables son exactamente iguales a las descritas en la sección 4.1., para así poder hacer una comparación justa entre los dos algoritmos. La trayectoria de los robots en 2D y la evolución de la distancia entre el centro de la formación se muestran en las Figuras 7(a) y 7(b), donde se muestra que el método converge mucho antes con la búsqueda de línea inexacta, como era de esperar.

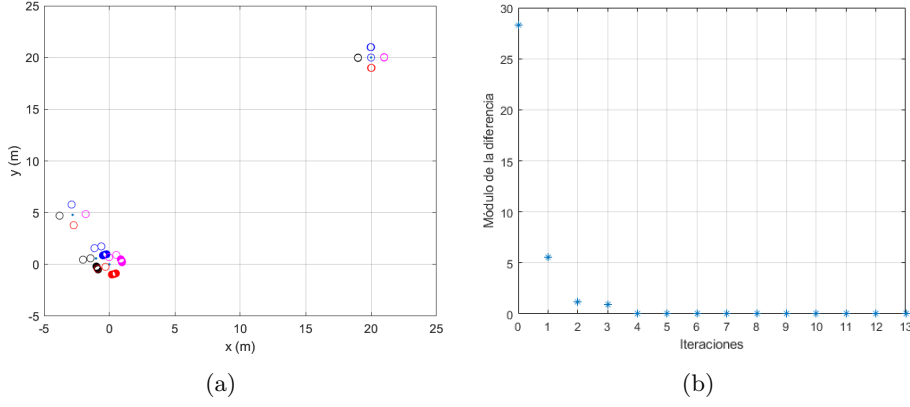


Figura 7: (a) Trayectoria de los robots en 2D con el método del gradiente ascendente con búsqueda de línea. (b) Evolución de la distancia entre el centro de la formación y la fuente con el método del gradiente ascendente con búsqueda de línea en 2D.

En lo que respecta al análisis del ruido, la diferencia en el tiempo de ejecución respecto al método sin búsqueda de línea es notable, y lo será también en el resto de los métodos. Esto es debido tanto al estudio de las condiciones Wolfe fuertes como a la necesidad de calcular la media del gradiente en cada punto en el que se encuentra el centro de la formación  $c_k$ . La media de la distancia entre el centro y la fuente y la desviación estándar después de 5 simulaciones se expone en la Figura 8.

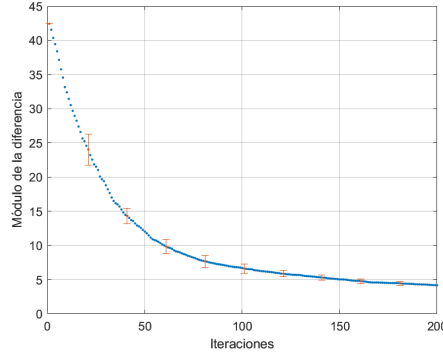


Figura 8: Media y desviación estándar después de 5 simulaciones con  $v = 2$  y  $\delta = 0,1m$  de la distancia entre el centro de la formación y la fuente en 2D utilizando el método del gradiente ascendente con búsqueda de línea.

#### 4.3.2. Método de Newton-Raphson con búsqueda de línea

En el caso del método de Newton-Raphson, también es lógico esperar una mejoría notable en la eficiencia de la búsqueda de la fuente, sobre todo en el caso donde el ruido es despreciable. La constante para la condición de curvatura elegida ha sido  $c_2 = 0,9$ , ya que a pesar de que un valor menor de  $c_2$  supondría generalmente una mejor convergencia, después de varias observaciones computacionales se ha determinado en [8] que esos son los valores más eficientes para este determinado problema.

Después de hacer el cálculo numérico correspondiente para este algoritmo, se obtienen las Figuras 9(a) y 9(b), donde se exponen la trayectoria de los vehículos autónomos y la distancia de su centro de formación a la fuente. Aunque pueda haber una sobreestimación del tamaño de paso al hacer su cálculo de manera inexacta, la formación se acerca a la fuente en cada iteración y converge de manera eficaz.

