

Spatially-dependent time synchronization of mobile robots simulating a firefly environment ^{*}

Brian Javier Aguilera-Cedillo^{*} Adrian Arellano-Delgado^{**,***}
César Cruz-Hernández^{*} Rosa Martha López-Gutiérrez^{***}

^{*} *Electronics and Telecommunications Department, CICESE, 22860
Baja California, Mexico (e-mail: baguilera@cicese.edu.mx,
ccruz@cicese.mx).*

^{**} *Secretariat of Science, Humanities, Technology and Innovation,
03940 Ciudad de Mexico, Mexico (e-mail:
adrian.arellano@uabc.edu.mx)*

^{***} *Enginnering, Architecture and Design Faculty, UABC, 22860 Baja
California, Mexico (e-mails: roslopez@uabc.edu.mx).*

Abstract: In this work we have a spatially dependent time synchronization with differential traction type mobile robots, which will simulate a firefly network environment. In particular, we have in a virtual environment a group of terrestrial mobile robots in which the synchronization of different oscillating signals emitted by each mobile robot is intended to be achieved. The conventional control law proposed to synchronize the oscillating signals coupled to the robots will depend on the position in space of the robots, which may be synchronized (in phase or anti-phase) or not, to a periodic or randomly generated trajectory.

Keywords: Synchronization, Oscillators, Mobile robot, Linearization by dynamic feedback.

1. INTRODUCCIÓN

La sincronización en la frecuencia de los destellos de las luciérnagas durante la época de apareamiento ocurre cuando los machos emiten señales luminosas de forma coordinada, permitiendo a las hembras reconocer patrones específicos de su especie, como lo demostraron Buck and Buck (1978) en sus estudios pioneros sobre bioluminiscencia sincronizada.

En este trabajo, se modela dicha interacción en un entorno de simulación con múltiples robots móviles de tracción diferencial. Cada robot sigue trayectorias pseudoaleatorias, emulando los movimientos naturales de las luciérnagas. Debido a su configuración, el control del movimiento requiere regular tanto la aceleración de cada rueda como la dirección general. Para ello, se emplea un esquema de control por retroalimentación dinámica. A cada robot se le asigna una señal sinusoidal de frecuencia única, representando los destellos.

La sincronización se logra mediante un esquema maestro-esclavo, donde los robots ajustan sus frecuencias a la de un robot principal que simula a la luciérnaga dominante. Cuando dos robots secundarios se aproximan, sus frecuencias se acoplan temporalmente hasta reencontrarse con el maestro, produciendo un fenómeno de sincronización emergente similar a un sistema de mundo pequeño.

El enfoque inspirado en luciérnagas resulta especialmente conveniente frente a otros sistemas multi-robot, como el modelo de Kuramoto (1984), limitado por el consumo energético. Aunque en años recientes se han explorado técnicas basadas en aprendizaje por refuerzo para la sincronización explícita de osciladores como la sincronización óptima en sistemas no lineales mediante aprendizaje distribuido en el trabajo de Guo et al. (2025) o la sincronización en tiempo finito con métodos actor-crítico en sistemas armónicos en el trabajo de Gu et al. (2024), estos enfoques suelen requerir alta complejidad computacional y un modelado detallado del sistema.

En contraste, los modelos bioinspirados como el de Buscarino et al. (2009) ofrecen soluciones más ligeras, escalables y eficientes en consumo energético, al basarse en interacciones locales dependientes de la proximidad espacial. De manera similar, enfoques contemporáneos como el modelo de forrajeo inspirado en peces de Li and Yang (2024) demuestran que la sincronización emergente puede alcanzarse mediante interacciones locales y dinámicas neurobiológicas, sin necesidad de control centralizado.

2. TOPOLOGÍA DE CONEXIÓN

Se empleará una topología de conexión en forma de estrella ya que, al tratarse de un grafo no cíclico, resulta adecuado para el sistema de control que se utilizará en el proceso de sincronización. En la figura 1, cuatro robots se sincronizan con un único robot maestro mediante un sistema maestro-esclavo, según lo propuesto por Arellano-Delgado et al. (2015) y López-Parra (2017).

