

## Modelo basado en agentes para exploración y optimización de trayectorias mediante cómputo bioinspirado

Marco A. Jiménez-Limas<sup>1,2,3</sup>, Emmanuel Salcedo-Carrillo<sup>4</sup>,  
Fernanda Borjas<sup>2</sup>, Amaury Salcedo-Carrillo<sup>5</sup>

<sup>1</sup> Instituto Politécnico Nacional,  
Centro de Investigación en Computación,  
México

<sup>2</sup> Universidad Nacional Autónoma de México,  
Facultad de Ciencias,  
México

<sup>3</sup> Universidad Autónoma de la Ciudad de México,  
Posgrado en Ciencias de la Complejidad,  
México

<sup>4</sup> Instituto Politécnico Nacional,  
Escuela Nacional de Ciencias Biológicas,  
México

<sup>5</sup> Instituto Politécnico Nacional  
Centro de Investigación y de Estudios Avanzados,  
Departamento de Fisiología, Biofísica y Neurociencias,  
México

{marcojimenez, fernandaborjas}@ciencias.unam.mx,  
escsal97@gmail.com, amaury.salcedo@cinvestav.mx

**Resumen.** El comportamiento colectivo durante el forrajeo es un fenómeno ubicuo en la naturaleza; diversos enfoques se han propuesto para modelar este tipo de comportamiento, dentro de los que destacan los modelos basados en agentes (MBA). En el presente trabajo se realizó un modelo de exploración de un espacio desconocido de dos dimensiones utilizando tres tipos de caminatas aleatorias (LF, URW, GRW) para evaluar su eficacia de búsqueda. Posteriormente, se realizó un MBA en tres dimensiones en el que el espacio a explorar consistió en tres componentes principales: punto de partida (*home*), punto de llegada (*target*) y obstáculos. A los agentes se les implementaron reglas de colectividad (separación, alineación y cohesión) y una caminata aleatoria tipo LF para explorar el espacio y así llegar al *target*. Se realizaron 30 simulaciones, que se detenían una vez que un agente alcanzaba el *target*, para posteriormente generar un grafo utilizando la información recabada por cada agente (la trayectoria recorrida por agente y la posible comunicación local). Se aplicó el algoritmo ACO sobre el grafo obtenido para encontrar la ruta más corta al objetivo, y se comparó la ruta optimizada con la realizada por el agente que alcanzó el *target* para evaluar si había diferencias significativas entre sus distancias. Las distancias optimizadas presentaron una diferencia importante al compararlas contra las originales y se logró una disminución del 22 % en la distancia recorrida del *home* al *target* utilizando el

método planteado. El uso de MBA y caminatas aleatorias, en especial los LF, resultan una forma adecuada de describir la dinámica del forrajeo en animales, ya que toman en cuenta la interacción entre los componentes del sistema y con el ambiente, dando lugar a comportamientos emergentes que no pueden ser descritos a partir únicamente de conductas individuales.

**Palabras clave:** Inteligencia colectiva, caminata aleatoria, forrajeo.

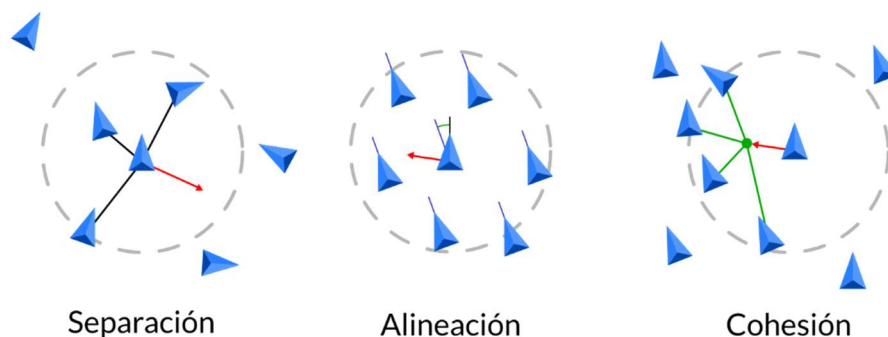
## Agent-Based Model for Exploration and Path Optimization Using Bio-inspired Computing

**Abstract.** Collective behavior during foraging is a ubiquitous phenomenon in nature; several approaches have been developed to model this type of behavior, within which stand out Agent-Based Models (ABM). In the present work, a two-dimensional exploration model of an unknown space was carried out using three types of random walks (LF, URW, GRW) to evaluate its search efficiency. Later, a three-dimensional ABM was developed, in which the space to explore has three main components: home, target and obstacles. The agents were implemented collective rules (separation, alignment, and cohesion) and a LF random walk to explore the space and thus reach the target. 30 simulations were carried out, which stopped once an agent reached the target, to later generate a graph using the information collected by each agent. The ACO algorithm was applied on the obtained graph to find the shortest path to the target, and the optimized route was compared with the route taken by the agent who reached the target to evaluate if there were significant differences between their distances. The optimized paths presented an important difference when compared against the originals, and a 22% decrease in the distance traveled from home to the target was achieved using the proposed method. The use of ABM and random walks, especially the LF, are an adequate way to describe the dynamics of foraging in animals, since they consider the interaction between the components of the system and with the environment, giving rise to emergent behaviors that cannot be described solely based on individual behaviors.

