

UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA

FACULTAD DE INGENIERÍA, ARQUITECTURA Y DISEÑO



Diseño de Trayectorias Caóticas en robots móviles.

TESIS

que se presenta para cubrir el grado de

DOCTORADO EN CIENCIAS

Juan José Cetina Denis

DIRECTORA DE TESIS:

Dra. Rosa Martha López Gutiérrez

CODIRECTOR DE TESIS:

Dr. César Cruz Hernández

Ensenada, B.C.

Abril del 2022.

UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA

FACULTAD DE INGENIERÍA, ARQUITECTURA Y DISEÑO

Diseño de Trayectorias Caóticas en Robots Móviles

TESIS

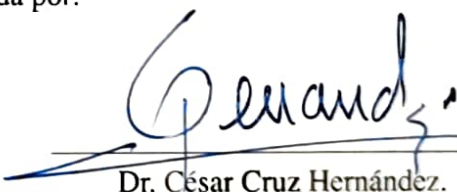
para obtener el grado de Doctorado en Ciencias

Presenta:

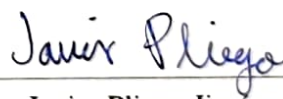
Juan José Cetina Denis

Aprobada por:



Dra. Rosa Martha López Gutiérrez.
Directora de Tesis.

Dr. César Cruz Hernández.
Codirector de Tesis.

Dr. Adrian Arellano Delgado.
Miembro del Comité.

Dr. Javier Pliego Jiménez.
Miembro del Comité.

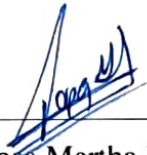
Dra. Liliana Cardoza Avendaño.
Miembro del Comité.

Ensenada, Baja California, Abril del 2022.

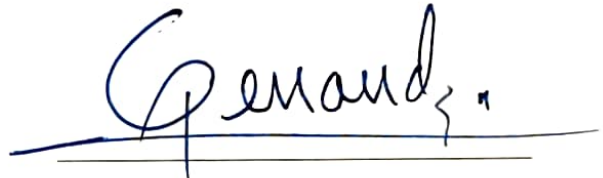
Resúmen de la tesis de **Juan José Cetina Denis**, presentada como requisito parcial para la obtención del grado de DOCTOR EN CIENCIAS del programa de Maestría y Doctorado en Ciencias e Ingeniería (MYDCI) de la UABC. Ensenada Baja California, México, Abril del 2022.

Diseño de Trayectorias Caóticas en robots móviles.

Resúmen Aprobado por:



Dra. Rosa Martha López Gutiérrez.
Directora de Tesis.



Dr. Cesar Cruz Hernández.
Codirector de Tesis.

En este trabajo de tesis se aborda el problema de diseño de trayectorias caóticas en robots móviles diferenciales. En particular, se utiliza como base el mapa caótico de Hénon, el cual induce comportamientos caóticos mediante su sincronización con las velocidades lineales y angulares de un robot móvil diferencial tipo Khepera III. Algunas de las posibles aplicaciones, aprovechando las propiedades del caos, son el patrullaje y la búsqueda de objetos. Se analizaron las trayectorias generadas para uno y múltiples robots móviles. Se reportan tanto resultados numéricos como experimentales. Además, se demuestra que las trayectorias resultantes de la sincronización entre los estados del mapa caótico de Hénon y las velocidades lineales y angulares del robot Khepera III son impredecibles. Por último, se presenta una aplicación a un cuadracóptero real.

Palabras Clave: *Caos, Robot móvil diferencial, Robot Khepera III, Trayectorias caóticas, Aplicación al patrullaje, Cuadracóptero.*

Dedicatoria

A mi bella compañera de vida y a mi hermosa hija, a quienes amaré por siempre.

Agradecimientos

A...

La Universidad Autónoma de Baja California, que me brindó la oportunidad de superarme e iniciar en el ámbito de la investigación y la docencia.

Al Centro de Investigación Científica y de Educación Superior de Ensenada, que me brindó de las herramientas y el apoyo para la elaboración de este trabajo, y que me permitió encontrar algo más a lo que aspirar.

A mi hermosa hija Ana Cecilia y a mi bella compañera de vida, futura esposa y madre de mi hija, Mariana Alejandra Chan Ley, por todos los momentos maravillosos que hemos vivido juntos. Su apoyo, paciencia y consejos han sido indispensables para la conclusión de este proyecto. Nuestra pequeña familia es mi principal motivación en la vida.

A mi familia, cuyo apoyo incondicional fue indispensable. En especial a mi madre Cecilia Noemí por haber estado ahí siempre que lo necesité y por darme las herramientas de vida necesarias para seguir adelante.

A mis directores de tesis, la Dra. Rosa Martha López Guitiérrez y el Dr. César Cruz Hernández, quienes me guiaron con paciencia a lo largo de este trabajo. Sus consejos y enseñanzas fueron invaluable y me inculcaron amor por el área y la investigación.

A los miembros de mi comité de tesis, Dr. Javier Pliego Jiménez, Dra. Liliana Cardoza Avendano y, en especial, al Dr. Adrian Arellano Delgado, por el tiempo, sugerencias y aportaciones que me permitieron realizar un trabajo de calidad.

A mis compañeros de estudio en la UABC, que recorrieron este arduo camino conmigo.

Al Consejo Nacional de Ciencia y Tecnología (CONACyT) a través del proyecto de investigación en Ciencia Básica entre instituciones “Sincronización de sistemas complejos y algunas aplicaciones”, Ref.A1-S-31628.

Al Consejo Nacional de Ciencia y Tecnología (CONACyT) por brindarme el apoyo económico necesario para realizar mis estudios de Doctorado.

Índice general

Índice de figuras	IX
-------------------	----

Índice de tablas	XIII
------------------	------

1. Introducción	1
1.1. Antecedentes	2
1.2. Motivación	4
1.3. Justificación	6
1.4. Hipótesis	6
1.5. Objetivos	7
1.5.1. Objetivo general	7
1.5.2. Objetivos específicos	7
1.6. Productividad	8
1.7. Estructura de la tesis	9
2. Sistemas Caóticos	10
2.1. Características del caos	10
2.1.1. Mapa caótico de Hénon	12
2.2. Fractales	13
2.2.1. Dimensión fraccionaria	15
2.3. Aplicaciones de la teoría del caos en la ingeniería	17
2.4. Conclusiones del capítulo	18
3. Robot Móvil Diferencial	19
3.1. Robótica	19
3.1.1. Tipos de robots	20

3.2. Robot móvil diferencial	23
3.3. Modelo matemático del robot diferencial	24
3.3.1. Simulación de movimientos básicos	25
3.4. Robot móvil Khepera III	25
3.5. Conclusiones del capítulo	27
4. Resultados	28
4.1. Algoritmo generador de trayectorias caóticas	28
4.2. Resultados numéricos	34
4.2.1. Simulaciones	34
4.2.2. Porcentaje de cobertura del área de trabajo	36
4.2.3. Prueba 0-1 Gottwald-Melbourne para determinación de caos de la trayectoria del robot móvil	39
4.2.4. Prueba de conteo de cajas ("Box-Counting")	41
4.2.5. Simulación: Aplicación a búsqueda y detección de objetos	48
4.3. Dos robots móviles en la misma área de trabajo	52
4.3.1. División horizontal del área de trabajo	53
4.3.2. División vertical del área de trabajo	56
4.3.3. División diagonal del área de trabajo	58
4.3.4. Comparativo	60
4.4. Tres robots móviles en la misma área de trabajo	60
4.5. n robots móviles en la misma área de trabajo	61
4.6. Presencia de caos en las trayectorias generadas para múltiples robots móviles	61
4.7. Diferentes tamaños de área de trabajo	64
4.8. Conclusiones del capítulo	66
5. Implementación experimental	67
5.1. Resultados experimentales	67
5.1.1. Porcentaje de cobertura experimental del área de trabajo	70
5.1.2. Prueba 0-1 Gottwald-Melbourne para resultados experimentales . . .	70
5.1.3. Prueba de conteo de cajas ("Box-Counting") para resultados experimen- tales	72
5.2. Aplicación del algoritmo a dos robots Khepera III	76
5.2.1. Porcentaje de cobertura	76

5.2.2. Conteo de Cajas	77
5.2.3. Prueba 0-1 Gottwald-Melbourne	77
5.3. Aplicación del algoritmo a un cuadrapterio	79
5.3.1. Cuadrapterio	79
5.3.2. CrazyFlie	81
5.3.3. Ambiente de programación	82
5.3.4. LOCO Positioning System	82
5.3.5. Metodología	84
5.3.6. Presencia de Caos en la trayectoria generada para un cuadrapterio . .	85
5.4. Conclusiones del capítulo	89
6. Conclusiones	91
6.1. Trabajo futuro	92
Bibliografía	94

Índice de figuras

1.1. Trayectorias predecibles vs. impredecibles.	5
2.1. Serie de tiempo del primer estado de dos circuitos de Chua con diferentes condiciones iniciales.	11
2.2. Atractor de Lorenz.	12
2.3. Atractor caótico del mapa de Hénon generado con 1000 iteraciones.	13
2.4. Fractal de Mandelbrot.	14
2.5. Fractal natural formado por la línea divisoria entre el mar y la porción de tierra en una isla, Días Brecia (2009).	14
2.6. Fractal sintético, Helecho de Barnsley.	15
2.7. Fractal generado por el método de Julia.	16
3.1. Robot tipo Ackerman.	21
3.2. Robot tipo triciclo.	22
3.3. Robot omnidireccional.	22
3.4. Vista superior de un robot móvil diferencial.	23
3.5. Movimiento básico de un robot móvil diferencial.	23
3.6. Movimientos básicos del robot diferencial.	26
3.7. Robot móvil diferencial Khepera III, Laboratorio de Sincronización y Sistemas Complejos.	27
4.1. Diagrama a bloques del algoritmo implementado para la generación de trayectorias caóticas para el robot móvil.	29
4.2. Histogramas de velocidades de las ruedas izquierda v_l y derecha v_r	30
4.3. Histogramas resultantes con dispersión de velocidades lineales de las ruedas izquierda v_{dl} y derecha v_{dr}	31

4.4. Trayectoria caótica del robot móvil generada por el algoritmo propuesto para $n = 20$	32
4.5. Delimitación del área de trabajo, mesa de 1400 <i>mm</i> de ancho por 2500 <i>mm</i> de largo. Laboratorio de Sincronización y Sistemas Complejos.	33
4.6. Estrategia reflexiva de evasión de bordes del robot móvil diferencial.	33
4.7. Trayectoria del robot móvil con las mismas condiciones iniciales $(x_0, y_0) = (0, 0)$ y diferente número de iteraciones.	35
4.8. Trayectorias generadas por el robot diferencial para diferentes condiciones iniciales con 200 iteraciones.	36
4.9. Porcentajes del área de trabajo cubiertas por el robot móvil diferencial para diferentes números iteraciones.	37
4.10. Porcentajes del área de trabajo cubiertas por el robot móvil diferencial para diferentes condiciones iniciales arbitrarias para 300 iteraciones.	38
4.11. Series de tiempo resultantes con condiciones iniciales $x_0 = 0, y_0 = 0$	40
4.12. Series de tiempo resultantes con condiciones iniciales $x_0 = 0.01, y_0 = 0$	40
4.13. Series de tiempo resultantes con condiciones iniciales $x_0 = -350, y_0 = 500$	41
4.14. Porcentajes de cobertura para diferentes tamaños de cuadro.	43
4.15. Resultado de la prueba de conteo de cajas para las condiciones iniciales $x_0 = 0, y_0 = 0$	45
4.16. Resultado de la prueba de de conteo de cajas para condiciones iniciales del robot móvil $x_0 = 0.01, y_0 = 0$	46
4.17. Resultado de la prueba de conteo de cajas para condiciones iniciales del robot móvil $x_0 = -350, y_0 = 500$	47
4.18. Resultados de búsqueda y detección de objetos para diferentes posiciones del objeto y condiciones iniciales del robot móvil: $x_0 = 0, y_0 = 0$	48
4.19. Resultado de búsqueda y detección de un objeto con ubicación fija ($x_{obj} = 0, y_{obj} = 0$) y condiciones iniciales arbitrarias del robot móvil diferencial en el área de trabajo.	50
4.20. Resultado de búsqueda y detección de objeto con ubicación arbitraria y condiciones iniciales arbitrarias del robot móvil diferencial en el área de trabajo.	51
4.21. Trayectorias generadas para dos robots móviles.	53
4.22. Trayectorias generadas con dos robots móviles diferenciales en el área de trabajo dividida horizontalmente.	54