^{*} This work was supported by the Secretariat of Science, Humanities, Technology and Innovation (SECIHTI).

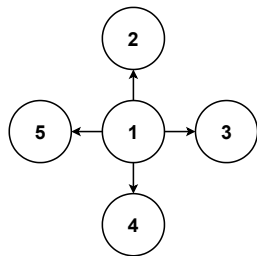


Fig. 1. Topología de conexión de red tipo estrella con 5 nodos (modo maestro/esclavos).

2.1 Matriz de acoplamiento

Para el sistema representado en la figura 1, la correspondiente matriz de acoplamiento $A(G)$ es la siguiente:

$$A = \begin{bmatrix} -4 & 1 & 1 & 1 & 1 \\ 1 & -1 & 0 & 0 & 0 \\ 1 & 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & -1 & 0 \\ 1 & 0 & 0 & 0 & -1 \end{bmatrix}. \quad (1)$$

3. MODELO CINEMÁTICO DEL ROBOT MÓVIL TIPO TRACCIÓN DIFERENCIAL

El modelo utilizado se basa en las ecuaciones de control cinemático desarrolladas para un robot móvil con una sola rueda, las cuales permiten describir su comportamiento dinámico de forma simplificada. Dichas ecuaciones han sido formuladas y aplicadas en trabajos previos, como el de Šuster and Jadlovska (2010), y se expresan de la siguiente manera:

$$\begin{aligned} \dot{x} &= v \cos \theta, \\ \dot{y} &= v \sin \theta, \\ \dot{\theta} &= \omega. \end{aligned} \quad (2)$$

Donde v y ω representan las entradas del sistema, correspondientes a la velocidad lineal y velocidad angular del robot, respectivamente. Este modelo puede adaptarse a un robot tipo tracción diferencial como el mostrado en la figura 2, permitiendo su control mediante la regulación de dichas velocidades de la siguiente manera:

$$v = \frac{vL + vR}{2}, \omega = \frac{vR - vL}{l}. \quad (3)$$

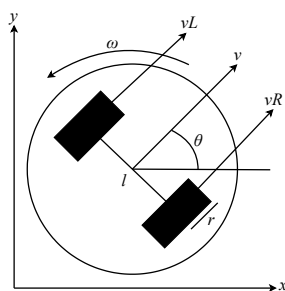


Fig. 2. Plano de movimiento de un robot diferencial.

De este modo, es posible controlar el movimiento de un robot diferencial mediante la modificación de las velocidades lineal y angular, tomando en cuenta la separación entre las ruedas y la velocidad de giro independiente de cada rueda.

4. CONTROL DE LINEALIZACIÓN POR RETROALIMENTACIÓN DINÁMICA

Dado que en (2) y (3) las variables de control del movimiento del robot móvil diferencial aún se encuentran acopladas, es necesario desacoplarlas mediante un controlador basado en linealización por retroalimentación dinámica como lo propone De Luca et al. (2000) con la siguiente estructura:

$$\begin{aligned} \xi &= a(R, \xi) + b(R, \xi)u^*, \\ \omega &= c(R, \xi) + d(R, \xi)u^*. \end{aligned} \quad (4)$$

Donde ξ y R representan las variables de estado del sistema, las cuales, al tratarse de un sistema mecánico, se utilizan para calcular la posición y velocidad del robot móvil. Los parámetros a , b , c y d corresponden a ganancias del sistema, y u^* es el vector de entradas.