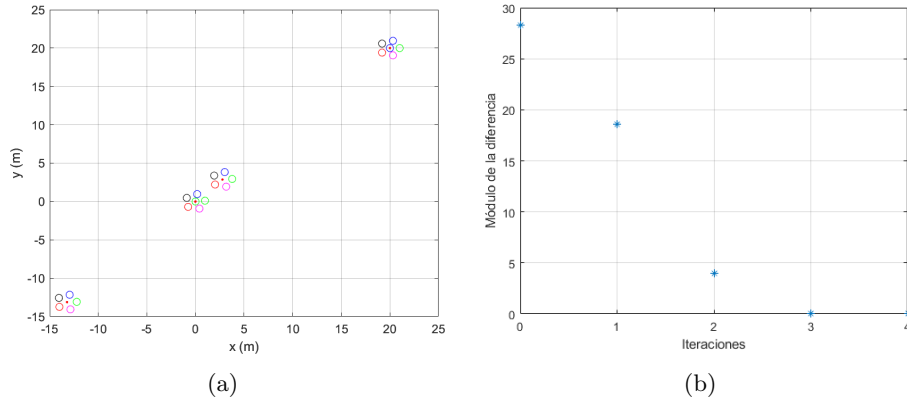


Figura 9: (a) Trayectoria de los robots en 2D con el método de Newton-Raphson con búsqueda de línea. (b) Evolución de la distancia entre el centro de la formación y la fuente con el método de Newton-Raphson con búsqueda de línea en 2D.

Es importante destacar la influencia que tiene el ruido en este método, ya que no solo afecta considerablemente a la aproximación del gradiente, sino también a la de la matriz Hessiana. Al igual que con el gradiente, se va a proceder a tomar varias medidas en torno al mismo punto y a calcular la media de estas, lo cual hace que sea más costoso computacionalmente. Los resultados obtenidos están en la Figura 10.

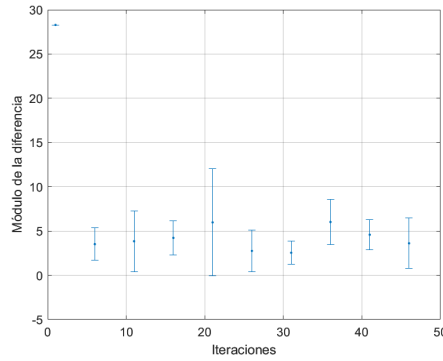


Figura 10: Media y desviación estándar después de 5 simulaciones con  $v = 2$  y  $\delta = 0,1m$  de la distancia entre el centro de la formación y la fuente en 2D utilizando el método de Newton-Raphson con búsqueda de línea.

A primera vista, sería lógico argumentar que este método es muy efectivo incluso con medidas defectuosas, puesto que en tan solo 50 iteraciones la formación se acerca satisfactoriamente a la fuente. Pero si se examina el número de veces en las que se ha tenido que implementar las condiciones Wolfe fuertes para obtener el tamaño de paso adecuado  $\alpha_k$  para cada iteración, se puede ver que se ha ejecutado de media en 62.326 ocasiones para cada una de las 5 simulaciones hechas. Esto sumado a la necesidad de la aproximación y la inversión de la matriz Hessiana, podría suponer un problema al intentar aplicar este método en la vida real.

#### 4.3.3. Método del gradiente conjugado con búsqueda de línea

El método del gradiente conjugado es similar al del gradiente convencional, con la diferencia de que la dirección de ascenso va variando iterativamente. En la primera iteración la dirección de ascenso es  $p_0 = \hat{\nabla}\sigma(c_0)$ , pero para las siguientes direcciones se debe calcular en primer lugar el siguiente centro de la formación

$$c_{k+1} = c_k + \alpha_k p_k \quad (29)$$



para así poder calcular la siguiente dirección de ascenso  $p_{k+1}$

$$p_{k+1} = \hat{\nabla}\sigma(c_{k+1}) + \beta_k p_k \quad (30)$$

donde  $\beta_k$  viene dado por

$$\beta_k = \frac{\hat{\nabla}\sigma(c_{k+1})^T(\hat{\nabla}\sigma(c_{k+1}) - \hat{\nabla}\sigma(c_k))}{\hat{\nabla}\sigma(c_k)^T \hat{\nabla}\sigma(c_k)}. \quad (31)$$

A esta variante del gradiente conjugado se le llama método Polak-Ribière. Es uno de los métodos más populares junto con el de Fletcher-Reeves. Estos son idénticos cuando el campo escalar es fuertemente convexo, cuadrático y la búsqueda de línea es exacta. Para búsquedas de línea inexactas, en cambio, las pruebas numéricas hechas indican que el algoritmo Polak-Ribière es más robusto y eficiente que el de Fletcher-Reeves [8].

