

Algoritmo de navegación de móvil robótico de trabajo en sociedad para mapeo de escenarios

Jesús Yaljá Montiel Pérez, Rodolfo Romero Herrera, Félix Serrano Talamantes

Escuela Superior de Cómputo, Instituto Politécnico Nacional
Unidad Profesional Adolfo López Mateos, Av. IPN S/N Col. Lindavista 07738
México, D. F.; becarios COFAA y EDD del IPN; yalja@ipn.mx

Resumen. Este artículo describe un algoritmo de navegación móvil aplicándolo en simulación robótica de trabajo en sociedad para lograr visitar en su totalidad un escenario delimitado.

1 Introducción

Actualmente la robótica ocupa un lugar importante en las actividades del hombre. Desde los procesos industriales hasta el hogar. Las barreras económicas y tecnológicas están siendo superadas y se está dando un auge en las técnicas y algoritmos de inteligencia artificial aplicadas a los robots. Los diseños mecánicos y algoritmos utilizados en los robots están siendo basados en la naturaleza [1]. Desplazamiento de un móvil basado en una araña, sistemas de estabilización neumático establecido a partir de la estructura de la cucaracha [2].

Por otro lado, algoritmos simples del comportamiento animal están siendo aplicados en la robótica y sistemas computacionales. El comportamiento grupal de las abejas exploradoras se está investigando en la aplicación de encontrar rutas óptimas para la transferencia de información en Internet y redes de comunicación [3]. Entonces, en una gran parte de la investigación en robótica móvil esta siendo orientada a la utilización y aplicación de características de seres vivos y este artículo describe el comportamiento de un algoritmo de navegación pseudo aleatorio en el cual existe una cooperación grupal por parte de los robots en la navegación en un escenario.

Un grupo de robots pueden trabajar cooperativamente para lograr un objetivo. Hasta el momento esta sociedad se basa en distribución del trabajo, especializándose cada robot en una tarea. Este tipo de organización ha demostrado ser eficaz, teniendo sus aspectos negativos, como el que si uno de los robots falla, el trabajo no puede ser finalizado.

2 Descripción del problema y algoritmo de navegación

Se tiene un escenario delimitado en el cual actúan tres móviles autónomos. Cada uno posee las mismas características como dimensiones físicas, velocidad y algoritmo de navegación.

El objetivo de los móviles es navegar a través del escenario y en conjunto visitar completamente a este escenario.

En la naturaleza se tienen a las hormigas y abejas, que tienen que salir al exterior de su hormiguero o colmena y buscar alimento. La navegación es hasta cierto punto aleatoria, por parte de las abejas buscan las flores que para ellas están marcadas por emisión ultravioleta de la luz solar. En el caso de las hormigas, buscan su alimento sin que se encuentre marcado por algún factor en especial.

Para ambos casos, se tienen varias hormigas o abejas que realizan este trabajo, es una colaboración para lograr un objetivo [4, 5].

Para nuestro caso, se utiliza una simulación de tres móviles los cuales tienen idénticas dimensiones y velocidad de navegación, así como el algoritmo de navegación. El algoritmo que se utiliza es relativamente simple y se puede observar en la Figura siguiente.

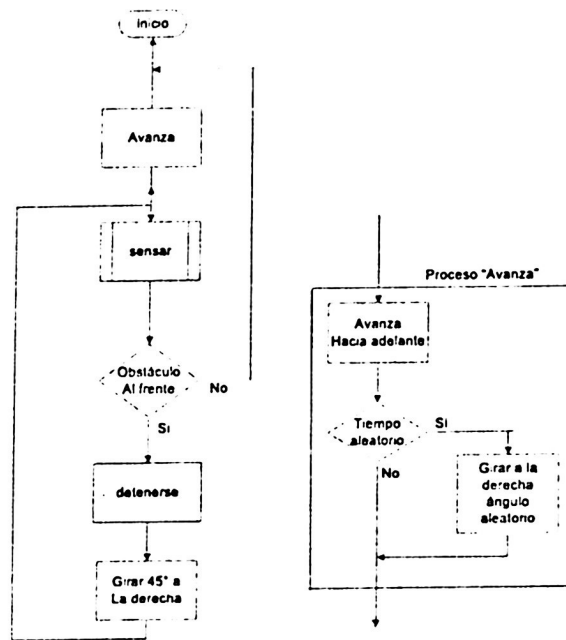


Figura 1. Escenario sin obstáculos donde el área de color negro pertenece al área visitada por los móviles.