4.23. Porcentajes del área de trabajo cubiertas por dos robots móviles diferenciales en el área de trabajo dividida horizontalmente.	55
4.24. Trayectorias generadas con dos robots móviles diferenciales en el área de trabajo dividida verticalmente.	56
4.25. Porcentajes del área de trabajo cubiertas por tres robots móviles diferenciales en el área de trabajo dividida verticalmente.	57
4.26. Trayectorias generadas con dos robots móviles diferenciales en el área de trabajo dividida diagonalmente.	58
4.27. Trayectorias generadas con dos robots móviles diferenciales en el área de trabajo dividida diagonalmente.	59
4.28. Trayectorias generadas para tres robots móviles.	61
4.29. Porcentajes de cobertura para diferentes tamaños de área de trabajo.	65
5.1. Software Motive ®.	68
5.2. Resultados experimentales: 3 diferentes trayectorias generadas por 3 diferentes condiciones iniciales del robot Khepera III.	69
5.3. Porcentajes del área de trabajo cubiertas por trayectorias del robot Khepera III generadas con diferentes condiciones iniciales.	70
5.4. Series de tiempo resultantes con condiciones iniciales del inciso a).	71
5.5. Series de tiempo resultantes con condiciones iniciales del inciso b).	71
5.6. Series de tiempo resultantes con condiciones iniciales del inciso c).	72
5.7. Resultado experimental de la prueba de conteo de cajas del inciso a).	73
5.8. Resultado experimental de la prueba de conteo de cajas del inciso b).	74
5.9. Resultado experimental de la prueba de conteo de cajas del inciso c).	75
5.10. Resultados experimentales para 2 robots móviles simultáneos.	76
5.11. Resultados experimentales de cobertura para 2 robots móviles simultáneos.	77
5.12. Dron.	79
5.13. Sentido de rotación de las hélices de un dron.	80
5.14. Ejes de movimiento de un dron.	80
5.15. Crazyflie.	81
5.16. LOCO Positioning System.	82
5.17. Ancla utilizada por LOCO Positioning System.	83
5.18. Etiqueta utilizada por LOCO Positioning System.	83

5.19. Atractor caótico en tres dimensiones del sistema MACM.	84
5.20. Trayectoria generada en un cuadracóptero Crazyflie.	86
5.21. Porcentaje de cobertura del área de trabajo por la trayectoria generada por el cuadracóptero Crazyflie.	87
5.22. Diferentes tamaños de cubo para el conteo de cajas de la trayectoria generada.	88

Índice de tablas

4.1. Porcentajes de cobertura del área de prueba con posiciones iniciales arbitrarias para 300 iteraciones.	39
4.2. Resultados numéricos de la prueba 0-1 de Gottwald-Melbourne.	41
4.3. Prueba de conteo de cajas para las condiciones iniciales del robot móvil $x_0 = 0$, $y_0 = 0$	44
4.4. Prueba de conteo de cajas para condiciones iniciales del robot móvil $x_0 = 0.01$, $y_0 = 0$	45
4.5. Prueba de conteo de cajas para condiciones iniciales del robot móvil $x_0 = -350$, $y_0 = 500$	47
4.6. Resultados de búsqueda y detección de un objeto en diferentes posiciones y condiciones iniciales del robot móvil: $x_0 = 0, y_0 = 0$	49
4.7. Resultados de búsqueda y detección de objeto con ubicación fija ($x_{obj} = 0, y_{obj} = 0$) y condiciones iniciales arbitrarias del robot móvil diferencial. . .	51
4.8. Resultados de búsqueda de objeto con ubicación y condiciones iniciales arbitrarias del robot móvil diferencial.	52
4.9. Porcentaje de cobertura del área de trabajo.	55
4.10. Porcentaje de cobertura del área de trabajo.	58
4.11. Porcentaje de cobertura del área de trabajo.	59
4.12. Porcentaje promedio de cobertura para n robots simultáneamente.	62
4.13. Resultados de la prueba Gottwald-Melbourne para 10 robots simultáneamente.	63
4.14. Resultado de Conteo de Cajas para n robots simultáneamente.	64
4.15. Porcentajes de cobertura para diferentes tamaños de área de trabajo (mm^2). . .	65
5.1. Resultados de la prueba 0-1 de Gottwald-Melbourne para resultados experimentales.	72

5.2. Prueba de conteo de cajas para experimento de las condiciones iniciales del inciso a).	73
5.3. Prueba de conteo de cajas para experimento de las condiciones iniciales del inciso b)).	74
5.4. Prueba de conteo de cajas para experimento de las condiciones iniciales del inciso c).	75
5.5. Prueba de conteo de cajas para el primer experimento.	78
5.6. Prueba de conteo de cajas para el segundo experimento.	78
5.7. Resultados de la prueba 0-1 para dos robots móviles.	79
5.8. Prueba de conteo de cajas para la trayectoria del Crazyflie.	87
5.9. Resultados de la prueba 0-1 de Gottwald-Melbourne para la trayectoria del Crazyflie.	89

Capítulo 1

Introducción

En este capítulo se proporciona una introducción al estudio de los sistemas caóticos, así como el estado del arte y los objetivos generales y particulares de este trabajo de tesis. Se pretende dar al lector un panorama general de trabajos realizados en el pasado que involucran las propiedades de los sistemas caóticos y su importancia, de manera que se comprenda el alcance y la relevancia de esta tesis.

El *caos* es un término que ha acompañado a la humanidad desde el principio de su existencia. En el pensamiento popular, se asocia al caos con una situación azarosa que, generalmente, deriva en infortunio. El *Diccionario de la Lengua Española* [1] define al caos como “estado amorfo e indefinido que se supone anterior a la ordenación del cosmos; también lo define como confusión, desorden”.

La llamada teoría de caos nace de la mano de matemáticos interesados en la vinculación entre sistemas dinámicos y su topología, como Jules Henri Poincaré y Stephen Smale; de físicos de campos tan dispares como la meteorología o la astronomía, como Edward Lorenz y Michel Hénon, respectivamente; de biólogos estudiosos del crecimiento de poblaciones, como Robert May.

El estudio de la teoría de caos rápidamente se ha arraigado en la sociedad científica en las últimas décadas, al grado tal que se han creado diversas revistas científicas dedicadas al tema. Es una rama de las matemáticas que involucra el estudio de los sistemas dinámicos, los cuales se definen como un conjunto de elementos interconectados entre sí y destinados a cumplir una tarea específica. Este conjunto de elementos describe un comportamiento cuya evolución depende del tiempo, ya sea de forma explícita o implícita y de forma continua o discreta.

En la ciencia se aprovechan las características caóticas de algunos sistemas para crear caos y aplicarlo, por ejemplo, a las comunicaciones seguras [2], [3], el diseño de antenas o la planeación de trayectorias [4]. Otros trabajos realizados en estas áreas se pueden ver en [5] y [6]. Por otra parte, el uso de vehículos autónomos no tripulados es de gran interés en la sociedad y la comunidad científica ha encaminado sus esfuerzos para que estos autómatas sean capaces de realizar eficientemente tareas como el patrullaje, transportación, búsqueda, rescate, limpieza, entre otras [7].

Por otra parte, la interacción entre la teoría del caos y la robótica móvil se estudió intensamente en la última década, debido a la riqueza de comportamientos dinámicos que caracterizan a los sistemas caóticos. Las dinámicas caóticas se pueden utilizar para guiar robots autónomos en exploración de terrenos, vigilancia, búsqueda, etc. La principal característica de los sistemas caóticos que se aprovecha para lograr estos objetivos eficientemente, es la sensibilidad a las condiciones iniciales, la cual permitirá al robot adoptar *trayectorias altamente impredecibles*.

En condiciones adversas, la impredecibilidad en la trayectoria de un robot es muy importante. Se puede diseñar una trayectoria caótica a un robot móvil para que éste siga una secuencia de puntos generados de manera aleatoria o de manera caótica. En ambos casos, entidades enemigas (en el caso de patrullaje) no podrían predecir la trayectoria futura, desconociendo las condiciones iniciales. Sin embargo, debido a su naturaleza determinista, el camino caótico representa una mejor opción. El motivo es que las entidades aliadas, conociendo las ecuaciones de estado y condiciones iniciales del sistema caótico, podrían predecir esta trayectoria y tomar las decisiones pertinentes para realizar la tarea que se requiera llevar a cabo.

1.1. Antecedentes

Los estudios de comportamientos de búsqueda de alimentos en grupos de animales (especialmente de las hormigas), son de gran ayuda en la solución de gran número de problemas abiertos, sobre todo de optimización. Recientemente, los biólogos han descubierto que las hormigas realizan actividades de búsqueda de manera caótica, ver por ejemplo [8], [9] y las referencias citadas dentro de este artículo.

Tratando de imitar este comportamiento caótico, se han realizado diversos trabajos en el diseño de trayectorias caóticas para robots móviles. En la literatura podemos encontrar diferentes soluciones al problema en cuestión. Nehmzow y Walker [10] estudiaron el comportamiento de los robots móviles, aplicando algoritmos de evasión de obstáculos y seguimiento de paredes y

observando cuantitativamente el movimiento de estos. Concluyeron que el comportamiento de estos robots es caótico.

Nakamura y Sekiguchi en [11] desarrollaron un método que integra las ecuaciones de movimiento del robot móvil y el sistema dinámico de Arnold para definir un sistema de control de lazo abierto utilizando variables de estado del sistema caótico. De manera similar, Bae, Lee y Gatton [12] integraron los sistemas caóticos de *Lorenz*, *Hamilton* e *Hipercaótico* con el modelo matemático del robot diferencial para generar movimientos caóticos.

También, se ha estudiado el caso donde la trayectoria del robot consiste en una serie de puntos generada caóticamente. Volos, Kyprianidis y Stouboulos [13], utilizando TRBG caótico (True Random Bit Generator) crearon puntos de seguimiento de manera caótica para ser seguidos por el robot y miden el porcentaje del terreno recorrido por sus sistemas dividiéndolo en celdas y midiendo la cantidad de veces que cada una es visitada, mientras que Filho y Macau [14] [15] utilizan el mapa estándar (también conocido como el mapa de Taylor-Chirikov) para generar los puntos. Filho [16] implementa los resultados obtenidos al utilizar el sistema de *Lorenz* en una simulación del robot móvil diferencial Khepera II.