Tiene una complejidad  $\mathcal{O}(n)$  como el método del gradiente ascendente, sin embargo, la determinación de la dirección ascendente es más eficiente y por esta razón suele converger antes, como se ve en las Figuras 11(a) y 11(b).

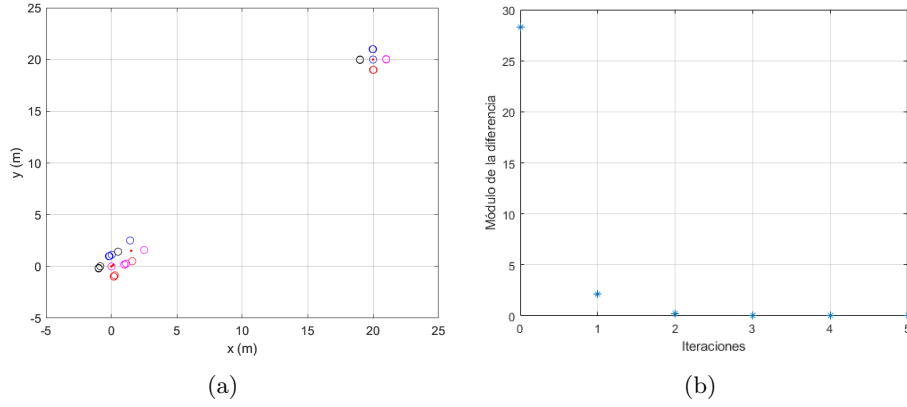


Figura 11: (a) Trayectoria de los robots en 2D con el método del gradiente conjugado con búsqueda de línea. (b) Evolución de la distancia entre el centro de la formación y la fuente con el método del gradiente conjugado con búsqueda de línea en 2D.

Cuando el ruido está presente, en cambio, se obtiene la Figura 12 para las 5 simulaciones realizadas.

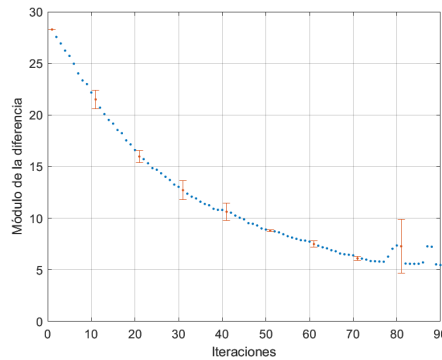


Figura 12: Media y desviación estándar después de 5 simulaciones con  $v = 2$  y  $\delta = 0,1m$  de la distancia entre el centro de la formación y la fuente en 2D utilizando el método del gradiente conjugado con búsqueda de línea.

El ruido en las medidas tiene una influencia directa en el cálculo de la dirección de descenso  $p_k$ . Además, como en el resto de los casos en los que se ha aplicado la búsqueda de línea inexacta

con métodos Wolfe fuertes, hace que la determinación de  $\alpha_k$  sea más ineficiente. Se puede apreciar que a medida que la formación se acerca a la fuente, al ser el gradiente más cercano a cero, es más probable que las direcciones de ascenso difieran mucho del valor anterior, incluso convirtiéndose en direcciones de descenso. Asimismo, en puntos muy lejanos al máximo, se podría dar la misma situación.

#### 4.3.4. Método Quasi-Newton con búsqueda de línea

El método Quasi-Newton consigue una convergencia superlineal a partir de las medidas del gradiente, ya que aproxima la inversa de la matriz Hessiana iterativamente. Cada iteración puede ser ejecutada con una complejidad temporal de  $\mathcal{O}(n^2)$ . A pesar de que el método de Newton-Raphson converja más rápidamente, el coste por iteración es más grande, ya que ha de solucionarse un sistema de ecuaciones lineales y hallar la inversa de la matriz Hessiana. Por este motivo, en algunas ocasiones aplicar el procedimiento Quasi-Newton resulta ser más eficiente.

Hay varios algoritmos que se pueden implementar para este procedimiento, pero el más popular de ellos es el método BFGS. Una de las principales diferencias con otros algoritmos como DFP es que si en algún momento la aproximación de la matriz Hessiana resulta ser errónea, se ha demostrado que la ecuación utilizada para BFGS es más efectiva para corregir esta mala estimación en unas pocas iteraciones [8].