Mientras el móvil no tenga obstáculos al frente, este avanza, si existe un obstáculo este se detiene y gira 45° a la derecha, si aún existe un obstáculo al frente, nuevamente vuelve a girar 45° a la derecha y así hasta tener campo libre al frente y poder avanzar. El algoritmo posee un tiempo aleatorio en el cual el móvil gira un ángulo aleato-

rio. Esto permite que no existan rutas cicladas y se quede el móvil recorriendo siempre el mismo camino.

3 Simulador

Se construye una simulación en lenguaje C, considerando 1, 2, 3 y 5 móviles dentro de un escenario sin obstáculos. Se aplica el algoritmo descrito en la anterior sección. Por facilidad en la programación se utilizan móviles circulares con diámetro de 20 píxeles. Para las pruebas realizadas con el simulador se estima una velocidad promedio de los móviles de 100 píxeles/seg. Al simulador se le plica que genere el tiempo aleatorio para realizar cambio de dirección del móvil en el intervalo de 1-100 segundos. También es posible almacenar en archivos de texto, información como % de área visitada por los móviles, tiempo, coordenadas de cambio de posición de los móviles, así como si velocidad.

Inicialmente se posicionan los móviles en posiciones aleatorias. El escenario sin visitar se marca con color rojo. Cuando un móvil se desplaza dentro del escenario, la zona por donde paso es pintada de color negro. Esta diferencia de colores se utiliza para obtener el porcentaje de área visitada. La Figura 2 muestra una prueba de simulación con 3 móviles después de algunos segundos de haber iniciado. Un aspecto en común para las pruebas realizadas es que, en los primeros segundos, se visita rápidamente gran parte del escenario, alrededor de 90%. Posteriormente los móviles pasan repetidamente por lugares ya visitados y no incrementa el porcentaje de área visitada.

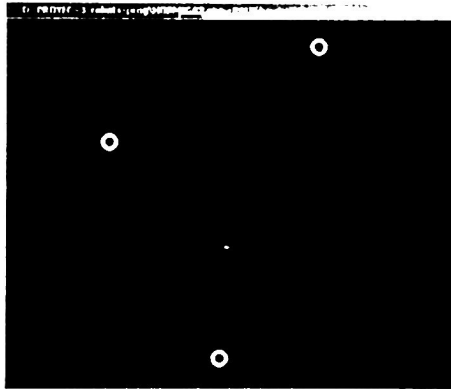


Figura 2. Escenario sin obstáculos donde el área de color negro pertenece al área visitada por los móviles.

Para visitar al menos el 98% de área del escenario, cada prueba tiene sus características. Mientras mayor sea el número de móviles más rápido se visita el escenario. La Figura 3 muestra este comportamiento. La línea para el comportamiento de un móvil, muestra un aumento de pendiente repentina en varias ocasiones a lo largo del

tiempo, estas variaciones de aumento de pendiente se deben al cambio de trayectoria en el tiempo aleatorio señalado en la simulación.