Otro caso que se ha estudiado es cuando se requiere que el robot móvil siga, de manera caótica, la frontera de un área predeterminada para misiones de patrullaje [17]. En este caso, se utiliza el mapa de Hénon (adaptado utilizando transformaciones afines) para definir el movimiento caótico a través de la frontera.

De igual manera, se ha utilizado el mapa logístico como un generador de números aleatorios en [18] con el fin de definir la trayectoria del robot. En este caso, se mantiene la velocidad del robot constante, mientras que su dirección está determinada por el mapa logístico. Por otro lado, Lin y Huang [19] combina los sistemas caóticos con *colmenas de abejas artificiales* (artificial bee colonies) para el diseño de trayectorias en robots móviles.

Otros trabajos, como los de Pliego-Jiménez y Martínez-Clark [20] [21], investigan la emergencia de comportamientos caóticos en robots móviles.

1.2. Motivación

Una de las principales características de los sistemas caóticos (como veremos en el capítulo 2) es su impredecibilidad. Aplicar dicha propiedad a robots móviles brinda numerosos beneficios y posibles aplicaciones. Algunos ejemplos son su aplicación en trayectorias para el patrullaje de un área de trabajo y para la búsqueda de objetos.

Supongamos que se desea patrullar un área de trabajo rectangular (como sería el caso de una mesa o una habitación) en la que se encuentra un objeto de valor. Un robot móvil terrestre diferencial es utilizado para dicha tarea, cuidando el objeto de ser tomado por alguna persona externa (un ladrón, por ejemplo). Existen diversas maneras en las que un robot móvil podría moverse en dicha área de forma "predefinida", es decir, con movimientos predecibles previamente programados como las que podemos ver en las figuras 1.1a), 1.1b) y 1.1c). Cualquier persona que pretenda tomar el objeto valioso podría observar el comportamiento del robot vigilante y, sin mucho esfuerzo, predecir los movimientos de éste y lograr evadirlo con facilidad. En cambio, con una trayectoria impredecible (como la mostrada en la figura 1.1d)), sería prácticamente imposible para el ladrón llegar al objeto sin ser detectado por el robot móvil.

Supongamos ahora que se desconoce la ubicación del objeto dentro del área de trabajo (por ejemplo, podría tratarse de un explosivo, o de una persona atrapada en una habitación inaccesible para otros humanos). Si el robot móvil se desplaza en dicha área de trabajo con las trayectorias predecibles predeterminadas (1.1a) y 1.1b)), podría tardar mucho tiempo en encontrar lo que se está buscando, incluso podría no llegar a encontrarlo (1.1c)). En cambio, con una trayectoria impredecible (1.1d)) el robot lo encontraría en un tiempo menor (como se mostrará en el capítulo 4).

Ambos ejemplos podrían ser aplicados en diversos tipos de robots móviles, un robot submarino podría buscar objetos en el océano, y uno volador podría patrullar un área en tres dimensiones. Incluso podrían aplicarse a múltiples robots móviles de manera simultánea para incrementar el área de cobertura (en el caso del patrullaje) o para disminuir el tiempo necesario para localizar un objeto.

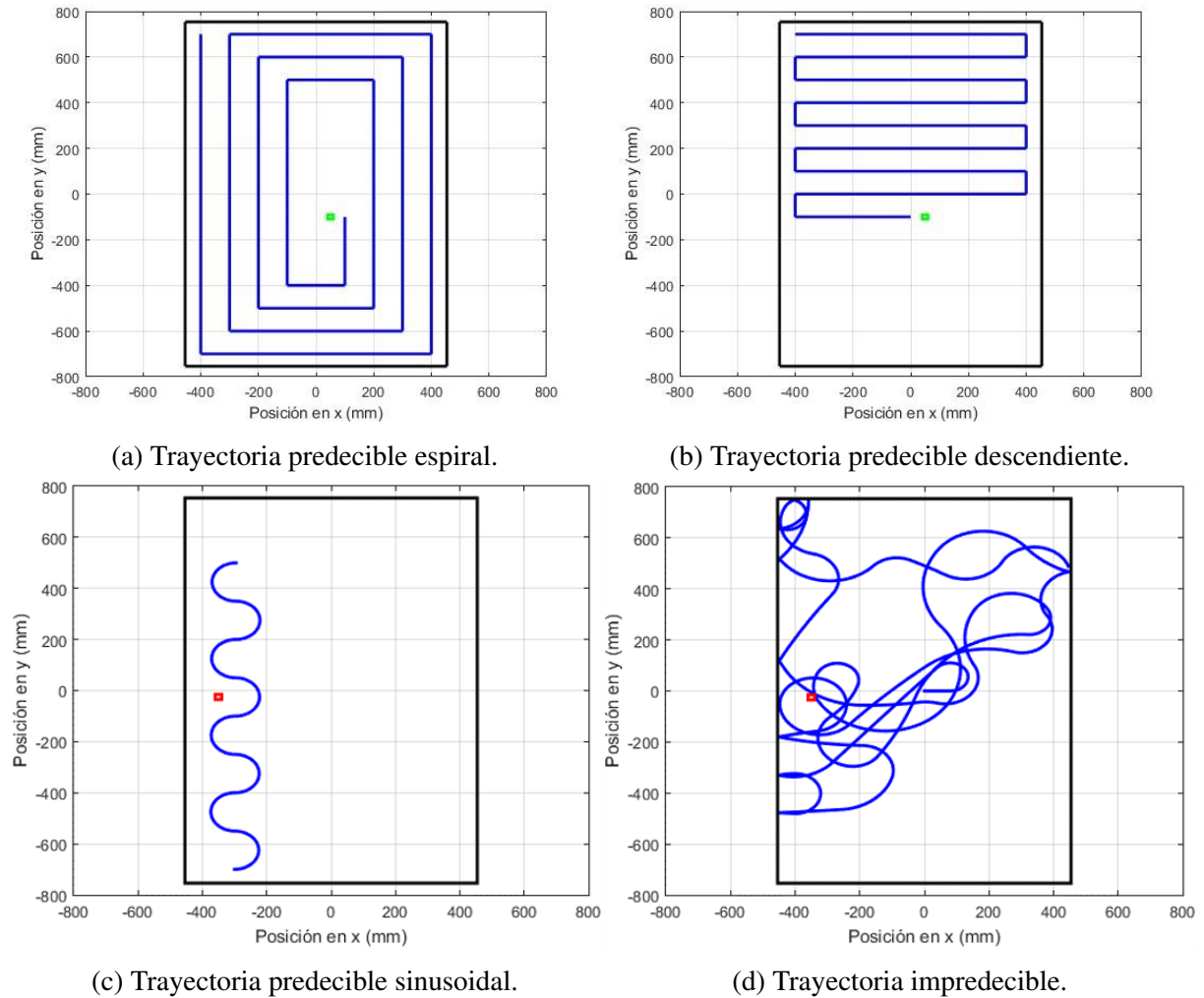


Figura 1.1 Trayectorias predecibles vs. impredecibles.

Para lograr estas trayectorias impredecibles (caóticas), algunos estudios recientes [22] [23] siguen el trabajo realizado por Nakamura [11], combinando el modelo matemático del robot móvil con las ecuaciones de estado de sistemas caóticos, como el de Lorenz y el de Chua. Otro trabajo [24] utiliza otro sistema caótico (mapa logístico) para generar puntos en el área de trabajo a través de los que el robot móvil debe pasar.

A diferencia de los trabajos previamente mencionados, el algoritmo propuesto en este artículo (en la sección 4) no requiere de la modificación del modelo matemático del robot móvil, ni la generación de puntos en el área de trabajo. En su lugar, las trayectorias caóticas emergen del comportamiento de las entradas del robot. En este caso en particular (usando un robot móvil

diferencial Khepera III), estas entradas son la velocidad deseada de cada una de las ruedas del robot. La principal ventaja de esta metodología es su sencillez, la cual permite aplicar el algoritmo a un robot móvil real.

La principal contribución de este trabajo es la manera de generar trayectorias caóticas en robots móviles; es decir, la aplicación del mapa de Hénon para asignar valores de velocidad a un robot móvil para generar una trayectoria caótica que le permita patrullar el área de trabajo de una manera eficaz e impredecible para observadores externos (por ejemplo, intrusos). A nuestro saber, no se han reportado metodologías similares en la literatura anteriormente.

1.3. Justificación

En este trabajo de tesis se generan trayectorias caóticas en robot móviles de manera numérica y experimental utilizando el mapa de Hénon para influir directamente en el comportamiento de las velocidades de los robots. A diferencia de otros trabajos, en el algoritmo aquí propuesto no se hace uso directo del modelo matemático del robot móvil, si no que las trayectorias emergen del comportamiento de cada una de las entradas al sistema; en el caso del robot móvil diferencial estas entradas son las velocidades de los motores (como se describe en el capítulo 3). Una de las ventajas de este método es su sencillez y la facilidad que tiene de ser exportado hacia otros tipos de robots móviles, ya que solamente es necesario conocer el número de entradas que tenga el robot y un sistema caótico que cuente con esa misma cantidad de estados. Por ejemplo, un cuadracóptero que cuenta con cuatro motores necesitaría un sistema caótico de cuatro estados para poder aplicar este algoritmo. A nuestro conocimiento, en base a una exhaustiva revisión de la literatura, no existe una metodología similar reportada en la literatura.

1.4. Hipótesis

Es posible generar trayectorias no predecibles con base en sistemas caóticos, para diferentes tipos robots móviles en aplicaciones prácticas de detección, búsqueda y rescate eficiente.

1.5. Objetivos

1.5.1. Objetivo general

Con la realización de la presente tesis de doctorado, se pretende alcanzar el siguiente objetivo general:

- Diseñar un algoritmo generador de trayectorias basadas en dinámicas caóticas para robots móviles con aplicaciones de exploración eficiente de terrenos inhóspitos, detección rápida de objetos peligrosos y rescate de víctimas en catástrofes.

1.5.2. Objetivos específicos

- Seleccionar el sistema caótico apropiado para el diseño de trayectorias en robots móviles.
- Realizar pruebas numéricas del comportamiento básico del modelo de robot móvil diferencial tipo Khepera III.
- Generar trayectorias basadas en dinámicas caóticas para robot móvil diferencial tipo Khepera III.
- Realizar aplicaciones en la exploración eficiente de terrenos inhóspitos, detección rápida de objetos peligrosos o rescate de víctimas en catástrofes.
- Generar trayectorias basadas en dinámicas caóticas para un cuadricoptero de manera experimental.

1.6. Productividad

Entre las contribuciones de este trabajo de tesis se encuentran las siguientes:

J. J. Cetina-Denis, A. Arellano-Delgado, A. López-Parra, R. M. López-Gutiérrez, C. Cruz-Hernández

Diseño de trayectorias caóticas en robots móviles.

Congreso Latinoamericano de Control Automático (CLCA 2016).

Medellín, Colombia. 12 - 15 de Octubre 2016. Pp 563-569.

J. J. Cetina-Denis, R. M. López-Gutiérrez, A. Arellano-Delgado, C. Cruz-Hernández

Diseño de trayectorias caóticas en robots móviles.

Congreso Mexicano de Robótica (COMROB) 2018.

Ensenada, Baja California, México.

J. J. Cetina-Denis, R. M. López-Gutiérrez, C. Cruz-Hernández, A. Arellano-Delgado

Design of a Chaotic Trajectory Generator Algorithm for Mobile Robots.

Applied Sciences, 12(5), 2587.

Además de la participación en el evento:

J. J. Cetina-Denis, A. López-Parra, W. Pecasso-Rubio, E. Cosío.

Impartición del taller: *Sincronización de robots móviles.*

INGENIUM - Sexto Simposium Internacional Regional de Ingeniería Industrial.