En este método se observa el comportamiento que tiene el campo escalar y el gradiente en dos puntos consecutivos para adquirir información de la curvatura y aproximar iterativamente el Hessiano. En la primera iteración, se toma la aproximación de la matriz Hessiana como  $B_0 = \beta I$ , donde  $I$  es la matriz identidad y  $\beta$  una constante. El valor ideal de  $\beta$  varía dependiendo del problema propuesto, así que se ha probado con  $\beta = -1$  en este caso. Cabe destacar que en un problema de maximización,  $B_0$  debe ser negativo definido, cosa que se cumple con el valor de  $\beta$  propuesto. El procedimiento en cada iteración a partir de la primera se fundamenta en calcular el siguiente punto donde se debe encontrar el centro de la formación

$$c_{k+1} = c_k + \Delta c_k = c_k - \alpha_k B_k^{-1} \hat{\nabla} \sigma(c_k). \quad (32)$$

Después de esto se calcula el gradiente en el punto  $c_{k+1}$  y se determina  $y_k = \hat{\nabla} \sigma(c_{k+1}) - \hat{\nabla} \sigma(c_k)$ . Por último, se computa la inversa de la aproximación de la matriz Hessiana para la siguiente iteración  $B_{k+1}^{-1}$ :

$$B_{k+1}^{-1} = \left( I - \frac{\Delta c_k y_k^T}{y_k^T \Delta c_k} \right) B_k^{-1} \left( I - \frac{y_k \Delta c_k^T}{y_k^T \Delta c_k} \right) + \frac{\Delta c_k \Delta c_k^T}{y_k^T \Delta c_k}. \quad (33)$$

En caso de que la formación no se encuentre lo suficientemente cerca de la fuente, se repite el procedimiento volviendo a la ecuación (32).

Al igual que en el método de Newton-Raphson con búsqueda de línea, es recomendable utilizar los parámetros  $c_1 = 10^{-4}$  y  $c_2 = 0,9$ , ya que así los cálculos son más eficientes computacionalmente.

A pesar de que no se ha estudiado cuál es el valor óptimo de  $B_0$  para la convergencia de la formación de robots, se puede comprobar que este método funciona eficientemente. En la Figura 13(a) se ve que hay una sobreestimación del tamaño de paso, aun así, esto no conlleva a que el método diverja, sino que la formación va acercándose iterativamente al punto que queremos, como se observa en la Figura 13(b).

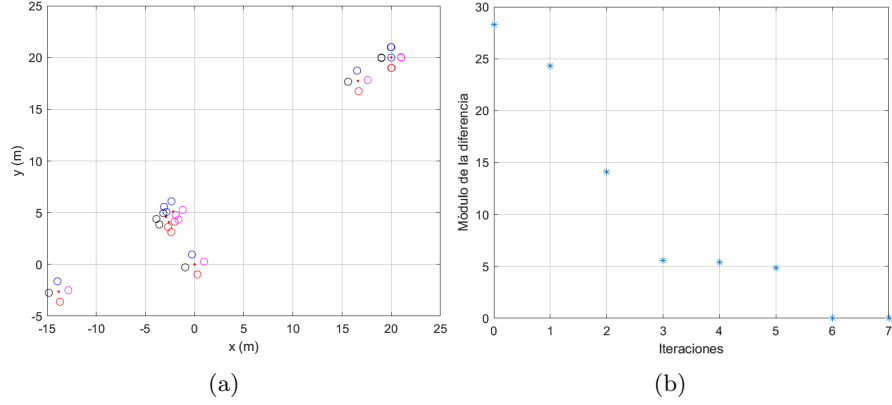


Figura 13: (a) Trayectoria de los robots en 2D con el método Quasi-Newton con búsqueda de línea. (b) Evolución de la distancia entre el centro de la formación y la fuente con el método Quasi-Newton con búsqueda de línea en 2D.

En las medidas con ruido y error en las posiciones de los agentes, en cambio, se puede observar un comportamiento similar al método de Newton-Raphson con búsqueda de línea. Como se muestra en la Figura 14, en pocas iteraciones la formación se acerca considerablemente al origen, pero después se necesitan más iteraciones para aproximarse satisfactoriamente a la localización deseada.