**Keywords:** Collective intelligence, random walks, foraging.

### 1. Introducción

La presencia de comportamiento colectivo en ciertos grupos de animales es un fenómeno ubicuo en la naturaleza. Se han brindado diversas explicaciones a este tipo de comportamiento, dentro de las que destacan la evitación de depredadores y el forrajeo [1]. Este tipo de agrupaciones pueden ser vistos como sistemas colectivos, los cuales son sistemas complejos de agentes que interactúan entre sí y en los que cada agente tiene una tarea propia a realizar [2].



**Fig. 1.** Reglas de colectividad: Separación, Alineación y Cohesión. Dado un agente con un rango de visión, su vecindad se define como todos aquellos agentes que se encuentran a su alrededor dentro del radio dado por el rango de visión.

Algunos ejemplos de estos sistemas son las colonias de hormigas, termitas o las colmenas de abejas, las cuales están constituidas de muchos individuos con una función específica [3]. Otros sistemas que presentan comportamiento colectivo sin una división de la labor son los bancos de peces o las parvadas de pájaros.

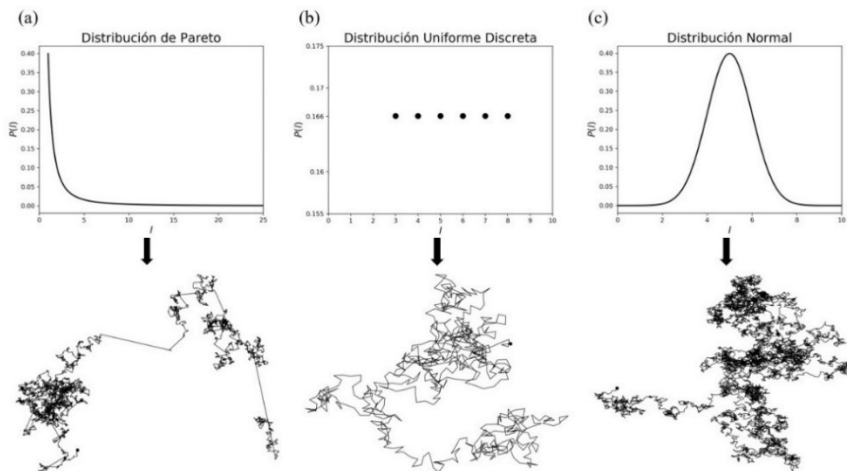
El estudio del comportamiento colectivo en sistemas de animales es ampliamente utilizado dentro del ámbito del cómputo bioinspirado, especialmente para el desarrollo de algoritmos de exploración y de optimización. Uno de los primeros acercamientos al modelado del comportamiento colectivo de animales fue la simulación *Boids*, desarrollada por Craig Reynolds, la cual tuvo como objetivo la replicación del comportamiento de las parvadas de aves o de los bancos de peces utilizando geometría computacional 3D [4].

Este modelo implementa tres reglas simples a nivel local que dan lugar a la colectividad: i) *separación*: cada agente debe mantener una distancia entre sí para evitar la sobre agrupación; ii) *alineación*: cada agente ajusta su posición para que corresponda con el alineamiento promedio del resto de los agentes; iii) *cohesión*: cada agente intenta moverse a la posición promedio de otros agentes. Estas reglas describen como cada agente reacciona respecto a la ubicación y velocidades del resto de agentes cercanos en el Flock (Figura 1).

Cada agente es independiente, ya que posee una velocidad, posición y orientación propia. Sin embargo, estas propiedades se ven alteradas por las reglas de colectividad, cuando hay otros agentes en su vecindad (la cual se caracteriza por una distancia medida desde el centro del agente y un ángulo medido desde la dirección de vuelo del agente). Esto ocasiona que cualquier agente que se localice fuera de esa región sea ignorado (Figura 1).

En cuanto a los algoritmos bioinspirados de exploración, un método ampliamente utilizado es el uso de caminatas aleatorias (RW) [5]. Dentro de ellas podemos encontrar caminatas aleatorias uniformes, gaussianas o con difusión anómala (como los vuelos de Lévy). Las RW son procesos estocásticos que describen el movimiento de agentes o partículas en un espacio, en el que los agentes se mueven a una distancia determinada por una distribución de probabilidad y un ángulo aleatorio.

Las caminatas aleatorias uniformes (URW por sus siglas en inglés) tienen un paso que viene dado por una distribución uniforme; en las gaussianas (GRW por sus siglas en inglés), la longitud del paso está dada por una distribución normal; en los vuelos de Lévy



**Fig. 2.** (a) Distribución de Pareto que da lugar al vuelo de Lévy; (b) Distribución uniforme discreta que da lugar a una caminata aleatoria uniforme; (c) Distribución normal que da lugar a una caminata aleatoria gaussiana.

(LF por sus siglas en inglés), la longitud del paso está dada por una distribución libre de escala (anómala), como puede ser una distribución de Pareto (Figura 2).