Dado que se dispone de un vector de salidas $\eta = (x, y)$, y considerando las ecuaciones cinemáticas definidas en (2), es posible reconstruir la matriz de velocidad del robot a partir de la primera derivada temporal de estas salidas:

$$\dot{\eta} = \begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} = \begin{bmatrix} \cos(\theta) & 0 \\ \sin(\theta) & 0 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix}, \quad (5)$$

$$\dot{\eta} = v \begin{bmatrix} \cos(\theta) \\ \sin(\theta) \end{bmatrix}. \quad (6)$$

Para incluir la aceleración del robot en la matriz, es necesario realizar una segunda derivada temporal. De este modo, se obtiene la siguiente matriz, considerando que $\xi = v$ (velocidad) y que su derivada temporal $\dot{v} = a$ corresponde a la aceleración:

$$\ddot{\eta} = a \begin{bmatrix} \cos(\theta) \\ \sin(\theta) \end{bmatrix} + \xi \cdot \omega \begin{bmatrix} -\sin(\theta) \\ \cos(\theta) \end{bmatrix}. \quad (7)$$

Expresando (7) de manera matricial se tiene:

$$\ddot{\eta} = \begin{bmatrix} \ddot{x} \\ \ddot{y} \end{bmatrix} = \begin{bmatrix} \sin(\theta) & -\xi \sin(\theta) \\ \cos(\theta) & \xi \cos(\theta) \end{bmatrix} \begin{bmatrix} a \\ \omega \end{bmatrix} = \begin{bmatrix} u_1^* \\ u_2^* \end{bmatrix}. \quad (8)$$

Al reorganizar las variables de estado de la matriz (8) mediante operaciones algebraicas, se obtiene la siguiente expresión:

$$\begin{bmatrix} a \\ \omega \end{bmatrix} = \begin{bmatrix} \cos(\theta) & \sin(\theta) \\ -\sin(\theta) & \cos(\theta) \end{bmatrix} \begin{bmatrix} u_1^* \\ u_2^* \end{bmatrix}. \quad (9)$$

Con ello se obtienen las siguientes ecuaciones, las cuales permiten controlar la aceleración (a) y la velocidad angular (ω) del robot móvil diferencial. Este procedimiento transforma el sistema original no lineal en uno lineal, ya que las variables a calcular aparecen únicamente en sus propias ecuaciones, lo que facilita su análisis y control.

$$\begin{aligned} a &= u_1^* \cos(\theta) + u_2^* \sin(\theta), \\ \omega &= \frac{-u_1^* \sin(\theta) + u_2^* \cos(\theta)}{\xi}. \end{aligned} \quad (10)$$

Este compensador dinámico transforma las entradas de control u^* en variables desacopladas del sistema (ξ, ω) , lo cual es muy útil porque los robots móviles diferenciales tienen no linealidades y restricciones no holonómicas, lo que complica el control directo. Aplicando este control se obtiene un modelo lineal equivalente sobre el cual es posible aplicar técnicas como control predictivo basado en modelo, seguimiento de trayectorias o control robusto. Diferentes trabajos de investigación como Thai et al. (2022), Ali et al. (2024) y Tran and Vu (2024) los cuales utilizan robots móviles diferenciales aplican linealización por retroalimentación dinámica para el seguimiento de trayectorias por lo que sigue siendo una opción sólida para su control.

5. SIMULACIONES DE TRAYECTORIAS

Para validar el correcto funcionamiento del modelo cinemático del robot móvil diferencial, junto con el controlador propuesto, se realizaron simulaciones en Matlab. Dado que se trabajó en un entorno virtual, fue necesario establecer las siguientes tres suposiciones:

- Ausencia de fricción y peso: El robot no presenta fricción ni peso que interfieran con el movimiento de las ruedas.
- Condiciones iniciales libres: La posición y la orientación angular inicial del robot se definieron como parámetros configurables.
- Control total de velocidad y aceleración: Es posible asignar de manera libre los valores de velocidad y aceleración del robot. En consecuencia, en ciertos casos el robot puede completar sus trayectorias en un tiempo menor al que tomaría en condiciones reales.

5.1 Trayectoria de rosa polar

La trayectoria seleccionada para la simulación corresponde a una rosa polar, ya que, al presentar múltiples curvas (como se muestra en la figura 3), permite evaluar si el sistema de control de movimiento por retroalimentación es capaz de gestionar adecuadamente la aceleración y desaceleración del robot al desplazarse a lo largo de dichas curvas. Esta capacidad de control puede observarse en la figura 4.