Instituto Tecnológico Superior de Puerto Peñasco, Sonora.

12 - 15 de abril 2016.

Impartición del Taller: “Simulación Virtual de Robots Móviles”.

Congreso Mexicano de Robótica (COMROB) 2018.

Ensenada, Baja California.

1.7. Estructura de la tesis

Este trabajo de tesis se organiza de la siguiente manera: En el capítulo 2 se discute acerca de los sistemas caóticos y sus propiedades, describiendo específicamente el mapa de Hénon y los fractales. En el capítulo 3 se describe el modelo matemático del robot diferencial, hablando particularmente de las características del robot Khepera III, el cual será utilizado en la etapa numérica y experimental de este trabajo. En el capítulo 4 se describen los experimentos realizados y se muestran los resultados numéricos obtenidos, demostrando la presencia de caos en las trayectorias generadas. En el capítulo 5 se muestran los experimentos y los resultados que se obtuvieron de manera experimental al aplicar el algoritmo diseñado en este trabajo de tesis al robot móvil diferencial Khepera III y a un cuadracóptero Crazyflie. Finalmente, en el capítulo 6 se mencionan las conclusiones y en el capítulo 7 el trabajo futuro.

Capítulo 2

Sistemas Caóticos

En este capítulo se presentan las propiedades más importantes que distinguen a los sistemas caóticos. En particular, se describe el mapa caótico de Hénon y las ecuaciones de estado que rigen su comportamiento dinámico. Se habla también de los fractales, dando una definición de los mismos y sus características.

Para el lector interesado en profundizar en el tópico de caos, se recomienda consultar, por ejemplo, las referencias [25], [26], [27], [28] y [29].

2.1. Características del caos

La teoría de caos puede definirse como el estudio cualitativo del comportamiento dinámico aperiódico mostrado por algunos sistemas deterministas no lineales. Puede observarse su manifestación en gran cantidad de sistemas naturales e ingenieriles.

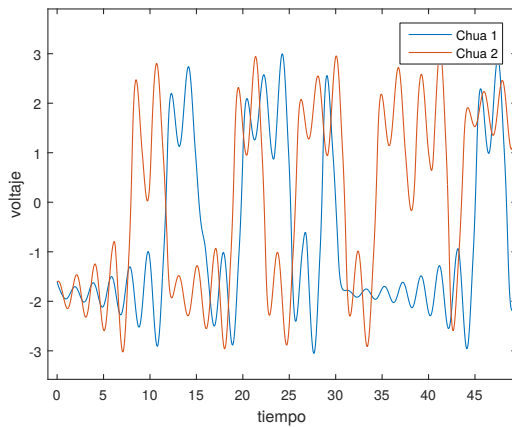
Los sistemas caóticos poseen características que los identifican, algunas de ellas son las siguientes:

- Presentan alta sensibilidad a las condiciones iniciales.
- Son deterministas, con comportamiento aparentemente aleatorio.
- Presentan transitividad topológica.
- Presentan comportamientos oscilatorios, pero no periódicos.

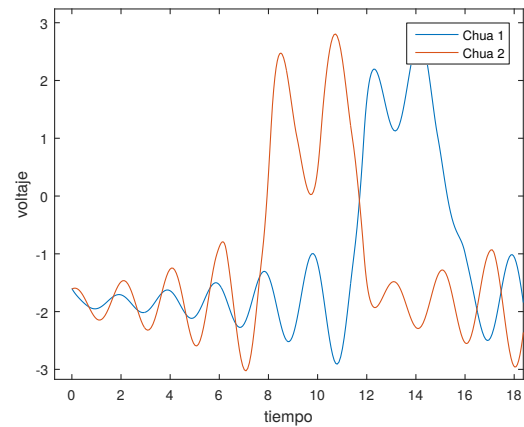
Alta sensibilidad a las condiciones iniciales implica que el mismo sistema caótico tendrá evoluciones temporales sumamente diferentes, a pesar de que éstas partan de condiciones iniciales ligeramente diferentes. Esta característica es altamente deseable en robots de patrullaje, ya que ocasionaría que la trayectoria generada caóticamente del mismo sea *altamente impredecible*.

Los sistemas caóticos son *deterministas* debido a que, conociendo las ecuaciones de estado, las condiciones iniciales del sistema y los valores de sus parámetros, es posible predecir su comportamiento.

La *transitividad topológica* implica la existencia de valores de entrada en un sistema que, dadas ciertas iteraciones, pueden moverse entre diferentes conjuntos abiertos. Es debido a esta última propiedad, que se puede garantizar que un robot móvil pueda recorrer un espacio cerrado de trabajo por completo. En la figura 2.1 se pueden apreciar estas propiedades en el sistema caótico de Chua, en especial la sensibilidad a las condiciones iniciales.



(a) Serie de 0 a 50 unidades de tiempo.



(b) Serie de 0 a 18 unidades de tiempo.

Figura 2.1 Serie de tiempo del primer estado de dos circuitos de Chua con diferentes condiciones iniciales.

Tomando en cuenta las características anteriormente mencionadas, podemos decir que un robot cuya trayectoria sea caótica será altamente impredecible para el observador externo, pero transparente para el diseñador y para cualquier observador autorizado que tenga conocimiento de las ecuaciones de estado del sistema, los valores de los parámetros y los valores iniciales utilizados para su diseño. También sería sensible a las condiciones iniciales, que en el caso del robot móvil serían las posiciones en x y y en las que este iniciaría su recorrido.

También, se puede identificar a un sistema caótico por la presencia de un atractor extraño, como el de la figura 2.2. De forma general, se puede decir que un atractor es una región o conjunto cerrado en el espacio de fase, al cual convergen las trayectorias de un sistema, entendiéndose por trayectoria la evolución temporal del sistema a partir de una condición inicial específica [30]. Por último, los sistemas caóticos poseen al menos un exponente de Lyapunov positivo.

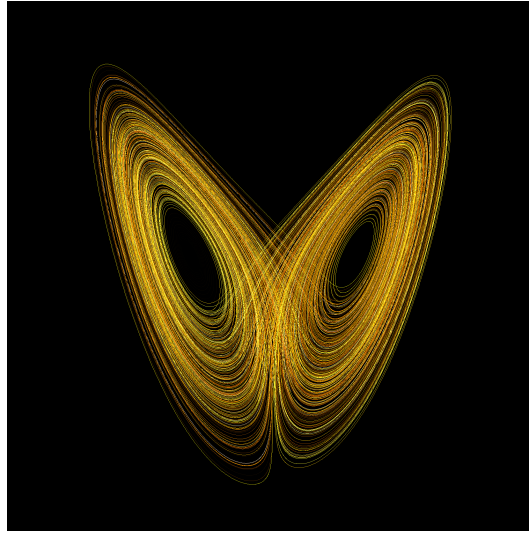


Figura 2.2 Atractor de Lorenz.

2.1.1. Mapa caótico de Hénon

El mapa de Hénon es un sistema dinámico en tiempo discreto. Es uno de los ejemplos de sistemas dinámicos más estudiado que muestra comportamiento caótico. El mapa de Hénon toma un punto $(x_h(n), y_h(n))$ en el plano y lo mapea a un nuevo punto. Está representado por las siguientes ecuaciones de estado no lineales en diferencias [31]:

$$\begin{aligned} x_h(n+1) &= 1 - ax_h^2(n) + y_h(n+1), \\ y_h(n+1) &= bx_h(n). \end{aligned} \tag{2.1}$$

El mapa depende de dos parámetros a y b , los cuales, para el mapa clásico de Hénon tiene valores de $a = 1.4$ y $b = 0.3$. Para los valores clásicos, el mapa de Hénon (2.1) es caótico. Para otros valores de a y b el mapa puede presentar otras dinámicas, por ejemplo, converger a un

ciclo límite o presentar un punto fijo. Una visión general del tipo de comportamiento del mapa de Hénon en valores de parámetros diferentes puede obtenerse de su diagrama de fase, ver figura 2.3.

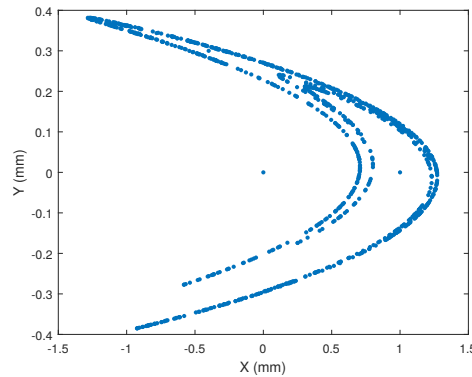


Figura 2.3 Atractor caótico del mapa de Hénon generado con 1000 iteraciones.

2.2. Fractales

Siempre que un sistema manifiesta dinámica caótica, ésta aparece asociada con un tipo de objetos geométricos caracterizados por su dimensión no entera, los objetos fractales. El término *fractal* fue acuñado por el matemático francés de origen polaco Benoit Mandelbrot [32]. En sus propias palabras: “Acuñé la palabra *fractal* del latín *fractus*, que proviene del verbo *frangere*, quebrar: crear fragmentos irregulares. Es por esto especialmente adecuado, puesto que además de rotura, *fractus* alude a irregular, siendo esta una característica presente en cada fragmento”. En la figura 2.4 se observa el famoso fractal generado por Mandelbrot.

La *Real Academia Española* define a los fractales como “Figura plana o espacial, compuesta de infinitos elementos, que tiene como propiedad que su aspecto y distribución estadística no cambian cualquiera que sea la escala con que se observé”. Los llamados *objetos fractales* son pues, figuras geométricas sumamente complejas y detalladas. Si amplificamos una sección de las mismas nos encontramos con tanta complejidad y detalle como en la situación anterior, de tal manera que las secciones más pequeñas son similares a las grandes [32].

Son muchas las entidades que presentan esta estructura, algunas son las galaxias, montañas, al igual que los bosques, también las líneas divisorias entre países y las plumas de las aves [33]. A manera de ejemplo, a través de los fractales se puede estudiar la forma de las nubes,

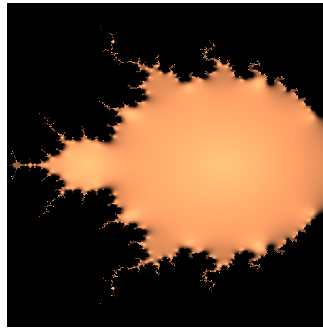


Figura 2.4 Fractal de Mandelbrot.

o ¿es qué son círculos perfectos? cuando nos acercamos visualmente a una nube, observamos que está formada por muchos fragmentos de nubes y estos fragmentos a su vez en otros y así sucesivamente [34]. Se puede considerar también, la línea que se forma en las costas, en donde se une la tierra con el mar, en gran escala se puede observar que la línea divisoria está formada por penínsulas (entradas de tierra en el mar) y ensenadas (partes del mar que entran en la tierra) y a medida que la escala se hace más pequeña se observa que cada península y ensenada están formadas por penínsulas y ensenadas más pequeñas y, si se reduce aún más la escala, se observa el mismo fenómeno, ver figura 2.5.



Figura 2.5 Fractal natural formado por la línea divisoria entre el mar y la porción de tierra en una isla, Días Brexia (2009).

Los fractales tienen tres propiedades definitorias [32]:

- *Autosimilaridad*. Sus partes tienen la misma forma o estructura que el todo.