Existen también diversos algoritmos bioinspirados para optimizar (maximizar o minimizar) una función, dentro de los cuales están los basados en hormigas (ACO: Ant Colony Optimization) o abejas (ABC: Artificial Bee Colony) [6]. Dentro del campo de la inteligencia artificial, un paradigma bioinspirado es la Inteligencia de enjambre (Swarm intelligence), la cual es definida como “la inteligencia colectiva emergente de los grupos de agentes simples” [7, 8, 9]. El ACO es una metaheurística que engloba una serie de técnicas de optimización inspiradas en el comportamiento colectivo de las hormigas durante el forrajeo, mediante la cual es posible dar solución a problemas computacionales, como la búsqueda de la trayectoria más corta entre dos nodos en un grafo [10].

La modelación basada en agentes es una herramienta ampliamente utilizada para estudiar comportamientos colectivos de distintos grupos de animales. Dentro de algunos trabajos que se relacionan con estos tópicos destaca el realizado por Sellers y colaboradores [11] se utilizó un MBA para modelar y simular el forrajeo colectivo de babuinos, con el cual obtuvieron resultados que son concordantes con el comportamiento etológico de este grupo de animales.

Por otro lado, Robinson y colaboradores [12] hicieron un MBA para entender el rol de las feromonas en un grupo social de hormigas durante el forrajeo, y encontraron que diferentes tipos de feromonas tienen efectos determinantes en la dinámica del forrajeo colectivo. Otros enfoques toman en cuenta el ambiente en el que se desenvuelven estos grupos de animales, como el estudio desarrollado por Rivière y colaboradores [13], quienes desarrollaron un MBA para modelar y simular el forrajeo colectivo de abejas utilizando datos del ambiente como el tiempo local y las fuentes de néctar cercanas; sus resultados mostraron la formación de procesos autoorganizados de selección de las mejores fuentes de néctar disponibles por parte de los agentes.

En este trabajo se presenta un modelo basado en agentes (MBA) capaz de simular en 3D la exploración y optimización de trayectorias utilizando como base el comportamiento de sistemas colectivos. Para la exploración, se utilizaron caminatas aleatorias combinadas con el modelo *Boids*, variando el tipo de caminata aleatoria para encontrar aquella que realiza una búsqueda más eficiente. Para la optimización de la trayectoria se utilizó el algoritmo ACO.

## 2. Métodos

El trabajo se llevó a cabo en tres partes, en la primera se exploraron tres tipos de caminatas aleatorias (uniformes, gaussiana y vuelos de Lévy) y se examinó cuál de las tres fue la más eficiente en cuanto al tiempo de convergencia (número de iteraciones en el que los agentes alcanzaron el objetivo; es decir, a la fuente de alimento). En la segunda parte se realizó un modelo basado en agentes, en el cual se implementó el modelo *Boids* [4] para simular el forrajeo colectivo en un espacio tridimensional utilizando los tres tipos de caminatas aleatorias. De esta simulación se obtuvo un grafo cuyos nodos fueron las posiciones que tuvieron todos los agentes durante el forrajeo. Finalmente, se aplicó el ACO en dicho grafo para encontrar la ruta más corta del origen al objetivo.

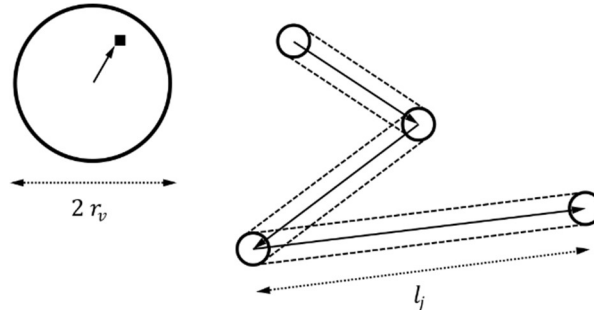
### 2.1. Forrajeo mediante caminatas aleatorias

La parte de la exploración del espacio durante el forrajeo se modeló usando caminantes aleatorios, ya que este tipo de movimiento es el que rige el forrajeo de varios animales en presencia de incertidumbre en el ambiente [14, 15, 16].

Para esta parte, se probó cada uno de los tres tipos de caminata aleatoria utilizando un solo agente, al cual se le implementaron las condiciones correspondientes a cada caminata aleatoria: para la uniforme, se utilizó una distribución uniforme discreta que tomaba los valores 4, 5 y 6 con la misma probabilidad; para la gaussiana, se utilizó una distribución normal con una media de 5 y desviación estándar de 1; y para los vuelos de Lévy, se utilizó una distribución de Pareto con  $\mu = 2$ , paso mínimo de 1 y paso máximo de 25. El agente contaba con un rango de visión  $r_v$  y se desplazaba en un espacio bidimensional. En el espacio se colocaron dos puntos importantes: el punto de partida (*home*) y el objetivo (*target*). El *target* representaba la fuente de alimento que el agente estaba buscando.

El modelo de forrajeo usado fue el descrito en [17]; dado un espacio D-dimensional, un agente con un rango de visión  $r_v$  y un *target* (Figura 3):

1. Si hay un *target* ubicado dentro del rango de visión, entonces el agente se mueve en una línea recta hacia él. Si hay más de un *target*, se mueve hacia el más cercano.
2. Si no hay un *target* dentro del radio de visión, entonces en su  $j$ -ésimo paso el agente elige una dirección al azar y una distancia  $l_j$  dependiendo del tipo de caminata aleatoria que tenga. Luego se mueve incrementalmente al nuevo punto, buscando continuamente un *target* dentro de su radio de visión durante su desplazamiento. Si no detecta un *target*, se detiene después de recorrer la distancia  $l_j$  y elige una nueva dirección y una nueva distancia  $l_{j+1}$ . De otra manera se dirige al *target* como en el paso 1.