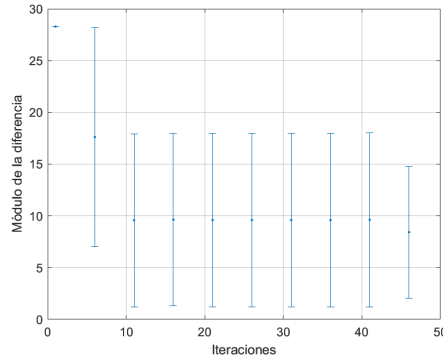


Figura 14: Media y desviación estándar después de 5 simulaciones con  $v = 2$  y  $\delta = 0,1m$  de la distancia entre el centro de la formación y la fuente en 2D utilizando el método Quasi-Newton con búsqueda de línea.

Además, es obvio que el valor de la desviación estándar es mayor que en los demás casos. Esto es a causa de que el error en la medida provoca que las aproximaciones de las inversas de la matriz Hessiana varíen notablemente de una simulación a otra. Se puede disminuir esta variación fácilmente aumentando el radio  $D$  de la formación. Una mejor opción sería aumentar el número de robots  $N$ , ya que además de la disminución de la desviación típica, también mejora considerablemente la convergencia.

## 5. Conclusiones

En este trabajo se han analizado opciones para la búsqueda de fuente de una señal para 2D y 3D con una formación de agentes de simetría circular. Se han estudiado los casos propuestos en el artículo [5] (gradiente ascendente y ascenso tipo Newton-Raphson), esos dos métodos con búsqueda de línea, el método del gradiente conjugado y el de Quasi-Newton.

En el caso del algoritmo del gradiente ascendente se puede concluir que a pesar de que es un procedimiento robusto incluso para medidas ruidosas y errores en las posiciones de los robots, re-

sulta ser el menos eficiente de los algoritmos estudiados por su convergencia lineal. No obstante, su complejidad temporal y espacial es  $\mathcal{O}(n)$ , lo cual facilita la implementación de esta técnica en aplicaciones del mundo real. Al aplicarle a este algoritmo la búsqueda de línea inexacta con condiciones Wolfe fuertes, se aprecia una mejoría en la eficiencia cuando el ruido es despreciable (más adelante se hace el estudio de las medidas con ruido para los métodos con búsqueda de línea), ya que se hallan tamaños de paso ideales para cada iteración.

El procedimiento más similar a los dos anteriores es el del gradiente conjugado. También tiene una complejidad espacial y temporal  $\mathcal{O}(n)$ , pero para situaciones en los que el ruido es despreciable, converge mucho más eficientemente. Esto es debido no solo a la búsqueda de línea, sino también a la determinación iterativa de la dirección de búsqueda  $p_k$ , que tiene en cuenta también la dirección de ascenso anterior además del gradiente en el punto.

El método Newton-Raphson tiene la principal ventaja de que converge cuadráticamente cuando se puede aplicar. En caso de que la matriz Hessiana no esté bien condicionada o sea positiva semidefinida, se aplica el gradiente ascendente. Esto hace que este algoritmo sea robusto y eficiente incluso para las simulaciones con el campo escalar ruidoso. Si se le añade la búsqueda de línea, el algoritmo es todavía más eficiente cuando las medidas del campo escalar no están afectadas por errores. Hay que tener en cuenta que la complejidad temporal de este algoritmo, al resolver el sistema de ecuaciones lineales y la inversa de la matriz de manera exacta, es de  $\mathcal{O}(n^3)$  y la complejidad espacial, en cambio, es de  $\mathcal{O}(n^2)$ . La necesidad de tener que hacer el cálculo del Hessiano y su inversa en cada iteración puede dificultar su implementación en aplicaciones reales.

Una alternativa a este procedimiento es el método Quasi-Newton, el cual también hace uso de la curvatura de la función que se estudia, pero no requiere del cálculo de la matriz Hessiana ni de su inversión. Por ello, tiene complejidad temporal  $\mathcal{O}(n^2)$  y complejidad espacial  $\mathcal{O}(n^2)$ . Asimismo, su velocidad de convergencia es similar a los demás métodos con búsqueda de línea.