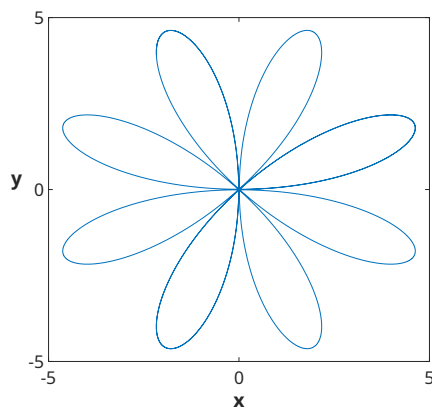


Fig. 3. Trayectoria en el espacio por el robot diferencial para la rosa polar.

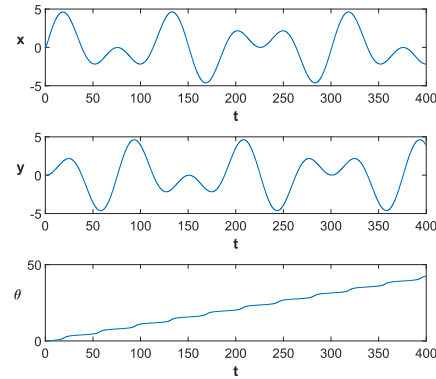


Fig. 4. Movimiento temporal para la trayectoria de la rosa polar.

6. SISTEMA GENERADOR DE RUTAS ALEATORIAS

6.1 Ecuación generadora de rutas aleatorias

Una vez verificado que el sistema de control de movimiento propuesto funciona correctamente con trayectorias predefinidas, el siguiente paso consiste en incorporar un generador de rutas aleatorias. Para ello, se utiliza la función $rand()$ de Matlab, que permite crear trayectorias no deterministas para evaluar la capacidad del sistema de adaptarse a condiciones más variables.

$$\begin{aligned} f_1(n) &= a + (b - a) * rand(1, 1), \\ f_2(n) &= a + (b - a) * rand(1, 1). \end{aligned} \tag{11}$$

Las funciones f_1 y f_2 se encargan de generar coordenadas aleatorias en los ejes x y y , respectivamente, con el fin de crear los puntos de ruta aleatorios. Las variables a y b definen los límites espaciales dentro de los cuales se generan dichos puntos, correspondientes al área de simulación en la que se desplazan los robots móviles. Este sistema opera mediante una condición temporal, que permite generar un nuevo punto de ruta aleatorio en intervalos de tiempo predefinidos. El proceso se detiene automáticamente una vez que se alcanza el tiempo total de simulación. Por lo tanto, la cantidad total de rutas generadas está directamente relacionada con el intervalo de tiempo utilizado.

6.2 Simulación con destinos aleatorios

Al igual que con la trayectoria de la rosa polar, se llevó a cabo una simulación en un espacio de 10 x 10 metros para verificar el funcionamiento del sistema generador de rutas aleatorias. Como resultado, se obtuvo la trayectoria aleatoria mostrada en la figura 5. En el análisis del movimiento temporal (figura 6), se observa que las rotaciones del robot son más angulosas o cuadradas en comparación con la trayectoria anterior. Esto se debe a que el robot móvil ajusta su orientación únicamente al llegar a cada punto de destino, y luego se desplaza hacia el siguiente punto en línea recta, lo que genera cambios bruscos en la dirección del movimiento.

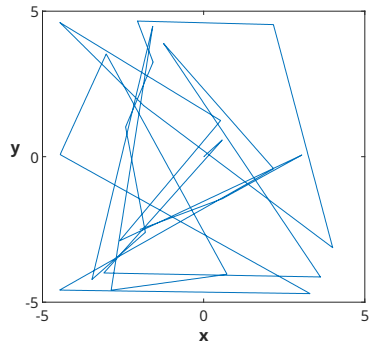


Fig. 5. Trayectoria por el robot diferencial para una trayectoria aleatoria.