**Fig. 3.** Modelo de forrajeo [17]: si un *target* (cuadrado obscuro) se encuentra dentro del rango de visión, entonces el agente se mueve directamente hacia el objetivo; de otra forma, el agente escoge una dirección aleatoria y una longitud de paso de acuerdo al tipo de caminata aleatoria que posee y se desplaza buscando continuamente el *target* dentro de su rango de visión.

Se generaron diferentes escenarios para probar la eficiencia de búsqueda de los agentes utilizando los tres tipos de caminatas aleatorias. Para cada una de ellas, se implementaron tres escenarios que variaban en cuanto a la distancia entre el *home*, en la posición (0,0), y el *target*, en las posiciones (10,10): correspondiente a la distancia corta; (-50,0): distancia media; y (90,-90): distancia larga.

El espacio en el que se generaron los escenarios fue un cuadrado centrado en el punto (0,0) cuyos lados tenían longitud de 200 (su rango iba desde -100 hasta 100). Para cada uno de los tratamientos (combinación tipo de caminata y posición del *target*), se colocó a un agente en el espacio y se fijó un límite de 1000 iteraciones para llegar al *target* (el experimento se paraba si el agente no lo alcanzaba en este tiempo, ya que se consideró que este límite era suficiente para evaluar la eficiencia de la caminata)

El valor del rango de visión  $r_v$  utilizado fue de 1. Cada uno de los tratamientos se repitió 100 veces y se obtuvo un promedio del número de iteraciones que le tomaba al agente alcanzar el *target*. A partir de los resultados de estas exploraciones, se eligió el tipo de caminata aleatoria más eficiente para la optimización de trayectorias que se describen más adelante.

## 2.2. Modelo basado en agentes para forrajeo colectivo

Dado que el objetivo del presente trabajo fue modelar el forrajeo colectivo presente en sistemas de animales de la manera más realista posible, se implementó un modelo basado en agentes en tres dimensiones utilizando el lenguaje de programación NetLogo [18], el cual es un software para modelado en 2D y 3D multi-agente, que permite generar código de fácil manejo que puede asociarse con una interfaz gráfica para variar los parámetros programados de manera dinámica durante la simulación.

Se utilizó como base el modelo realizado por [19]. Este modelo utiliza las reglas de colectividad descritas anteriormente (Figura 1a) como base para el comportamiento de los agentes. Este modelo fue modificado para que los agentes pudieran tener distintos tipos de caminatas aleatorias. Además, se añadieron obstáculos al espacio, y a los agentes se les implementó la capacidad de evadir obstáculos y de identificar *targets* dentro de su rango de visión.

Se generó un espacio en tres dimensiones y una interfaz gráfica (GUI por sus siglas en inglés), el espacio consistía en un universo cúbico de 64,000 *patches* o parcelas, con limitación de localización, es decir, con contención en el movimiento de los agentes dentro del espacio.

Se agregaron 40 obstáculos en el espacio con posiciones aleatorias (con la posibilidad de aumentar el número de éstos), además se colocó la posición de *home* y de *target*. A la GUI se le agregaron *sliders* y *switches*, que permitían variar los valores de los parámetros para controlar el tipo de comportamiento deseado en la simulación (tipo de caminata aleatoria, rango de visión, de separación, de alineación y de cohesión).

Para la simulación, se utilizaron 500 agentes, las coordenadas del *home* (0, 0, 0), las coordenadas del *target* (10, -10, 10) y tipo de caminata LF. Con estos parámetros establecidos, se corrió la simulación hasta el momento en el que uno de los agentes tenía en su rango de visión al objetivo. Las posiciones de cada agente se guardaron en un archivo durante cada iteración, y una vez concluida la simulación se utilizó el archivo de posiciones para generar un grafo, cuyos nodos correspondían a las posiciones que tuvieron los agentes a lo largo de la simulación.

Posteriormente, se calcularon las distancias entre nodos y se establecieron sus conexiones, para finalmente calcular la ruta más corta del *home* al *target* en el grafo utilizando el ACO.