Un inconveniente que se puede discernir en los métodos que hacen uso de la búsqueda de línea es que se necesitan varias iteraciones para determinar el tamaño de paso  $\alpha_k$ . En el caso donde el ruido no está apenas presente, esto no supone mayor problema, ya que se calcula rápidamente y además, los puntos donde se analizan las condiciones Wolfe fuertes se hallan en la dirección de búsqueda, por lo que no provoca que el algoritmo sea ineficiente si se pretende aplicarlo a situaciones prácticas reales.

Aunque los métodos de búsqueda de línea inexacta sean altamente eficaces para hallar la fuente cuando el ruido es despreciable, en los casos donde sí está presente, resultan no ser tan eficientes. La principal razón por la que ocurre esto es porque al analizar las condiciones antes mencionadas, cabe la posibilidad de que no se halle un tamaño de paso  $\alpha_k$  ideal y se entre en bucle intentando calcularlo. Aún consiguiéndolo después de muchos intentos, es posible que no sea el valor óptimo y puede provocar errores en la dirección de búsqueda  $p_k$ . Este problema es especialmente relevante en puntos cercanos a la fuente, debido a que en esta zona la diferencia entre el gradiente aproximado y el real es más grande. Por ello, en puntos alejados de ella, se consigue acercar la formación de manera efectiva y no se requiere de muchas iteraciones para hacer la búsqueda de línea inexacta.

En consecuencia, en situaciones donde el ruido esté presente en la medida del campo escalar, es recomendable combinar los métodos con búsqueda de línea inexacta con los métodos robustos propuestos en [5]. Este tipo de combinación ya se ha hecho de modo similar en el método de Newton-Raphson, puesto que se emplea el gradiente ascendente cuando la matriz Hessiana no es

definida negativa o no está bien condicionada. Al inicio del estudio de la localización de la fuente se debería empezar implementando un método que aplique la búsqueda de línea, puesto que se acerca al punto deseado de una manera más eficiente que en el resto de los métodos. A medida que la formación se va acercando a la fuente, se van necesitando más iteraciones para la búsqueda del tamaño de paso ideal. En caso de que se sobrepase un número determinado de iteraciones, se podría implementar el método del gradiente ascendente o de Newton-Raphson, los cuales garantizan la convergencia. De esta manera, se obtendría la eficacia de los métodos de búsqueda de línea con la robustez de los demás algoritmos.

El ruido en el campo escalar afecta mucho más en la convergencia que el error en las posiciones de los robots. Esto es debido a los valores  $v$  y  $\delta$  seleccionados para los cálculos numéricos, que se han tomado de [5], donde se argumenta que son valores realistas para estos parámetros. Cabe destacar que en [5] se ha estudiado por separado el error en la medida de la función y la perturbación de los agentes respecto a sus posiciones determinadas. En este trabajo, sin embargo, se ha decidido simular estos dos factores simultáneamente para que los cálculos numéricos fueran los más similares posibles a una situación en la que nos podríamos encontrar en la vida real.

Hay que tener en cuenta que solo en los métodos con búsqueda de línea se ha hecho la media del gradiente para que el error en las medidas tuviera menos repercusión. Esto significa que se requiere de más capacidad computacional y hace que la convergencia se haya ralentizado. La razón por la que no se ha computado la media del gradiente ni del Hessiano para el algoritmo del gradiente ascendente o de Newton-Raphson es que disminuye considerablemente el valor de la desviación estándar al hacer muchas simulaciones, pero no acelera la convergencia de la media.

Las variables que sí podrían modificar el ritmo de convergencia son el número de robots  $N$  y el radio de la formación  $D$ . Cuando el ruido es despreciable, la modificación de  $N$  no tiene una repercusión notable en la eficiencia del algoritmo. Esto es debido a que las aproximaciones del gradiente y la matriz Hessiana en el centro son satisfactorias con el número de agentes  $N$  que se ha utilizado. Esto se ha mostrado anteriormente en las secciones 4.1. y 4.2. con los valores de los escalares  $L$  y  $M$ . El radio  $D$ , en cambio, sí que tiene una repercusión directa como se ha demostrado en las ecuaciones (5), (7), (13) y (15).