- *Autorreferencia.* Determina que el propio objeto aparece en la definición de sí mismo.
- *Dimensión fraccionaria.*

La primera propiedad se manifiesta en que las sucesivas ampliaciones del objeto fractal de cualquier detalle de la misma son indistinguibles de los originales, es decir, son semejantes. La segunda propiedad implica que la forma de generar un fractal es mediante un algoritmo recurrente o regla de construcción. Por último, la dimensión fraccionaria alude al hecho de que los objetos fractales se encuentran en un espacio geométrico de dimensión no entera [32].

Es posible encontrar objetos fractales en la naturaleza, por ejemplo, las hojas de los helechos, el brócoli, copos de nieve, la corteza de los árboles y la coliflor, o el caso del helecho sintético creado por computadora de Barnsley (figura 2.6). Además de esto, es posible generarlos mediante algoritmos iterativos, como los generados por medio del método de Julia (figura 2.7), los cuales se generan mediante un algoritmo computacional.

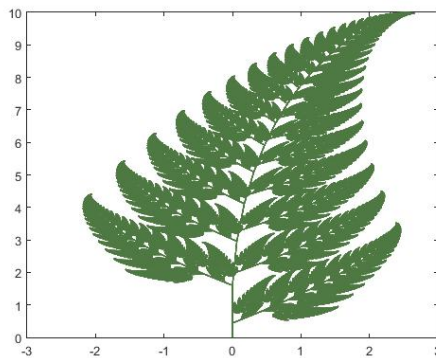


Figura 2.6 Fractal sintético, Helecho de Barnsley.

2.2.1. Dimensión fraccionaria

Desde un punto de vista topológico sabemos que la circunferencia y un segmento rectilíneo son la misma curva y encierran el mismo tipo de superficie (pues es posible transformar una en la otra mediante una deformación continua, es decir, sin que sea preciso someter a ninguna de las dos a manipulaciones “no topológicas”).

Ahora bien, desde un punto de vista métrico no son la misma curva, ya que la circunferencia y el área que encierra el círculo, son finitos y en cambio, el segmento, aunque es finito, no encierra con su borde un área finita.

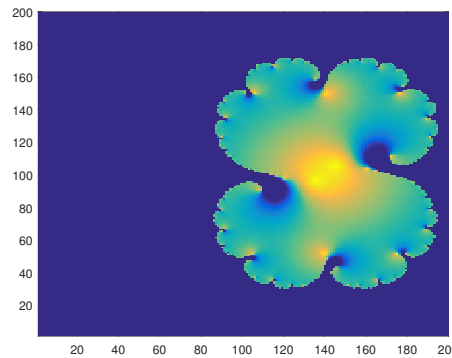


Figura 2.7 Fractal generado por el método de Julia.

Analicemos brevemente lo que significa la dimensión topológica, que es un término que introdujo Henri Poincaré para discernir sobre cuestiones de este tipo. La definición inductiva dada por Poincaré al introducir este concepto fue la siguiente:

- El conjunto vacío tiene dimensión -1 .
- Si los bordes de los entornos pequeños de todos los puntos del ente son espacios $(n-1)$ -dimensionales, decimos que el espacio que consideramos es n -dimensional

Así, según Poincaré, se tiene:

- Conjunto vacío: Dimensión topológica: $D=-1$.
- Punto: Dimensión topológica: $D=0$.
- Segmento de línea: Dimensión topológica: $D=1$.
- Cuadrado: Dimensión topológica: $D=2$.
- Cubo: Dimensión topológica: $D=3$.
- Hipercubo: Dimensión topológica: $D=4$.

Una definición distinta de dimensión topológica es la definición por semejanza, llamada también de *autosemejanza*, que sugirió Felix Hausdorff en 1919 [35], readaptada posteriormente por Besicovich (dimensión de Hausdorff-Besicovich) [36]:

Si al obtener desde un ente H , N entes iguales, semejantes al original, con razón de semejanza l , entonces la dimensión topológica de H es el número real D que verifica,

$$N + l^D = 1. \quad (2.2)$$

Es decir,

$$D = \frac{\log(N)}{\log \frac{1}{l}}. \quad (2.3)$$

La dimensión topológica en el sentido de Poincaré coincide en general con la dimensión por semejanza de Hausdorff-Besicovich. Pero hay ciertos objetos geométricos en los que no ocurre así. A estos objetos geométricos los denominaremos, usando la terminología de Benoit Mandelbrot, fractales. Diremos que la dimensión definida por Poincaré o Devlin es su dimensión topológica, la cual siempre es un entero y que la dimensión por semejanza de Hausdorff-Besicovich es su *Dimensión Fraccionaria*, la cual será utilizada en capítulos posteriores de este trabajo de tesis.

2.3. Aplicaciones de la teoría del caos en la ingeniería

En esta última sección se mencionan algunas aplicaciones del caos en la ingeniería. El lector interesado puede revisar la fuente bibliográfica [30].

En los casos presentados hasta este momento, el papel de la teoría del caos es el de explicar un fenómeno que se presenta naturalmente, el cual no puede ser entendido por técnicas convencionales. La mayoría de estas aplicaciones involucran el análisis de series de datos obtenidos en los experimentos para buscar por ejemplo, un atractor extraño o calcular la dimensión fraccionaria y los exponentes de Lyapunov positivos.

En el caso de la ingeniería, se aprovechan las características caóticas de algunos sistemas para crear caos y aplicarlo por ejemplo a las comunicaciones seguras, el diseño de antenas o la planeación de trayectorias. A continuación se presentan algunos casos en los que se aplica la teoría del caos en sistemas de ingeniería.

- **Diseño de radares.** La baja predictibilidad, característica de los sistemas caóticos, es deseable en el diseño de antenas. La teoría del caos se utiliza para procesar la señal de regreso, generar códigos binarios, implementar sistemas codificados e implementar

radares de señal ruidosa (NSR por sus siglas en inglés). Ejemplo de esto se puede encontrar en [37] y [38].

- **Control de movimiento en robots móviles.** Los sistemas caóticos, aplicados a la robótica móvil, son utilizados para guiar robots autónomos para exploración de terrenos, vigilancia, búsqueda y desactivación de bombas. Existen trabajos que reportan todo tipo de resultados, tanto numéricos como experimentales, empleando diversas metodologías. Algunos ejemplos se pueden encontrar en [13] y [39].
- **Sistemas de comunicaciones seguras.** En este caso, la sensibilidad a las condiciones iniciales y la “semejanza al ruido” de los sistemas caóticos permite el cifrado de mensajes para su envío por canales públicos. La ventaja principal es la baja probabilidad de decodificación del mensaje en caso de interceptación. Ejemplos de estas aplicaciones las podemos encontrar en [2], [3], [5], [40], [41] y [42].
- **Diseño de antenas fractales.** La evolución de los sistemas de comunicación obliga a los dispositivos manejar antenas multibanda. La limitante en tamaño de antenas, dictada por los dispositivos móviles, han obligado a los científicos a cambiar el enfoque en el diseño de éstas. Una de las soluciones que mejor desempeño presentan es el uso de antenas con patrones fractales. En [43] podemos encontrar un ejemplo.

2.4. Conclusiones del capítulo

Este capítulo se concentró en describir las principales propiedades del caos, como la sensibilidad a las condiciones iniciales y su naturaleza determinista, así como diferentes aplicaciones del caos a diversas ramas de la ciencia y de la ingeniería. En particular, se describieron las ecuaciones de estado en diferencias del mapa caótico de Hénon y sus propiedades básicas. También se describieron los fractales y sus principales características.

Capítulo 3

Robot Móvil Diferencial

En este capítulo se describe de manera general el robot móvil diferencial, mencionando sus características y las ecuaciones matemáticas que describen su comportamiento dinámico. Se habla en particular del robot móvil diferencial Khepera III, el cual será utilizado para la validación experimental de este trabajo.

3.1. Róbotica

En las últimas décadas, el concepto de robot ha pasado de ser una visión de ciencia ficción como dispositivo mecánico super-humano a ser una realidad como máquina autónoma sorprendentemente animada con un gran potencial de aplicaciones en un sin fin de actividades cotidianas.

El buen entendimiento y el desarrollo de aplicaciones de robótica está condicionado al dominio de diversas áreas del conocimiento. Las disciplinas a las que normalmente se orientan los cursos de robótica en las universidades incluyen las Ingenierías Eléctrica, Electrónica, Mecánica, Industrial, las Ciencias Computacionales, la Mecatrónica y la Cibernética [44].

Los robots han sido objeto de estudio desde hace mucho tiempo. Se han empleado en todo tipo de aplicaciones. Desde el primer uso de la palabra robot en [45], estos han sufrido una evolución que les ha permitido diversificarse en varios tipos diferentes, dependiendo de su uso y aplicación.

3.1.1. Tipos de robots

Existen diversas maneras de clasificar a los robots, una de ellas es por su movimiento:

1. **Fijos.-** La mayoría de los robots fijos son robots industriales que trabajan en ambientes bien definidos. Se encuentran comúnmente montados en una base estable en el suelo y, según su configuración interna, pueden calcular su posición. Algunas aplicaciones de estos robots pueden observarse en áreas como la medicina [46], la industria [47] y la educación [48].
2. **Móviles.-** Como sugiere el nombre, los robots móviles no están fijados a una superficie, sino que pueden moverse libremente sobre la superficie.
 - a) **Robots Submarinos.-** Estos robots son la última incorporación a la familia de robots, funcionan bajo el agua y se utilizan para navegar, asistir en la natación, etc. Un ejemplo práctico incluye el Vehículo Subacuático Autónomo (AUV), PacX Wave Glider [49] (que viajó una distancia de aproximadamente 17000 km sobre el Océano Pacífico para recopilar datos oceánicos de alta resolución) y el robot Swumanoid [50] (que está siendo construido por investigadores japoneses para ayudar a los atletas a nadar más rápido)
 - b) **Robots Aéreos.-** Estos son en su mayoría robots de tamaño micro que imitan la morfología de un insecto. Algunos también están diseñados como vehículos aéreos no tripulados (UAV). Estos drones pueden variar desde un pequeño helicóptero de juguete controlado por radio hasta un Global Hawk de 32000 libras. Entre las principales aplicaciones de los UAV podemos encontrar la agricultura [51], Ingeniería de costas [52] y manejo de desastres [53].
 - c) **Robots Terrestres.-** Pueden ser un robot con patas, oruga o con ruedas.
 - 1) **Con patas.-** Estos son generalmente los robots que caminan que usan extremidades para moverse. Estos se utilizan principalmente para proporcionar movimiento en entornos muy des estructurados. Aunque su diseño es complejo, estos robots tienen una mayor ventaja sobre los robots con ruedas en términos de navegación en cualquier tipo de camino o terreno. Algunos ejemplos de sus aplicaciones podemos encontrarlos en [54].

- 2) **Oruga.-** Es normalmente utilizado para explorar  reas de dif cil acceso, su peculiar estructura le permite moverse por terrenos irregulares. Para un movimiento en l nea recta es muy eficiente, pero presenta una desventaja al realizar giros ya que su eficiencia es muy baja. Algunos ejemplos de sus aplicaciones se pueden ver en [55], [56] y [57]
- 3) **Con ruedas.-** Al igual que los veh culos, los robots con ruedas son guiados por modelos cinem ticos que se basan en el movimiento en lugar de la causa del movimiento. La versi n m s primitiva fue el Veh culo Guiado Autom tico (VGA), un veh culo el ctrico de 4 ruedas.