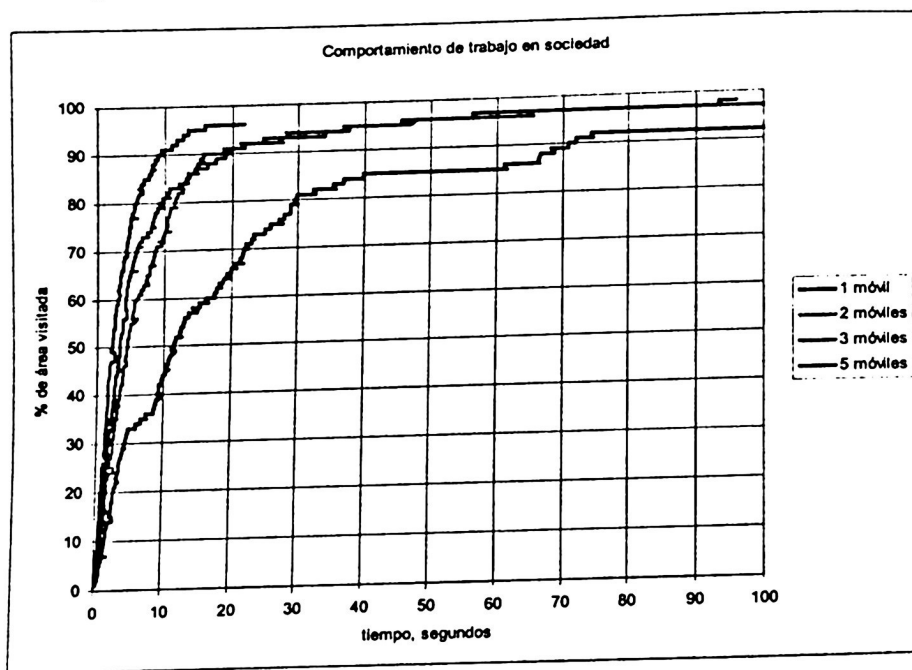


Figura 3. Comportamiento de %de área visitada con respecto al tiempo en función del número de móviles que intervengan.

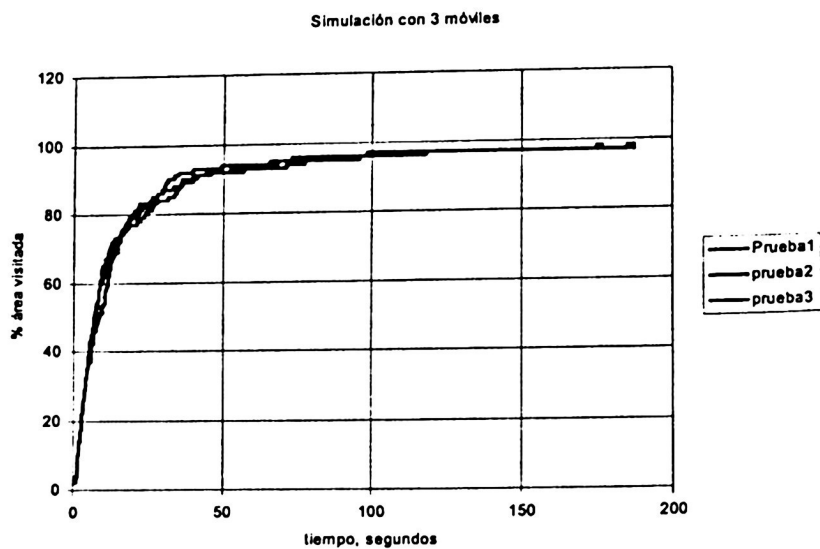


Figura 4. Similitud de simulación con 3 móviles.

El cambio de dirección aleatorio en un tiempo aleatorio, ayuda a no ciclar las trayectorias de los móviles.

Los tiempos de prueba se acortan en función al número de móviles utilizados en la simulación. Para el caso de la simulación con un móvil, el proceso de simulación se ejecuto aproximadamente por 90 minutos, sin superar el 97% de área visitada.

Cabe señalar que para cada situación de número de móviles se realizaron múltiples ejecuciones del software, comportándose de manera similar y teniendo sus cambios de pendiente en diferente tiempo, Figura 4.

4 Modelo probabilístico

Un modelo probabilístico de navegación utilizando las reglas antes descritas se describe a continuación y será válido para un móvil.

Suponiendo un móvil de área rectangular A_0 , recorriendo un escenario, podríamos conocer el área que ha recorrido el móvil si se conoce el ancho de este y la distancia que ha recorrido en un tiempo dado. Suponiendo que el móvil se desplaza con movimiento lineal rectilíneo, se tiene:

$$A = xd \quad (1)$$

Siendo x la distancia que recorre el móvil y d el ancho del mismo.