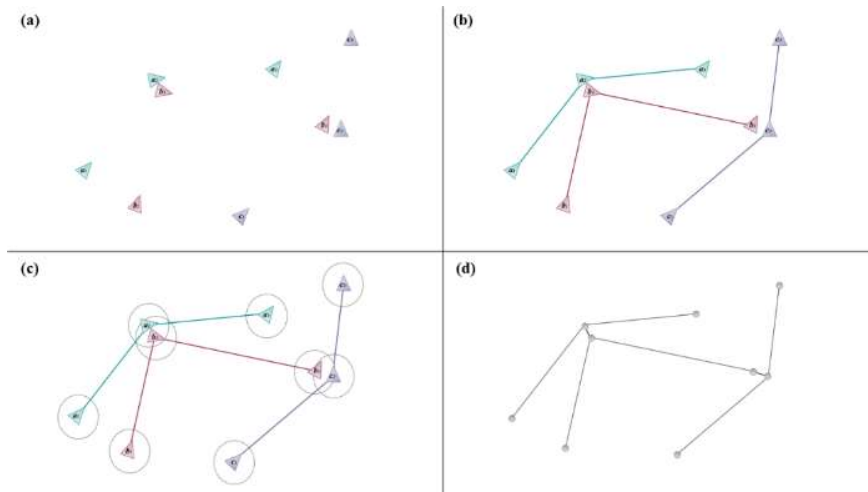
### 2.3. Optimización de trayectorias

Como se mencionó anteriormente, para la optimización se utilizó el algoritmo ACO, el cual consiste en optimizar utilizando como base el comportamiento de forrajeo de las hormigas. Éstas no conocen el territorio al inicio del forrajeo, por lo que empiezan a buscar el alimento de manera aleatoria. Una vez que encuentran la fuente de alimento, regresan a su hogar dejando un rastro de feromona. En esta búsqueda, diferentes hormigas pueden encontrar diferentes rutas a la fuente de alimento, sin embargo, permanece el problema de cuál de todas es la más cercana a su nido.

Esto se resuelve con la concentración de feromonas en los rastros, ya que el camino más corto es aquel en el que las feromonas tardarán más en evaporarse. Al tener más concentración de feromonas, más hormigas van siguiendo este rastro aumentando así su concentración de feromona. De manera estocástica, algunas hormigas salen de esta trayectoria y generan una nueva ruta; de ser una ruta más corta que la original, la concentración de feromona de la ruta tardará menos en evaporarse ocasionando que las hormigas opten por tomar esta ruta.

Este proceso se repite hasta que las hormigas logran encontrar la ruta más corta en su exploración del espacio [10]. El ACO aplica estas ideas sobre un grafo, en el cual las hormigas van recorriendo los nodos para encontrar la ruta más corta entre dos puntos. El algoritmo inicia con una cantidad  $n$  de hormigas exploradoras definidas por el usuario y con una matriz de concentración de feromonas, donde la concentración inicial también está definida por el usuario.

Para aplicar el ACO, una vez concluida la simulación se obtuvo un archivo con las posiciones espaciales de cada agente en cada iteración. El objetivo del presente trabajo fue encontrar la ruta más corta que un agente puede tomar para llegar del *home* al *target*; dado que se desconoce el espacio, solo es posible obtener información de los puntos por los que pasaron los agentes durante la simulación, y la optimización de la ruta más corta



**Fig. 4.** Proceso de generación del grafo. Se toma el caso de tres agentes  $a$ ,  $b$  y  $c$  en tres iteraciones. (a) Se identifican como nodos las posiciones que cada agente tuvo en cada iteración. (b) Se unen los nodos que corresponden a las posiciones de cada agente en las tres iteraciones; así se genera un subgrafo por cada agente correspondiente a su trayectoria recorrida. (c) Para cada nodo se hace una búsqueda de radio  $r$  a su alrededor para detectar la presencia de nodos cercanos y de ser el caso, se genera una conexión a dichos nodos. (d) El grafo resultante contiene los subgrafos formados por las trayectorias de cada agente y las uniones entre nodos que cuyas posiciones fueron cercanas.

solo puede hacerse sobre dichos puntos conocidos. Para ello, fue necesario convertir las posiciones que fueron teniendo los agentes en un grafo cuyos nodos representan dichas posiciones. Estos nodos fueron conectados mediante aristas utilizando los siguientes criterios.

1. Sea  $A = \{a_1, a_2, \dots, a_n\}$  el conjunto de todos los agentes, donde  $a_i$  representa al agente  $i$ . Sea  $T = \{t_1, t_2, \dots, t_{m-1}\}$ , donde  $t_j$  representa la  $j$ -ésima iteración; en la simulación se consideraron  $m$  iteraciones. Un nodo representa la posición de  $a_i$  al tiempo  $t_j$ . Se conectó el nodo correspondiente a  $a_i$  en el tiempo  $t_j$  con el nodo correspondiente a  $a_i$  en el tiempo  $t_{j+1}$ , para todo agente  $a$  en  $A$  y para todo  $t$  en  $T$  (Figura 4a). Esto permitió generar una trayectoria por cada agente que conectaba a todas las posiciones que tuvo ese agente durante la simulación (Figura 4b).
2. Del paso anterior se obtuvo un grafo disconexo que contenía  $n$  subgrafos correspondientes a las rutas que siguieron los  $n$  agentes; sin embargo, para obtener una ruta más corta es necesario tener un grafo más denso, que utilice la información recabada por todos los agentes. Se procedió a conectar los nodos entre ellos de acuerdo con la distancia que los separaba, para lo cual se seleccionó un parámetro  $r$  el cual corresponde a un radio; para cada nodo  $a_i$ , se obtuvieron los nodos que estaban a una distancia menor o igual a  $r$  respecto a su posición (Figura 4c). Dichos nodos se conectaron a  $a_i$ , lo que permitió generar un grafo más denso sobre el cual se llevó a cabo la optimización (Figura 4d).

A partir de la matriz de adyacencias correspondiente al grafo generado se calcularon las distancias entre los nodos conectados para obtener una matriz de distancias. Dicha matriz es la entrada que recibe el algoritmo ACO, por lo que con ella fue posible aplicar la optimización para encontrar la ruta más corta. Los valores de los parámetros utilizados para el ACO fueron los siguientes:  $Q = \alpha = \beta = 1$ ,  $\rho$  (coeficiente de evaporación) = 0.01 y la matriz de feromonas fue inicializada con un valor de 0.1.