Tipos de robots m viles con ruedas

Dentro de los robots m viles con ruedas existen diferentes configuraciones cinem ticas las cuales le confieren al robot diferentes caracter sticas y propiedades en su maniobrabilidad y dependen del prop sito final del robot. Las siguientes configuraciones cinem ticas son las m s utilizadas, clasificadas por su tipo de tracci n:

1. **Ackerman.-** Esta configuraci n es la utilizada en los veh culos de cuatro ruedas convencionales. Las dos ruedas delanteras giran para controlar la orientaci n y las traseras se mantienen paralelas. Cuando se efect a un giro, la rueda interior gira en un  ngulo mayor que la exterior para evitar el deslizamiento (figura 3.1). Un ejemplo de aplicaci n de este tipo de robot m vil en la agricultura puede consultarse en [58]. Algunos trabajos de investigaci n relacionados pueden verse en [59] y [60].

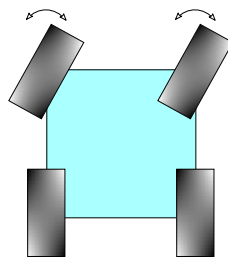


Figura 3.1 Robot tipo Ackerman.

2. **Triciclo.-** Esta configuraci n cuenta con una rueda delantera encargada de la tracci n y direcci n del robot y 2 ruedas traseras que se mueven libremente encargadas de proporcionar la estabilidad (figura 3.2). Esta configuraci n ofrece una alta maniobrabilidad, sin

embargo, presenta ciertos problemas de estabilidad y tracci n en terrenos accidentados por lo que es utilizado principalmente en robots para interiores. Algunas de sus aplicaciones se encuentran en la industria [61] [62] y monitoreo de casas verdes [63], entre otros [64].

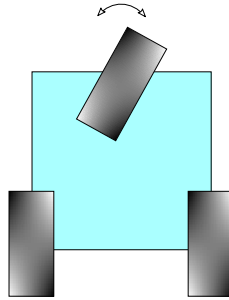


Figura 3.2 Robot tipo triciclo.

3. **Omnidireccional.-** Un robot omnidireccional, tambi n conocido como holon mico, es un tipo de robot m vil con ruedas, cuya configuraci n le permite desplazarse en cualquier direcci n sin la necesidad de alcanzar previamente una orientaci n espec fica (figura 3.3). Algunas de sus aplicaciones se pueden encontrar en [65] y [66].

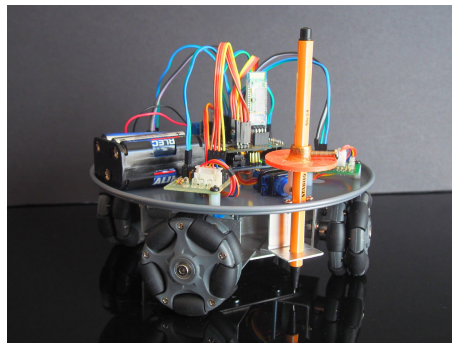


Figura 3.3 Robot omnidireccional.

4. **Diferencial.-** El robot de ruedas con tracci n diferencial realiza su direccionamiento con la diferencia de velocidades de dos ruedas instaladas en un  nico eje, de esta manera se puede conseguir que avance en l nea recta cuando los motores est n a la misma velocidad y que cuando se apliquen velocidades diferentes en los motores, este logre hacer un giro en una u otra direcci n, adem s de girar en su propio eje cuando se aplican las mismas velocidades, pero en sentido opuesto.

3.2. Robot móvil diferencial

Un robot móvil diferencial es aquel cuyo desplazamiento está basado en dos ruedas de tracción independientes colocadas en ambos lados del cuerpo del robot, ver figura 3.4. El movimiento se consigue aplicando velocidades a cada una de las ruedas motrices. De esta forma, podemos hacer que el robot avance en línea recta fijando ambos motores a la misma velocidad, puede girar en una u otra dirección aplicando velocidades diferentes, y haciéndolas girar en sentido inverso una rueda respecto a la otra, el robot girará sobre su eje (figura 3.5).

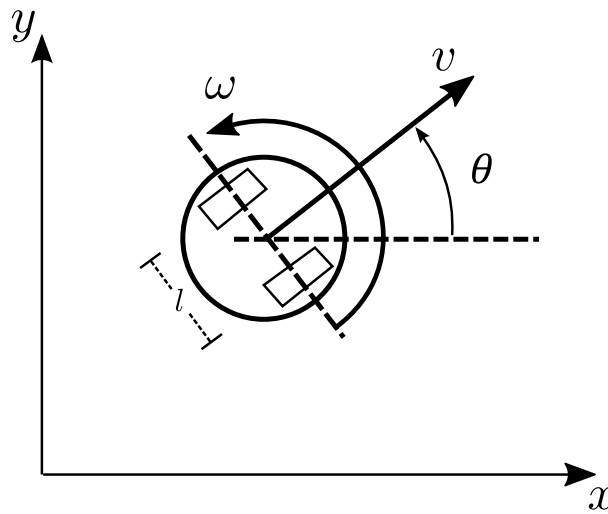


Figura 3.4 Vista superior de un robot móvil diferencial.

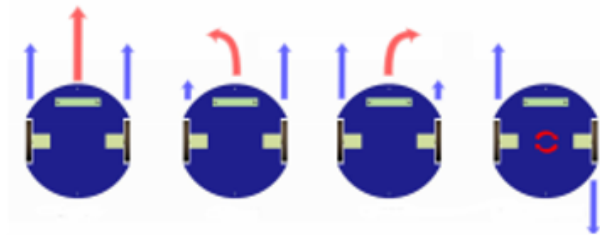


Figura 3.5 Movimiento básico de un robot móvil diferencial.

3.3. Modelo matemático del robot diferencial

En el trabajo [67] se reporta un modelo matemático para un robot móvil diferencial, tomando en cuenta las siguientes consideraciones, ver figura 3.4:

- El robot se mueve en una superficie perfectamente plana sin deslizamiento, despreciando la resistencia de las llantas.
- La posición del robot está determinada por las coordenadas x , y y un ángulo θ , el cual representa la rotación del robot en relación al sistema de coordenadas.

La posición y la rotación del robot en el plano están basadas en las siguientes ecuaciones, formando el modelo cinemático del robot [68]:

$$\begin{aligned}\dot{x} &= v \cos(\theta), \\ \dot{y} &= v \sin(\theta), \\ \dot{\theta} &= \omega.\end{aligned}\tag{3.1}$$

con v y ω definidas como:

$$\begin{aligned}v &= \frac{v_l + v_r}{2}, \\ \omega &= \frac{v_r - v_l}{l},\end{aligned}\tag{3.2}$$

donde:

v : Velocidad lineal del robot móvil.

ω : Velocidad angular del robot móvil.

θ : Ángulo de orientación del robot móvil.

l : Distancia entre las ruedas.

v_r : Velocidad de la rueda derecha.

v_l : Velocidad de la rueda izquierda.

Despejando v_l y v_r de la ecuación (3.2) obtenemos:

$$\begin{aligned}v_l &= v - (l * \omega), \\v_r &= v + (l * \omega).\end{aligned}\tag{3.3}$$

Estas ecuaciones (3.3) serán útiles para realizar las simulaciones numéricas reportadas en secciones y capítulos posteriores de la tesis.

3.3.1. Simulación de movimientos básicos

Utilizando el programa Matlab se realizaron simulaciones del modelo matemático (con las ecuaciones (3.1) y (3.3)) del robot móvil diferencial para tres casos:

- a) $v_l = 20 \text{ mm/s}$, $v_r = 10 \text{ mm/s}$ y $l = 88.41 \text{ mm}$.
- b) $v_l = 10 \text{ mm/s}$, $v_r = 20 \text{ mm/s}$ y $l = 88.41 \text{ mm}$.
- c) $v_l = 20 \text{ mm/s}$, $v_r = 20 \text{ mm/s}$ y $l = 88.41 \text{ mm}$.

Estas velocidades se ingresan al modelo matemático mostrado en la ecuación (3.1) por medio de (3.2). En la figura 3.6 se muestran los resultados de estas simulaciones numéricas. Se puede apreciar que el robot móvil diferencial (ecuaciones (3.1)-(3.3)), se comporta según lo descrito al inicio de la sección 3.3.

3.4. Robot móvil Khepera III

El Khepera III (figura 3.7) es un robot circular y pequeño que tiene dos ruedas y un soporte deslizante. El diámetro es de aproximadamente 130 mm, una altura de 70 mm y un peso de 650 gramos. Cuenta con 11 sensores infrarrojos (9 alrededor y 2 por debajo) y 5 ultrasónicos que le permiten detectar objetos y obstáculos que se encuentran alrededor de él. Las ruedas cuentan con encoders incrementales. Permite la comunicación vía Serial, USB y Bluetooth.

Por medio de cualquiera de sus opciones de comunicación es posible conectarse al robot móvil diferencial con una computadora para obtener los valores de los sensores. Obteniendo las mediciones de los encoders podemos estimar la posición del robot Khepera III. Sin embargo, este método obliga a depender de una computadora para realizar las operaciones necesarias para la localización del robot.

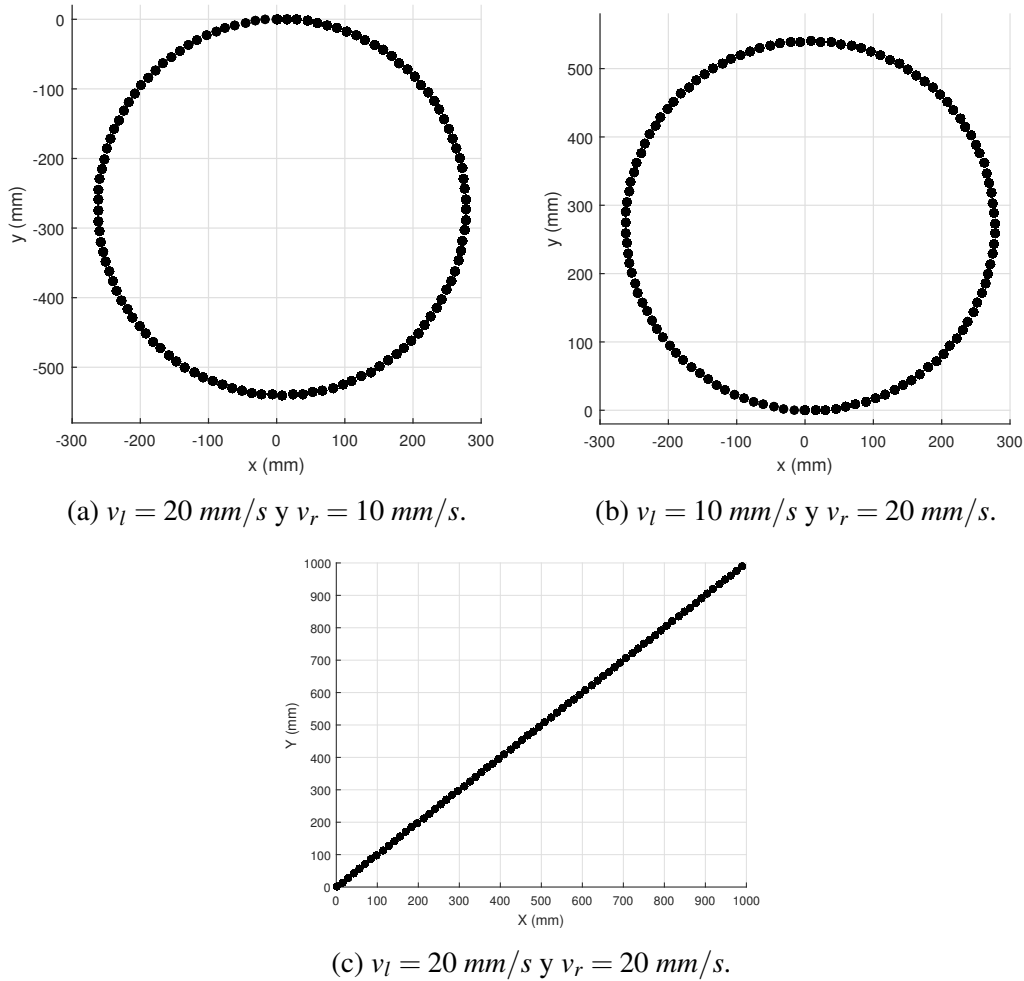


Figura 3.6 Movimientos básicos del robot diferencial.