Ahora bien, si se tiene un objeto en el escenario de manera tal que al chocar el móvil con este, tuviera una desviación de 45 grados a la derecha con respecto a su dirección de avance, el área que recorrería el móvil se calcula como:

$$A_T = A_1 + x_2 d - A_0 \quad (2)$$

Donde A_1 es el área recorrida por el móvil antes de colisionar con un obstáculo, x_2 es la distancia recorrida después de la colisión. El área del móvil A_0 es necesario restarla ya que al girar el móvil debido al obstáculo y empezar nuevamente el recorrido se volvería a sumar esta área por la intersección de de las dos trayectorias, antes y después de la colisión.

Generalizando, si se detectan m obstáculos dentro del escenario, se tendrán $m+1$ recorridos del móvil dentro de la superficie. En principio el área total recorrida será:

$$A_T = \sum_{i=1}^{m+1} A_i - mA_0 \quad (3)$$

en esta aproximación aún no se han considerado los cruces de trayectoria. Si se tienen n de estos cruces se tendrán que restar del área total considerada, entonces la expresión 3 queda:

$$A_T = \sum_{i=1}^{m+1} A_i - (m+n)A_0 \quad (4)$$

Las paredes del escenario se consideran como un obstáculo. Por otro lado ya que se ha considerado un movimiento rectilíneo uniforme, es posible usar la ecuación que describe tal movimiento para obtener el área visitada en función de parámetros conocidos, es decir la velocidad del móvil. Y dejar la expresión en términos del tiempo de recorrido.

La expresión de movimiento rectilíneo uniforme es:

$$x = vt \quad (5)$$

Donde v es la velocidad del móvil y t el tiempo de recorrido. Entonces, la expresión (4) se transforma en:

$$A_T = \sum_{i=1}^{m+1} vt_i d - (m+n)A_0 \quad (6)$$

la velocidad es una constante y definiendo al tiempo como $T = \sum_{i=1}^{m+1} t_i$ la ecuación anterior se convierte en

$$A_T = vTd - (m+n)A_0 \quad (7)$$

De donde se puede obtener el tiempo total de recorrido

$$T = \frac{A_T + (m+n)A_0}{vd} \quad (8)$$

Por otro lado, en el algoritmo existe un factor de tiempo aleatorio que modifica la trayectoria del móvil. Este término puede ser representado mediante la teoría de caminata al azar. Entonces para un objeto que se mueve k pasos en una dirección, la posición después de esos pasos será [6]:

$$\begin{aligned} x &= (k - (N - k))s \\ x &= (2k - N)s \end{aligned} \quad (9)$$

donde S es el tamaño del paso y $N-k$ el número de pasos hacia el frente. Para nuestro problema el tamaño de paso se puede considerar como la longitud total del móvil y cada vez que avanza el móvil se da un paso. Considerando que el móvil es cuadrado y tiene un área de A_0 , se puede suponer que $s = \sqrt{A_0}$ y $N=n$ el número de ocasiones en que se repite un área. De tal manera que

$$x = (2k - n)\sqrt{A_0} \quad (10)$$

por otro lado, en la caminata al azar se puede considerar que está o no en la posición x determinada por una probabilidad de tipo binomial, así que ésta estará dada en términos antes considerados:

$$P = \binom{n}{k} p^k q^n = \frac{n!}{(n-k)!k!} p^k q^n \quad (11)$$

con p la probabilidad de dar k pasos y $q=1-p$. k se encuentra de despearle de (10):

$$k = \frac{n}{2} + \frac{x}{2\sqrt{A_0}} \quad (12)$$

Además de

$$n-k = \frac{n}{2} - \frac{x}{2\sqrt{A_0}} \quad (13)$$

Esta expresión puede dejarse en términos de la expresión (8), así que la probabilidad se puede obtener en función del área recorrida y la misma expresión podría quedar en términos del tiempo de recorrido del móvil.

$$P = \frac{n!}{\left(\frac{n}{2} - \frac{x}{2\sqrt{A_0}}\right)! \left(\frac{n}{2} + \frac{x}{2\sqrt{A_0}}\right)!} p^{\frac{n}{2} + \frac{x}{2\sqrt{A_0}}} q^n \quad (14)$$

Esta expresión es la probabilidad de que el móvil se encuentre en la posición x y se recorra un área en un tiempo dado.