Se utilizaron 10 hormigas para la optimización y 1000 iteraciones. El proceso descrito en las secciones 2.2 y 2.3 se repitió 30 veces para evaluar si en efecto el modelo estaba encontrando una ruta más corta que la del agente que llegó al *target*.

### 3. Resultados

#### 3.1. Forrajeo mediante caminatas aleatorias

A partir de la exploración realizada utilizando diferentes tipos de caminatas aleatorias (LF, URW y GRW) y con diferentes distancias del *home* al *target*, se observó que, para distancias cortas, la URW fue el tipo de caminata más eficiente, con un promedio de iteraciones de 20 para alcanzar el *target* (Figura 5a). Para distancias medias y largas, el LF resultó ser el tipo de caminata más eficiente, con un promedio de iteraciones para alcanzar el *target* de 153 y 378 respectivamente (Figura 5b, c).

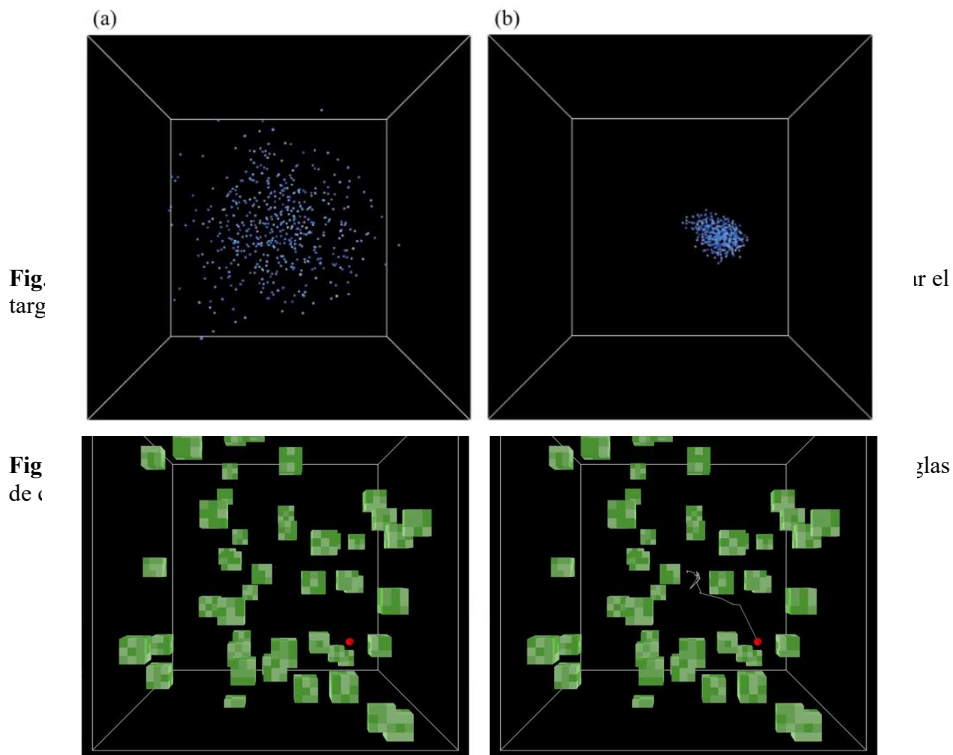
En los tres escenarios, se pudo observar que la GRW fue la que tuvo un peor desempeño en cuanto a las iteraciones que le tomó llegar al *target*; de hecho, se encontró que para distancias largas este tipo de caminata no logró llegar al objetivo en el límite de iteraciones impuesto (1000 iteraciones; Figura 5c).

#### 3.2. Modelo basado en agentes para forrajeo colectivo

En la Figura 6 se observa el ambiente en el que se desarrolló la simulación del MBA. Los cubos verdes representan los obstáculos añadidos al entorno para brindarle al sistema un poco más de realismo; el punto de partida (*home*) se encuentra en las coordenadas (0,0,0), es decir, en el centro del espacio: el punto rojo representa el objetivo que los agentes debían alcanzar (Figura 6a). Una vez realizada la exploración (que concluía cuando el primer agente llegaba al *target*), se optimizaba la ruta del *home* al *target* siguiendo los pasos descritos en la sección 2.3 y dicha ruta se visualizaba gráficamente como se muestra en la Figura 6b.

En la Figura 7 se observan las comunidades de agentes en un ambiente sin obstáculos y sin *target* para observar la dinámica colectiva. En ausencia de reglas de colectividad, se observa una comunidad dispersa que abarca un mayor volumen de exploración (Figura 7a); mientras que al aplicar las reglas de colectividad puede observarse la formación de un cumulo cuyo comportamiento asemeja a la forma en que algunos sistemas de animales llevan a cabo el forrajeo (Figura 7b). Se decidió mantener las reglas de colectividad debido a que le brindan realismo al modelo, ya que aparentan la forma en que este proceso de forrajeo es llevado a cabo en sistemas naturales.