El Khepera III cuenta con diferentes accesorios, el principal de ellos recibe el nombre de KoreBot II. Este dispositivo provee al robot Khepera III con un sistema Linux embebido que le permite utilizar todos los demás accesorios y también, ser programado en diversos lenguajes, entre ellos C, C++, Lua, etc. y le otorga además la capacidad de comunicación vía Wi-fi. Entre los demás accesorios existentes para el robot Khepera III se encuentran los Grippers, cámaras y sensores láser.

Entre las características principales del robot móvil Khepera III se encuentran las siguientes:

- Diámetro de las ruedas: 41 mm.
- Distancia entre las ruedas: $l = 88.41 \text{ mm}$.



Figura 3.7 Robot móvil diferencial Khepera III, Laboratorio de Sincronización y Sistemas Complejos.

- Encoder: 4191 mediciones por revolución (0.001499 *radianes* por medición).
- Rango de velocidades de las ruedas: $[-200$ a $-20]$ y $[20$ a $200]$ mm/s.

Este robot es ampliamente utilizado en diferentes trabajos de investigación que involucran el uso de robots móviles de tipo diferencial, ver por ejemplo [13] y [16].

3.5. Conclusiones del capítulo

En este capítulo se describió de manera general el modelo matemático del robot móvil diferencial y algunas de sus características, mostrando sus ecuaciones de movimiento y la manera en las que sus entradas (velocidades de las ruedas) afectan su comportamiento realizando algunas simulaciones básicas de movimiento. Este modelo será empleado en capítulos posteriores para el diseño, simulación numérica y experimentación del algoritmo generador de trayectorias caóticas para un robot móvil diferencial.

Capítulo 4

Resultados

En este capítulo se describe el proceso realizado para llevar a cabo la generación de trayectorias no predecibles con base en sistemas caóticos para robots móviles diferenciales. Se describe el algoritmo propuesto y las consideraciones que se tomaron para elaborarlo. Se muestran resultados numéricos de algunas aplicaciones, así como un análisis de éstos para poder determinar si las trayectorias generadas en los robots son efectivamente caóticas y si presentan las propiedades deseadas del caos en ellas.

4.1. Algoritmo generador de trayectorias caóticas

Para lograr la generación de trayectorias caóticas en el robot móvil diferencial, en esta tesis se propone el diseño del siguiente algoritmo (ver figura 4.1) tanto para las simulaciones numéricas como para el proceso experimental.

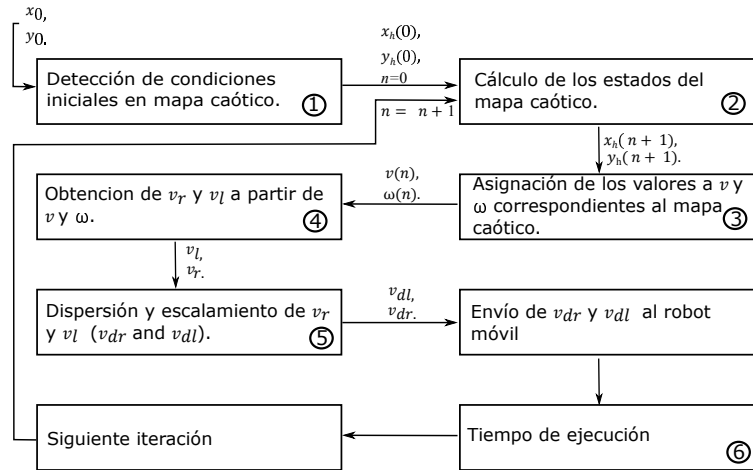


Figura 4.1 Diagrama a bloques del algoritmo implementado para la generación de trayectorias caóticas para el robot móvil.

Las ecuaciones correspondientes a las etapas del diagrama de bloques de la figura 4.1 son las siguientes:

Algorithm 1 Algoritmo Propuesto

Input: (x_0, y_0) = Posición inicial (x, y) del robot en el área de trabajo, $steps$ = iteraciones

Output: v_{dl}, v_{dr} = Velocidades de las ruedas izquierda y derecha del robot móvil

1: $x_h(0) = x(0)/1000$,

$y_h(0) = y(0)/1000$,

$n=0$;

While $1 \leq n \leq steps$ **do**

2: $x_h(n+1) = 1 - ax_h^2(n) + y_h(n)$

$y_h(n+1) = bx_h(n)$;

3: $v(n) = x_h(n+1)$

$\omega(n) = y_h(n+1)$;

4: $v_l = v(n) - l\omega(n)$

$v_r = v(n) + l\omega(n)$;

5: $v_{dl} = \text{floor}(((100v_l) - \text{floor}(100v_l)) * 180) + 20$

$v_{dr} = \text{floor}(((100v_r) - \text{floor}(100v_r)) * 180) + 20$;

6: $t = 2$ segundos;

7: $n = n + 1$.

end

return v_{dl}, v_{dr}

El proceso seguido por el algoritmo generador de trayectorias caóticas propuesto es el siguiente: En el paso 1 se toman las condiciones iniciales del robot móvil diferencial en el plano x - y y las utiliza como condiciones iniciales para el mapa de Hénon (ecuación (2.1)). El motivo de la división por 1000 es dar mayor énfasis en la sensibilidad a las condiciones iniciales. Por ejemplo, si el robot comienza en la posición $(x_0, y_0) = (1, 1) \text{ mm}$, las condiciones iniciales del mapa caótico de Hénon serán $x_h(0) = 0.001$ y $y_h(0) = 0.001$.

Con las condiciones iniciales obtenidas, en el paso 2 se calculan los valores actuales de los estados del mapa caótico de Hénon (iteración n), los cuales se asignan a las velocidades lineal y angular en el paso 3. De esta manera, los estados del mapa de Hénon influyen en su totalidad en el comportamiento de las velocidades del robot móvil diferencial.

Posteriormente, en el paso 4, se calculan las velocidades de las ruedas izquierda v_l y derecha v_r del robot. Es en este paso donde se obtienen los histogramas mostrados en la figura 4.2. En el paso 5 se realiza la dispersión de los datos de manera homogénea obteniendo los histogramas mostrados en la figura 4.3. Esta dispersión nos permite realizar un escalamiento de los datos para ajustar las velocidades obtenidas a los valores máximos y mínimos soportados por el robot móvil diferencial. A estas velocidades dispersas las llamaremos v_{dl} y v_{dr} .

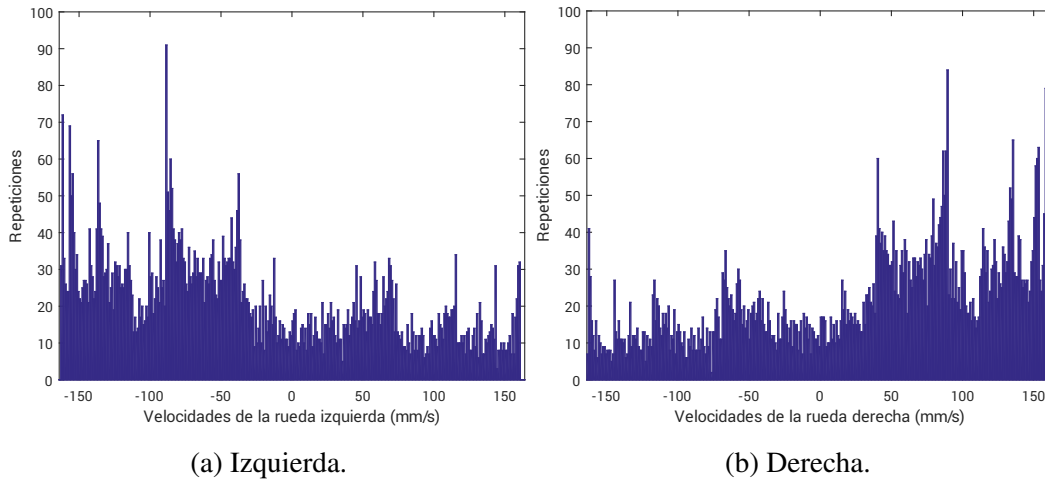


Figura 4.2 Histogramas de velocidades de las ruedas izquierda v_l y derecha v_r .

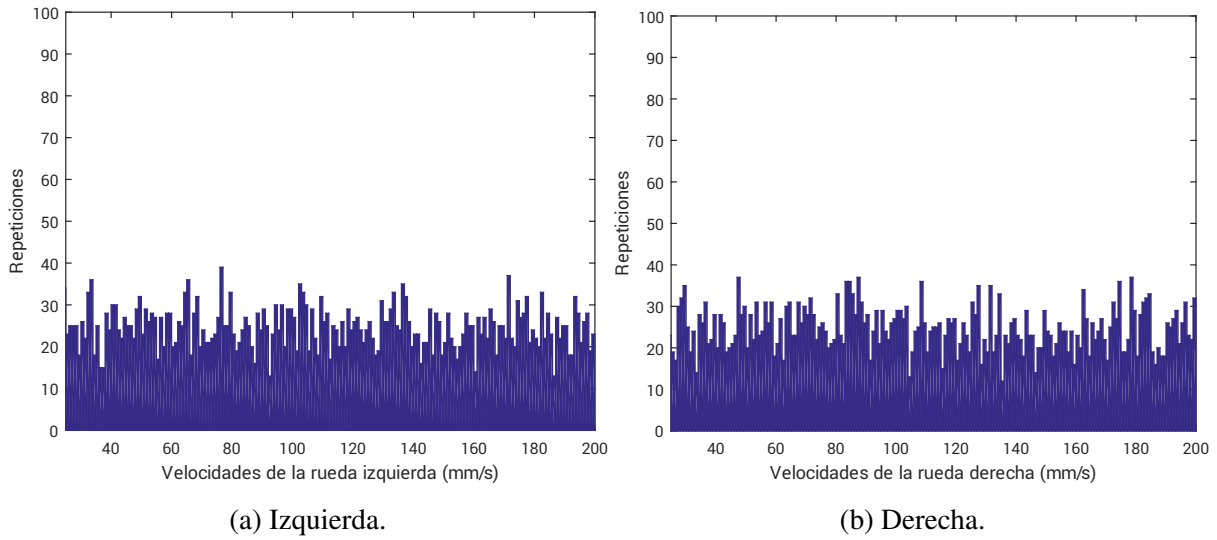


Figura 4.3 Histogramas resultantes con dispersión de velocidades lineales de las ruedas izquierda v_{dl} y derecha v_{dr} .

Por último, se envían estas velocidades al robot móvil diferencial y se deja en ejecución por dos segundos. El tiempo de ejecución de dos segundos en el paso 6, es para permitir que el robot tenga la oportunidad de desplazarse pertinentemente y pueda tener mayor cobertura del área de trabajo.