Experimentalmente en base a las simulaciones realizadas se encuentra mediante regresión polinomial y análisis de trayectorias, Figura 4, que el número de obstáculos está expresado mediante:

$$m = 0.43T^2 + 0.4T + 0.08 \quad (15)$$

Mientras que el número de cruces del móvil por áreas visitadas con anterioridad con respecto al tiempo se expresa con:

$$n = 0.1T^4 + 1.25T^3 - 3.93T^2 + 4.05T + 0.0002 \quad (16)$$

Además de que se tiene que el ancho del móvil, $d=20$ pixeles, y un área $A_0=400\text{pixeles}^2$ y una velocidad promedio de $v=100$ pixeles/seg.

Del análisis de datos de trayectoria, con respecto ala probabilidad de que el móvil de k pasos o secciones de trayectoria, tienen la misma probabilidad, 40% aproximadamente, de que ocurra tanto para cortas longitudes (30 pixeles) como para largas (180 pixeles); teniéndose baja probabilidad para longitudes de trayectorias que se encuentren entre estas longitudes.

La siguiente tabla muestra los resultados de la evaluación del modelo teórico con respecto a los datos obtenidos de la simulación.

Tabla 1. Resultados de la Evaluación del Modelo Teórico.

tiempo	% área cal.	% área sim.
1	2.26639804	1
2	4.91038983	9
3	7.55438162	17
4	10.1983734	25
5	12.8423652	31
10	26.0623242	54
20	52.5022421	69

5 Mapeo de escenarios

De manera experimental se utilizan 3 móviles autónomos con el algoritmo de navegación programado en un PIC. El móvil posee los sensores necesarios para detectar obstáculos y con el algoritmo se deciden trayectorias de navegación. Cada móvil tiene comunicación unidireccional con una computadora mediante un modulo de radio frecuencia. Con lo cual, se envían datos de avance, trayectoria ejecutada y detección de obstáculos por cada móvil a una computadora. Esta se encarga de analizar y sincronizar datos de tal manera que se puedan registrar puntos de detección de objetos y graficar estos para generar a lo largo del tiempo un mapa del escenario donde se desenvuelven los móviles.

La computadora recibe los datos de cada móvil. Con estos se crea un mapa indivi-

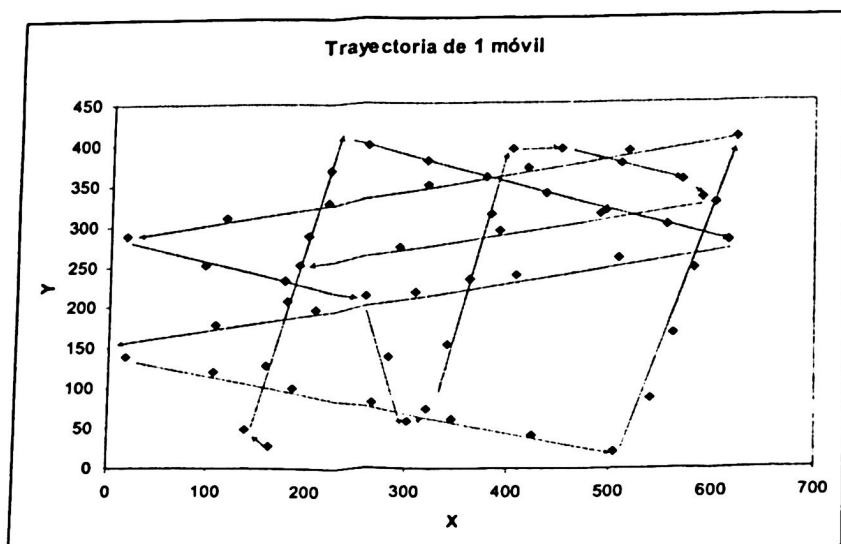


Figura 5. Trayectoria de un móvil ejecutada en un tiempo de 5.19 segundos.

dual de cada móvil. Mediante una red neuronal con entrenamiento hacia atrás, back-propagation, se analizan los mapas individuales, buscando coincidencias, de tal manera que una vez que se tenga los suficientes datos para poder encontrar coincidencias entre mapas con un error. Los gráficos que representan estos mapas se mezclan dando un mapa resultante.