A partir del modelo basado en agentes implementado en NetLogo, se observó que las comunidades de agentes creadas tuvieron un comportamiento colectivo similar independientemente del tipo de caminata aleatoria elegida. Sin embargo, esto no



**Fig. 6.** (a) Ambiente en el que se desarrolló la simulación del MBA. (b) Representación de la trayectoria optimizada en el ambiente utilizando el ACO.

necesariamente implica que el tipo de caminata aleatoria implementada en los agentes no influya en el comportamiento a nivel individual y colectivo para llevar a cabo el forrajeo.

En el caso de que un agente se separe del resto, el tipo de caminata que este posee puede ser determinante en su eficacia de exploración del espacio.

Asimismo, se observó que, al implementar las reglas de colectividad desde el inicio, los agentes tendían a explorar el espacio en conjunto en lugar de dispersarse y abarcar más área de exploración.

Por ende, si estas reglas se mantenían durante toda la simulación, el grafo resultante hubiera abarcado menos espacio ya que todas las trayectorias hubieran estado concentradas, por lo que se decidió únicamente mantener la regla de separación durante todo el transcurso de la simulación y aplicar las reglas de cohesión y alineamiento una vez pasadas las primeras diez iteraciones. De esta forma se generó una mayor cantidad de cúmulos de agentes que aumentaron el volumen de exploración.

### 3.3. Optimización de trayectorias

A partir de las 30 repeticiones de la simulación y posterior optimización descritas en las secciones 2.2 y 2.3, se obtuvo la distancia recorrida por el primer agente que logró

**Tabla 1.** Resultados de las repeticiones de las rutas originales y optimizadas.

Ruta	Promedio de distancias	Desviación estándar	Error estándar
Original	62.24	25.11	4.58
Optimizada	48.75	20.17	3.68

alcanzar el *target* en cada simulación (en adelante “ruta original”) y la distancia optimizada mediante el algoritmo ACO (en adelante “ruta optimizada”). El promedio de la distancia de las rutas optimizadas para las 30 repeticiones fue considerablemente menor al promedio de distancias de las rutas originales, tuvo una menor desviación estándar y un menor error estándar (Tabla 1).

Para comparar las distancias obtenidas de las rutas originales contra las de las rutas optimizadas, se realizó una prueba estadística de *t*. El resultado de la prueba indicó que las medias de las muestras difieren significativamente ( $p < 0.01$ ). Estos resultados indican que las distancias optimizadas presentan una diferencia importante al compararlas contra las originales, y que se logró una disminución aproximada del 22 % en la distancia recorrida del *home* al *target* al optimizar la ruta utilizando el método planteado.

#### 4. Discusión

Los resultados obtenidos de la exploración utilizando diferentes tipos de caminatas aleatorias permitieron identificar que el tipo de caminata más eficiente para la búsqueda de objetivos en un ambiente desconocido cuando estos se hallan en distancias cortas respecto al punto de partida es la URW. Mientras que los vuelos de Lévy representan el tipo de caminata más efectiva cuando las distancias entre el *home* y el *target* son más grandes. Esto puede deberse al hecho de que la caminata tipo LF está regida por una distribución de Pareto, la cual es una distribución anómala, es decir, una distribución libre de escala (sin escala característica).

Esto ocasiona que los pasos dados por los caminantes no tienen una longitud característica, permitiéndoles dar pasos muy variables en cuanto a su longitud. Al observar la dinámica seguida por los caminantes regidos por una caminata tipo LF (Figura 2a), puede notarse que la exploración realizada por los agentes consta de pasos regulares en un intervalo de tiempo y pasos de longitud muy larga que les permite salir completamente del área de exploración en la que se encontraban.

Este comportamiento permite a los agentes llegar a más zonas durante su exploración (pasos largos y súbitos) y en cada una de ellas realizar una búsqueda fina (pasos cortos y regulares). Este tipo de caminata permite equilibrar el área de exploración y la precisión de la búsqueda.

Precisamente por esta razón, para el MBA en tres dimensiones se decidió utilizar los vuelos de Lévy, pese a que el comportamiento colectivo de los agentes era similar independientemente del tipo de RW aplicado. Esto se hizo así ya que, si algún agente se llegaba a separar del grupo, este tipo de caminata resulta más eficiente para que pueda hacer una exploración del espacio efectiva [17].

Los resultados obtenidos de la optimización mostraron que la distancia recorrida del *home* al *target* disminuyó de manera significativa al optimizar la ruta con el método planteado. Este resultado es importante porque toma en cuenta las interacciones entre los agentes que pueden darse en una exploración, ya que la conexión de nodos cercanos entre sí al momento de generar el grafo permite tomar en cuenta la información recabada colectivamente, pero a nivel local, semejando comportamientos similares presentados en algunos entes en la naturaleza, como las bacterias que forman los *biofilms* o la comunicación en colonias de insectos sociales [20, 21].

## 5. Conclusiones y trabajo a futuro

El uso de modelos basados en agentes y de caminatas aleatorias, en especial aquellas con distribuciones libres de escala (como son los vuelos de Lévy), resultan una forma adecuada de describir la dinámica que presentan algunos grupos sociales de animales durante el forrajeo.

Esto debido a que toman en cuenta la interacción entre los componentes del sistema y con el ambiente, dando lugar a comportamientos emergentes que no pueden ser descritos a partir únicamente de conductas individuales. Asimismo, el uso del ACO permite hacer una optimización eficiente en cuanto a tiempo de cómputo, dado que los grafos obtenidos pueden ser muy grandes y representar un problema al utilizar otro tipo de algoritmos de ruta más corta.