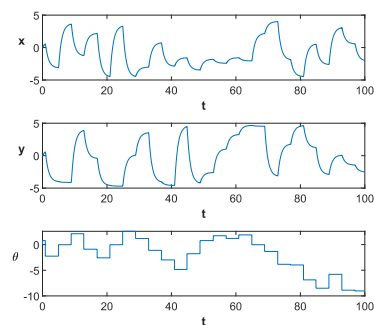


Fig. 6. Movimiento temporal para una trayectoria aleatoria.

7. SISTEMA ANTICOLISIONES

Para hacer uso de múltiples robots se implementó un sistema anticollisiones basado en condiciones de proximidad, siguiendo la propuesta de Aguilera-Cedillo (2024). El principio consiste en que, si uno o más robots se aproximan por debajo de una distancia umbral, sus trayectorias se modifican mediante la generación de un nuevo punto de ruta aleatorio, evitando colisiones.

Sean $R_1 = (x_1, y_1)$ y $R_2 = (x_2, y_2)$ las posiciones de dos robots en el plano. La distancia entre ellos se define como: $d = \overline{R_1 R_2} = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$, por lo que la fórmula utilizada para el sistema anticollisiones entre dos robots es la siguiente:

$$\begin{aligned} X \rightarrow d &= \overline{R_1 R_2} \\ \leq Y &\rightarrow \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \leq Y. \end{aligned} \quad (12)$$

Donde Y representa un valor arbitrario elegido apropiadamente como umbral de distancia mínima permitida. Si se cumple la condición establecida en (12), el sistema anticollisiones no se activa. Sin embargo, si la distancia entre los robots es menor o igual a Y , entonces el sistema procede a generar una nueva ruta con el fin de evitar una posible colisión.

7.1 Simulación con anticollisiones

Se realizó una simulación en un espacio de 10×10 metros con dos robots móviles de trayectorias aleatorias. Los resultados muestran curvas repentinas en las trayectorias

en la figura 7, producto de la activación del sistema cuando el robot 2 se aproxima al robot 1. Las Figuras 8 y 9 evidencian estas interacciones: perturbaciones en la orientación del robot 2 y la activación del sistema al alcanzarse una distancia menor o igual a 1 metro.

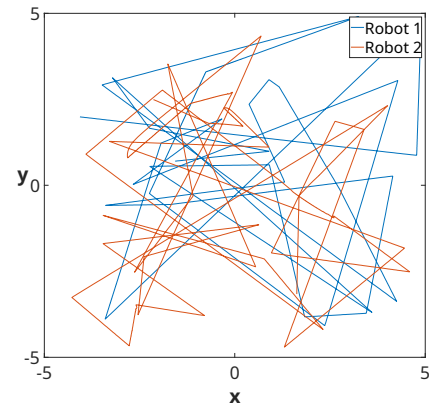


Fig. 7. Trayectoria en el espacio por 2 robots diferenciales para trayectorias aleatorias.

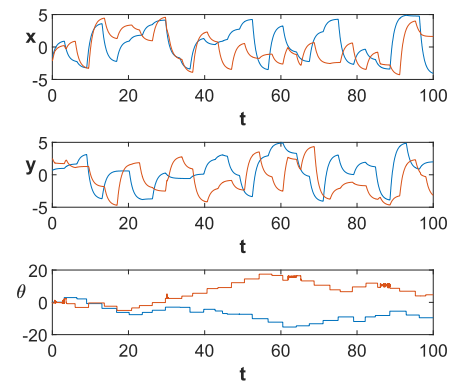


Fig. 8. Movimiento temporal realizado por 2 robots diferenciales para trayectorias aleatorias.

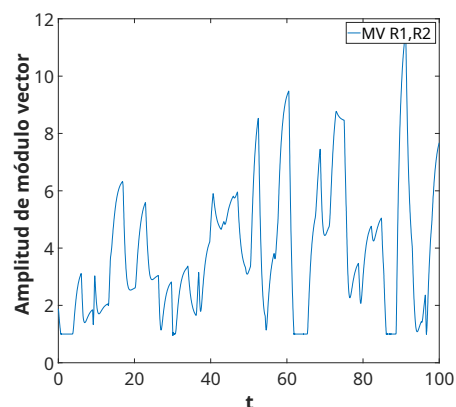


Fig. 9. Modulo vector de los 2 robots diferenciales.