Se utiliza un entrenamiento de la red neuronal con 10 posibilidades de coincidencia, tomando como referencia, esquinas y dimensiones de objetos. El tipo de red neuronal utilizado permite tener salidas de coincidencia aun con errores grandes. Lo cual da la posibilidad de que en tiempo de ejecución del sistema, se pueda corregir el mapa generado al inicio de la obtención de datos.

Por las características de diseño de los móviles, el trazar un mapa de un área de $5m^2$ tiene una duración aproximada de 15 minutos, Figura 5.

6 Conclusiones

Mediante la simulación en un escenario con 1 o más robots, se muestra que el algoritmo propuesto, aunque simple logrará que el móvil visite toda el área donde se encuentre. El factor a considerar es la rapidez con que lo realice, ya que varios móviles cubrirán el área del escenario de una manera más rápida que siendo menor su número.

El algoritmo implica que el móvil repetirá trayectorias en el escenario, pero el móvil no tiene que recordar por donde ya ha pasado; y este aspecto simplifica cuestiones de memoria y algoritmos complejos de navegación, lo cual repercute directamente en el costo de un móvil real, con elementos como microprocesadores, memoria, tiempo de respuesta a nuevos datos, etc.

Otra característica importante de este algoritmo es que, el móvil no necesita conocer previamente el escenario.

Este simple algoritmo, ejemplifica el comportamiento de las hormigas, las cuales simplemente por olfato y tacto navegan en su ambiente. Esto implica que los móviles a construir con este tipo de algoritmos tendrán entrada de datos, sensores de presencia, como micro switch o sensores ópticos de corto alcance.

El modelo teórico únicamente introduce aspectos superficiales de la teoría de caminata al azar, lográndose una aproximación de estimación de área en función de tiempo esencialmente de tipo lineal. Mientras que las simulaciones muestran un comportamiento más bien logarítmico. Y a esto se deben los errores mostrados en la comparación de datos de simulación y estimados. Por consiguiente, el siguiente paso con respecto a un modelo teórico es la introducción de un mayor número de conceptos de caminata al azar.

Este tipo de algoritmo se utiliza en la actualidad para barredores robot de piso caseros. Donde no importa tanto el tiempo en que realice el trabajo, sino que cubra la totalidad de superficie del escenario, sin que previamente conozca éste.

Por otro lado este algoritmo se encuentra en fase de investigación para la creación de mapas de un escenario. Donde se aplican las características de que el móvil no conoce el escenario, el tiempo no es factor predominante.

En esta aplicación el móvil posee el algoritmo descrito en este artículo y el sensor de obstáculos registra las coordenadas de este y en otra ventana se grafican los puntos donde se observó tal obstáculo.

Para finalizar, este algoritmo, aún que no nuevo, puede tener una gran variedad de aplicaciones, simplificando requerimientos de hardware en los móviles y simplificando la programación de estos.

References

1. Randall D. Beer, Roger D. Quinn, Hillel J. Chiel, and Roy E. Ritzmann; Biologically Inspired approaches to robotics"; Communications of the ACM; Vol. 40; No.3; March (1997); 31-38.
2. E. V. Krishnamurthy; V. K. Murthy, Vikram Krishnamurthy; Biologically Inspired Rule-Based Multiset Programming Paradigm for Soft-Computing; CF'2004 ACM; april (2004); 140-149.
3. Bonabeau, Théraulaz; "Swarm Smarts"; Scientific American; March (2000).
4. Tucker Balch, Zia Khan and Manuela Veloso; Automatically Tracking and Analyzing the Behavior of Live Insect Colonies; Agents 01 ACM; 28 May (2001); 521-528.
5. Lam, Y.K. Wong, E.K. Loo, C.K.; Explicit communication in designing efficient cooperative mobile robotic system; Proceedings IEEE International Conference on Robotics and Automation, 2003; Vol. 3; 14-19 Sept. (2003); 3869 – 3874.
7. A. M. Odizko; Search for the maximum of a random walk; ACM; (1994); 336-345.