En futuros trabajos, puede explorarse el comportamiento del sistema en ausencia de reglas de colectividad y evaluar si la reducción de la distancia al llevar a cabo la optimización se mantiene bajo este escenario. En caso de que esto suceda, sería interesante determinar qué parámetros o configuraciones del ambiente ocasionan que los sistemas naturales presenten este tipo de comportamientos colectivos. Queda por esclarecer la forma en que los agentes o individuos interactúan con el ambiente y como este influye en la manera en que estos se organizan.

## Referencias

1. Boyer, D., Miramontes, O., Ramos-Fernández, G., Mateos, J. L., Cocho, G.: Modeling the searching behavior of social monkeys. *Physica A: Statistical Mechanics and its Applications*, vol. 342, no. 1-2, pp. 329–335 (2004) doi: 10.1016/j.physa.2004.04.091
2. Namatame, A.: *Adaptation and evolution in collective systems*. World Scientific (2005)
3. Gotwald, H.: *Army ants the biology of social predators*. Cornell University Press, Ithaca, NY (1995)
4. Reynolds, C.: Flocks, herds, and schools: A distributed behavioral model. SIGGRAPH '87: In: *Proceedings of the 14th Annual Conference on Computer Graphics and Interactive Techniques*, vol. 21, no. 4, pp. 25–34 (1987) doi: 10.1145/37401.37406
5. Xia, F., Liu, J., Nie, H., Fu, Y., Wan, L., Kong, X.: Random walks: A review of algorithms and applications. *IEEE Transactions on Emerging Topics in Computational Intelligence*, vol. 4, no. 2, pp. 95–107 (2019) doi: 10.1109/TETCI.2019.2952908
6. Duan, H., Li, P.: *Bio-inspired computation in unmanned aerial vehicles*. Springer Berlin Heidelberg (2014)

7. Bonabeau, E.: Agent-based modeling: Methods and techniques for simulating human systems. In: Proceedings of The National Academy of Sciences, vol. 99, no. 3, pp. 7280–7287 (2002) doi: 10.1073/pnas.082080899
8. Russell, S., Norving, P.: Artificial intelligence a modern approach (2003)
9. Kennedy, J., Eberhart, R.: Swarm Intelligence. Elsevier (2001)
10. Dorigo, M., Di Caro, G.: Ant colony optimization: A new meta-heuristic. In: Proceedings of the 1999 congress on evolutionary computation-CEC99, vol. 2, pp. 1470–1477 (1999)
11. Sellers, W. I., Hill, R., Logan, B.: Biorealistic simulating baboon foraging using agent-based modeling (2007)
12. Robinson, E. J., Ratnieks, F. L., Holcombe, M.: An agent-based model to investigate the roles of attractive and repellent pheromones in ant decision making during foraging. *Journal of theoretical Biology*, vol. 255, no. 2, pp. 250–258 (2008)
13. Rivière, J., Alaux, C., Le Conte, Y., Layec, Y., Lozac’h, A., Rodin, V., Singhoff, F.: Toward a complete agent-based model of a honeybee colony. *Highlights of Practical Applications of Agents, Multi-Agent Systems, and Complexity: The PAAMS Collection, PAAMS 2018, Communications in Computer and Information, Springer, Cham*, vol. 887, pp. 493–505 (2018) doi: 10.1007/978-3-319-94779-2\_42
14. Abbott, J. T., Austerweil, J. L., Griffiths, T. L.: Random walks on semantic networks can resemble optimal foraging. *Neural Information Processing Systems Conference*, vol. 122, no. 3, pp. 558 (2015)
15. James, A., Plank, M. J., Edwards, A. M.: Assessing Lévy walks as models of animal foraging. *Journal of the Royal Society Interface*, vol. 8, no. 62, pp. 1233–1247 (2011)
16. Fronhofer, E. A., Hovestadt, T., Poethke, H. J.: From random walks to informed movement. *Oikos*, vol. 122, no. 6, pp. 857–866 (2013)
17. Viswanathan, G. M., Raposo, E. P., Da Luz, M. G. E.: Lévy flights and superdiffusion in the context of biological encounters and random searches. *Physics of Life Reviews*, vol. 5, no. 3, pp. 133–150 (2008) doi: 10.1016/j.plrev.2008.03.002
18. Wilensky, U.: NetLogo flocking model. Center for Connected Learning and Computer-Based Modeling, Northwestern University, Evanston, IL. <http://ccl.northwestern.edu/netlogo/models/Flocking> (1998)
19. Wilensky, U.: NetLogo. Center for connected learning and computer-based modeling. Northwestern University, Evanston, IL. <http://ccl.northwestern.edu/netlogo/> (1999)
20. Jacob, E. B., Becker, I., Shapira, Y., Levine, H.: Bacterial linguistic communication and social intelligence. *TRENDS in Microbiology*, vol. 12, no. 8, pp. 366–372 (2004)
21. Ben-Jacob, E.: Social behavior of bacteria: from physics to complex organization. *The European Physical Journal B*, vol. 65, no. 3, pp. 315–322 (2008)