8. SISTEMA DE SINCRONIZACIÓN

8.1 Señal para la sincronización

Con el fin de emular el fenómeno de sincronización de las luciérnagas, se definió el siguiente modelo de onda sinusoidal:

$$Luci\acute{e}rnaga = A * \sin(2 * \pi * f * t). \quad (13)$$

A partir (13) se asignan diferentes frecuencias a las señales de los robots diferenciales, permitiendo su posterior sincronización.

8.2 Método de sincronización

La sincronización de los osciladores en robots con tracción diferencial se basa en el módulo vectorial de la distancia entre robots, previamente utilizado en el sistema anticollisiones (12), activándose cuando dos robots se encuentran a una proximidad adecuada. Este mecanismo se inspira en el comportamiento natural de sincronización de las luciérnagas. Cuando la interacción ocurre entre dos robots esclavos, sus osciladores convergen hacia una frecuencia intermedia, produciendo una sincronización temporal. En contraste, si un esclavo se sincroniza con el robot maestro, adopta completamente su frecuencia; a partir de entonces, cualquier otro esclavo que se acerque se ajusta a esa misma frecuencia, lo que asegura que todos los robots converjan finalmente con el oscilador principal. A continuación, en la figura 10 se presenta un diagrama de flujo que ilustra la lógica secuencial del algoritmo completo, incluyendo los sistemas de generación de rutas aleatorias, sincronización y anticollisiones.

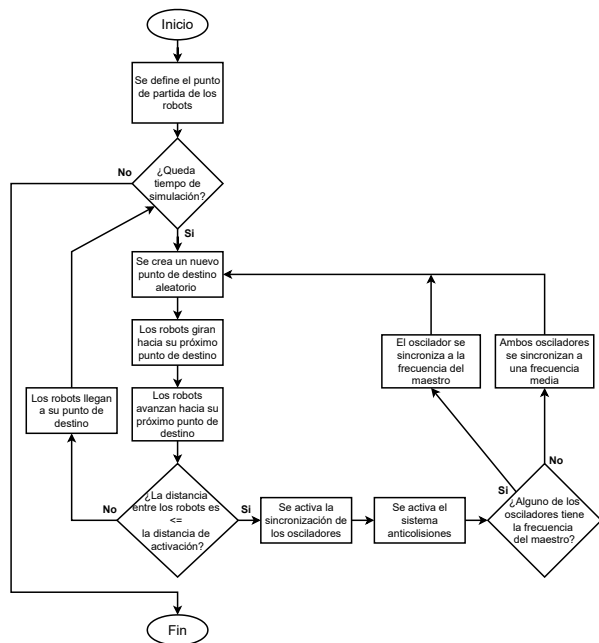


Fig. 10. Diagrama de flujo del algoritmo completo.

9. RESULTADOS FINALES

La simulación principal se realizó con cinco robots móviles que ejecutaron trayectorias aleatorias en un espacio de

40 × 40 metros (figura 11). La evolución temporal de sus movimientos y cambios de dirección se muestra en la figura 12. La figura 13 presenta el módulo vectorial de las distancias correspondientes a los 10 pares únicos de robots. Finalmente, la figura 14 ilustra el proceso de sincronización de los osciladores, donde todos convergen a la frecuencia del maestro.

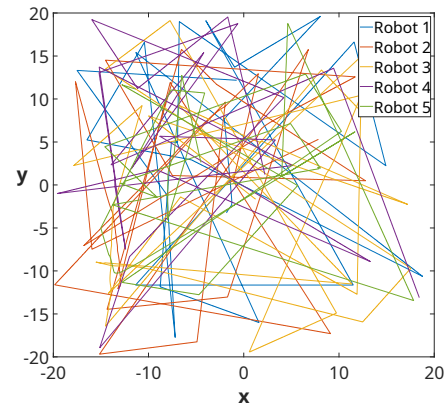


Fig. 11. Trayectoria en el espacio por 5 robots diferenciales para trayectorias aleatorias.