Por otro lado, cuando el error en el campo escalar no es despreciable, resulta que tanto el radio  $D$  como el número de vehículos autónomos  $N$  afectan directamente a la convergencia y las desviaciones estándares en las simulaciones. Para calcular el radio óptimo, en [5] se sugiere igualar el error en las aproximaciones hechas para el gradiente y la matriz Hessiana con las desviaciones estándares calculadas en las ecuaciones (22) y (24). Para el caso bidimensional estudiado en este trabajo, siguiendo el mismo procedimiento hecho en [7], se consigue:

$$D_{\nabla}^* = \left( \frac{2v}{N^{1/2}L} \right)^{1/2} ; \quad D_H^* = \left( \frac{v}{M} \right)^{1/3} \quad (34)$$

donde  $D_{\nabla}^*$  es el radio óptimo para la aproximación del gradiente y  $D_H^*$  para la del Hessiano. Debido a que estas aproximaciones también se usan para los métodos en los que se aplica búsqueda de línea inexacta, estas propiedades se cumplen para todos los algoritmos analizados. Por lo tanto, una manera de mejorar la eficiencia sería utilizando más robots y eligiendo el radio óptimo, aunque hay que tener en cuenta que esto sería más difícil de implementar en la vida real al necesitar más recursos y más capacidad computacional para hacer los cálculos.

Los futuros trabajos podrían enfocarse en implementar una combinación de los métodos aquí descritos para mejorar la eficiencia y la robustez para funciones con medidas ruidosas. Asimismo, se podría estudiar su eficiencia para funciones no convexas y compararlo con algoritmos estocásticos como el que aparece en [4]. Por otra parte, se podría mitigar el impacto de los errores en la medida de la función mediante otras técnicas como el filtro Kalman, el cual probablemente sea más eficiente que hacer la media del gradiente a la hora de atenuar el ruido.

A pesar de que a partir de la sección 4.2. no se han mostrado los resultados para el caso 3D, estos se pueden hallar en [10], donde se han subido todos los algoritmos propuestos en este trabajo para 2D y 3D.

## 6. Bibliografía

### Referencias

- [1] Hajieghrary, H., Mox, D., & Hsieh, M. A. (2017). Information theoretic source seeking strategies for multiagent plume tracking in turbulent fields. *Journal of Marine Science and Engineering*, 5(1), 3.
- [2] L. Marques, U. Nunes, and A. de Almeida, “Particle swarm-based olfactory guided search,” *Auton. Robots*, vol. 20, pp. 277–287, 2006
- [3] Chen, X. X., & Huang, J. (2019). Odor source localization algorithms on mobile robots: A review and future outlook. *Robotics and Autonomous Systems*, 112, 123-136.
- [4] Atanasov, N. A., Ny, J. L., & Pappas, G. J. (2014). Distributed algorithms for stochastic source seeking with mobile robot networks: Technical report. arXiv preprint arXiv:1402.0051.
- [5] Lara Briñon Arranz, Alessandro Renzaglia, Luca Schenato. Multi-Robot Symmetric Formations for Gradient and Hessian Estimation with Application to Source Seeking. *IEEE Transactions on Robotics*, IEEE, 2019, 35 (3), pp.782-789. [ff10.1109/TRO.2019.2895509](https://doi.org/10.1109/TRO.2019.2895509). [ffhal-01991153f](https://hal.inria.fr/hal-01991153f)
- [6] Brinón-Arranz, L., Seuret, A., & Canudas-de-Wit, C. (2014). Cooperative control design for time-varying formations of multi-agent systems. *IEEE Transactions on Automatic Control*, 59(8), 2283-2288.
- [7] L. Briñon-Arranz, A. Renzaglia, and L. Schenato, “Technical report on: Multi-robot symmetric formations for gradient and Hessian estimation with application to source seeking,” [https://hal.inria.fr/hal-01989573/ document](https://hal.inria.fr/hal-01989573/document)
- [8] Wright, S., & Nocedal, J. (1999). Numerical optimization. Springer Science, 35(67-68), 7.
- [9] Xu, C., Chen, Y., Duan, S., Wu, H., & Qi, Y. (2021). Cooperative Source Seeking in Scalar Field: A Virtual Structure-Based Spatial-Temporal Method. In *Collaborative Computing: Networking, Applications and Worksharing: 16th EAI International Conference, CollaborateCom 2020, Shanghai, China, October 16–18, 2020, Proceedings, Part II* 16 (pp. 320-336). Springer International Publishing.
- [10] Blázquez, X. (2023). Algoritmos multiagente. GitHub Repository. <https://github.com/XabiBlaz/algoritmos-multiagente>