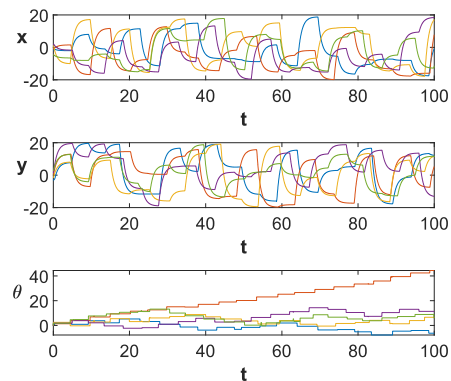


Fig. 12. Movimiento temporal realizado por 5 robots diferenciales para trayectorias aleatorias.

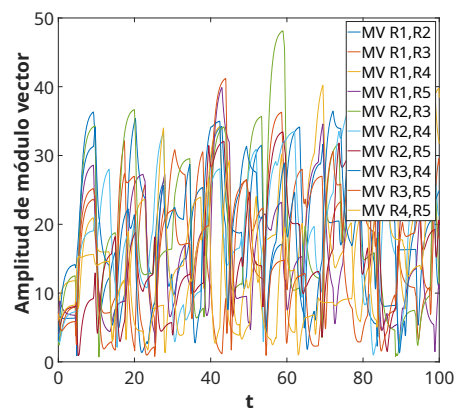


Fig. 13. Módulo vector de los 5 robots diferenciales.

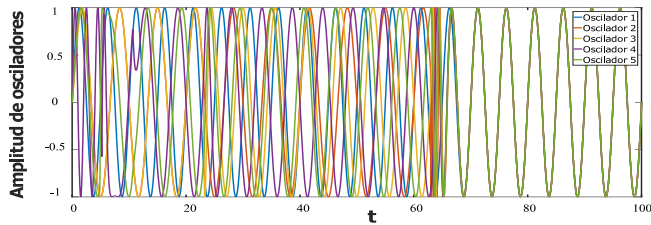


Fig. 14. Sincronización de todos los osciladores.

10. CONCLUSIÓN

Este trabajo demostró que los comportamientos de sincronización observados en las luciérnagas durante el apareamiento pueden ser replicados en un conjunto de robots móviles equipados con osciladores, capaces de simular sus destellos luminosos intermitentes. Aunque se empleó una señal sinusoidal simple por su semejanza con los patrones naturales, el sistema es compatible con señales más complejas, como las caóticas, dependiendo del objetivo del sistema. Además de la sincronización, otro aspecto relevante observado es que esta depende de la proximidad entre individuos, ya que las luciérnagas solo sincronizan sus destellos cuando se encuentran suficientemente cerca unas de otras. Este fenómeno fue aprovechado para el diseño del sistema de sincronización de los robots, que imitó la frecuencia de destello del macho que mejor cortejaba a las hembras. A su vez, esto se integró con un sistema de evasión de colisiones, dado que, a diferencia de las luciérnagas reales, los robots no poseen percepción integrada del entorno. Como trabajo futuro, se plantea la posibilidad de una implementación física del sistema, que permita observar de manera práctica —y no solo simulada— el comportamiento de sincronización en robots móviles. Asimismo, se propone ampliar el número de robots utilizados simultáneamente. En este estudio se limitaron a cinco unidades debido a restricciones de hardware, pero una mayor cantidad podría enriquecer el análisis del comportamiento colectivo.

REFERENCES

Aguilera-Cedillo, B.J. (2024). *Sincronización temporal espacialmente dependiente de robots móviles simulando un entorno luciérnaga*. Tesis de licenciatura, Universidad Autónoma de Baja California, Facultad de Ingeniería, Arquitectura y Diseño, Ensenada, Ensenada, Baja California, México. URL <https://hdl.handle.net/20.500.12930/12325>. Licenciatura.

Ali, A.M., Shen, C., and Hashim, H.A. (2024). A linear mpc with control barrier functions for differential drive robots. *IET Control Theory & Applications*, 18(18), 2693–2704. doi:10.1049/cth2.12709. URL <https://doi.org/10.1049/cth2.12709>.

Arellano-Delgado, A., Cruz-Hernández, C., López Gutiérrez, R.M., and Posadas-Castillo, C. (2015). Outer synchronization of simple firefly discrete models in coupled networks. *Mathematical Problems in Engineering*, 2015, 1–14. doi:10.1155/2015/895379. URL <https://doi.org/10.1155/2015/895379>.

Buck, J. and Buck, E. (1978). Toward a functional interpretation of synchronous flashing by fireflies. *The American Naturalist*, 112, 471 – 492. doi:10.1086/283291.

Buscarino, A., Fortuna, L., Frasca, M., and Rizzo, A. (2009). *Synchronization in Networks of Mobile Agents*, 3–25. Springer Berlin Heidelberg, Berlin, Heidelberg. doi:10.1007/978-3-642-03199-1_1. URL https://doi.org/10.1007/978-3-642-03199-1_1.

De Luca, A., Oriolo, G., and Vendittelli, M. (2000). Stabilization of the unicycle via dynamic feedback linearization. *IFAC Proceedings Volumes*, 33(27), 687–692. doi:10.1016/S1474-6670(17)38011-4. URL [https://doi.org/10.1016/S1474-6670\(17\)38011-4](https://doi.org/10.1016/S1474-6670(17)38011-4). 6th IFAC Symposium on Robot Control (SYROCO 2000), Vienna, Austria, 21-23 September 2000.

Gu, Y., Zhao, J., Sun, Z.Y., and Xie, X. (2024). Reinforcement learning-based optimized multi-agent finite-time optimal synchronisation control and its application to the harmonic oscillator. *Nonlinear Dynamics*. doi:10.1007/s11071-024-09758-7. URL <https://doi.org/10.1007/s11071-024-09758-7>.

Guo, Y., Sun, Q., Pan, Q., and Wang, Y. (2025). Pareto-optimal synchronization control of nonlinear multi-agent systems via integral reinforcement learnings. *Nonlinear Dynamics*. doi:10.1007/s11071-024-10461-w. URL <https://doi.org/10.1007/s11071-024-10461-w>.

Kuramoto, Y. (1984). *hematical Oscillations, Waves, and Turbulence*. Springer.

Li, J. and Yang, S.X. (2024). Intelligent fish-inspired foraging of swarm robots with sub-group behaviors based on neurodynamic models. *Biomimetics*, 9(1). doi:10.3390/biomimetics9010016. URL <https://www.mdpi.com/2313-7673/9/1/16>.

López-Parra, A. (2017). *Formación en grupos de robots móviles*. Tesis de maestría en ciencias, Centro de Investigación Científica y de Educación Superior de Ensenada, Ensenada, Baja California, México.

Thai, N.H., Thien, H., and Ly, T.T.K. (2022). Nurbs curve trajectory tracking control for differential-drive mobile robot by a linear state feedback dynamic controller. In A.T. Le, V.S. Pham, M.Q. Le, and H.L. Pham (eds.), *The AUN/SEED-Net Joint Regional Conference in Transportation, Energy, and Mechanical Manufacturing Engineering*, 610–623. Springer Nature Singapore, Singapore.

Tran, A.M. and Vu, T.V. (2024). A dynamic controller architecture for wheeled mobile robot trajectory tracking utilizing feedback linearization and state feedback. *Journal of Advanced Engineering and Computation*, 8(2), 119–129. doi:10.55579/jaec.202482.456. URL <https://doi.org/10.55579/jaec.202482.456>.

Šuster, P. and Jadlovska, A. (2010). Neural Tracking Trajectory of the Mobile Robot Khepera II in Internal Model Control Structure. In *Proceedings of the 9th International Conference Process Control*, C153a-1–C153a-13. Kouty nad Desnou, Czech Republic.