El número de iteraciones está determinado por la aplicación, cada una con un tiempo de ejecución de dos segundos. Por ejemplo, si se requiere que el robot opere por dos minutos (es decir, 120 segundos), el número deseado de iteraciones sería 60. El número de iteraciones puede también estar relacionado a otro tipo de aplicaciones; por ejemplo, el algoritmo podría detenerse cuando un objeto sea encontrado (véase la sección 4.2.5); o cuando se alcance un porcentaje de cobertura deseado.

Despejando las velocidades de las ruedas izquierda (v_l) y derecha (v_r) del sistema de ecuaciones (3.2), descrito en la sección 3.3, obtenemos los histogramas mostrados en la figura 4.2. Se puede observar que los valores de v_l son mayormente negativos y que los de v_r son principalmente positivos, lo cual es indicio de que las trayectorias del robot se basarán principalmente en giros cerrados hacia la izquierda. Comportamientos muy similares pueden ser observados con diversos números de iteraciones. Debido a esto, a los valores de v_l y v_r se les realiza una dispersión de manera que el histograma de las velocidades de las ruedas sea uniforme, como se mostró en la figura 4.3. Además, la dispersión permite realizar un re-escalamiento de los

valores a un rango utilizable por el robot móvil diferencial, limitando sus velocidades mínimas y máximas que puede alcanzar.

Intencionalmente el rango de velocidades no incluye velocidades negativas, de manera que el robot no se mueva en reversa, así, la trayectoria generada abarcará una mayor área de cobertura en menor tiempo.

Al realizar las simulaciones numéricas del algoritmo, realizando 20 iteraciones del mismo, obtenemos un resultado mostrado en la figura 4.4. La línea de colores representa la trayectoria generada, donde cada color representa un segmento de la trayectoria recorrida por el robot móvil, generado por cada iteración del algoritmo.

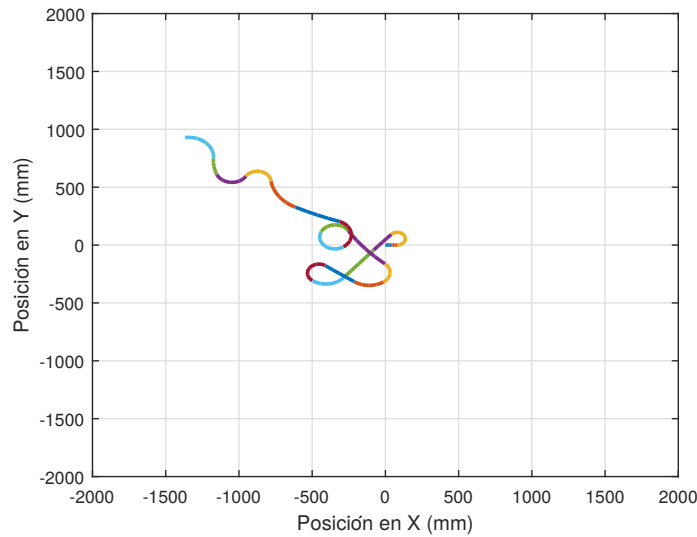


Figura 4.4 Trayectoria caótica del robot móvil generada por el algoritmo propuesto para $n = 20$.

Se puede observar que la trayectoria obtenida es impredecible. Sin embargo, en una aplicación real existen límites físicos (muros, puertas, obstáculos, etc.) que impiden un movimiento libre del robot. El siguiente paso es determinar una superficie de trabajo en la que el robot desempeñe su trayectoria. El área de trabajo elegida es una mesa que se encuentra en el *Laboratorio de Sincronización y Sistemas Complejos* (Departamento de Electrónica y Telecomunicaciones, CICESE) de 1400 mm de ancho por 2500 mm de largo que cuenta con bordes de 35 mm de altura. La mesa se muestra en la figura 4.5.



Figura 4.5 Delimitación del área de trabajo, mesa de 1400 *mm* de ancho por 2500 *mm* de largo. Laboratorio de Sincronización y Sistemas Complejos.

El objetivo es que el robot móvil diferencial se desplace sobre la superficie de esta mesa de manera caótica. El robot debe cambiar su trayectoria cuando llega a alguno de los límites del área de trabajo (bordes de la mesa) para evitar colisionar con ellos. En este caso, se adopta un movimiento reflexivo. Cuando el robot sensa una aproximación a uno de estos límites, el robot cambia su dirección de movimiento siguiendo la estrategia mostrada en la figura 4.6. Es decir, toma una trayectoria de salida en un ángulo α que refleja el de entrada, donde α es el ángulo formado entre el frente del robot móvil y el borde el área de trabajo.

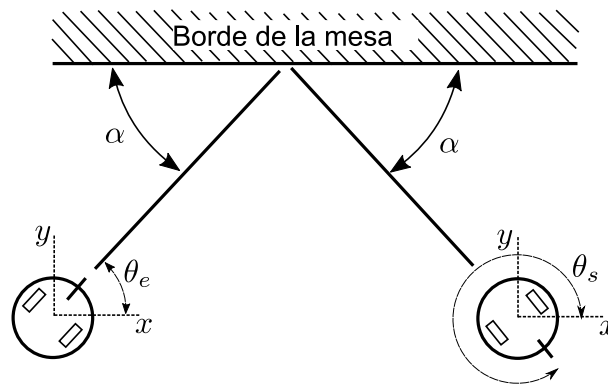


Figura 4.6 Estrategia reflexiva de evasión de bordes del robot móvil diferencial.

Tomando θ_e como el ángulo de entrada formado entre el frente del robot móvil y el eje horizontal tenemos:

$$\alpha = \theta_e, \quad (4.1)$$

Por lo tanto, el ángulo de salida θ_s del robot móvil será:

$$\theta_s = 270 + \alpha. \quad (4.2)$$

De manera similar se realiza la operación para todos los límites del área de trabajo.

4.2. Resultados numéricos

En esta sección se reportan los resultados de las simulaciones numéricas obtenidos al implementar el algoritmo generador de trayectorias caóticas, descrito en la sección anterior (figura 4.1), así como diferentes pruebas que nos indican, entre otras cosas, el porcentaje de cobertura del área de trabajo realizada por el robot móvil diferencial al seguir la trayectoria generada y determinar si existe o no caos en ellas.

4.2.1. Simulaciones

Una vez considerados los límites del área de trabajo, se realizaron simulaciones numéricas del algoritmo generador de trayectorias caóticas propuesto en esta tesis. Iniciando la trayectoria del robot móvil diferencial en el origen $(x_0, y_0, \theta_0) = (0, 0, 0^\circ)$, se realizan simulaciones numéricas del algoritmo generador de trayectorias caóticas para 10, 100, 300 y 500 iteraciones.

El cuadro en negro representa los límites del área de trabajo (mesa en el laboratorio) descrita anteriormente (figura 4.5). En las trayectorias del robot móvil, cada color representa un lapso de tiempo de dos segundos durante el cual, el robot utiliza las velocidades obtenidas del mapa caótico de Hénon, esto por cada iteración antes de pasar a las siguientes.

En la figura 4.7 se aprecia que la trayectoria del robot es impredecible y que con 300 iteraciones cubre (patrulla) gran parte del área de trabajo.

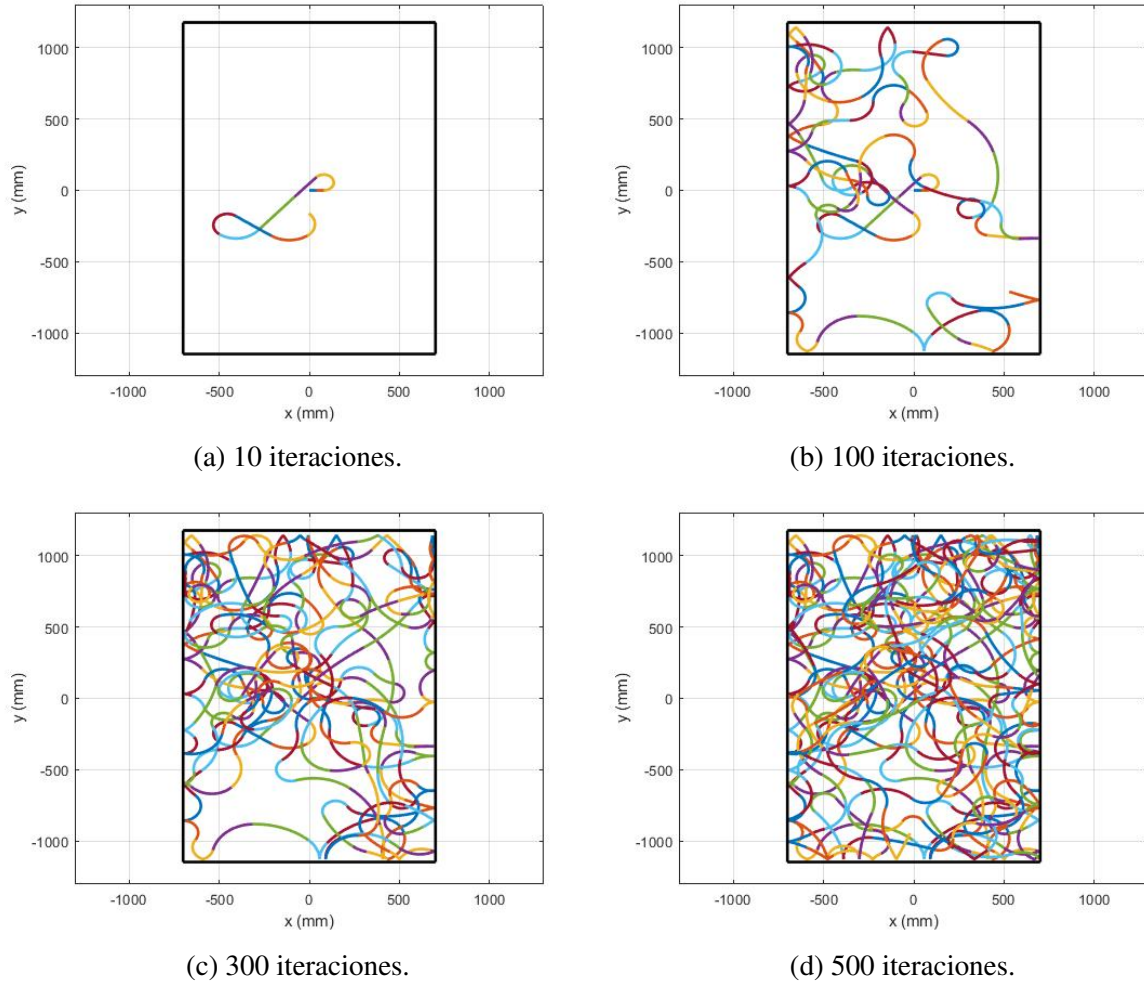


Figura 4.7 Trayectoria del robot móvil con las mismas condiciones iniciales $(x_0, y_0) = (0, 0)$ y diferente número de iteraciones.

Se puede observar también de estos resultados (figura 4.7), que las trayectorias descritas con determinado número de iteraciones, en todos los casos bajo las mismas condiciones iniciales, es la misma, confirmando la *naturaleza determinista del caos en el desplazamiento del robot*. Por otra parte, se realiza la simulación utilizando tres diferentes posiciones iniciales (en milímetros) con la misma orientación ($\theta_0 = 0^\circ$) en los casos a) $x_0 = 0, y_0 = 0$; b) $x_0 = 0.01, y_0 = 0$ y c) $x_0 = -350, y_0 = 500$. Para cada caso, se realizan 200 iteraciones del algoritmo con el fin de comparar las trayectorias generadas por estas condiciones iniciales. Los resultados se muestran en la figura 4.8.

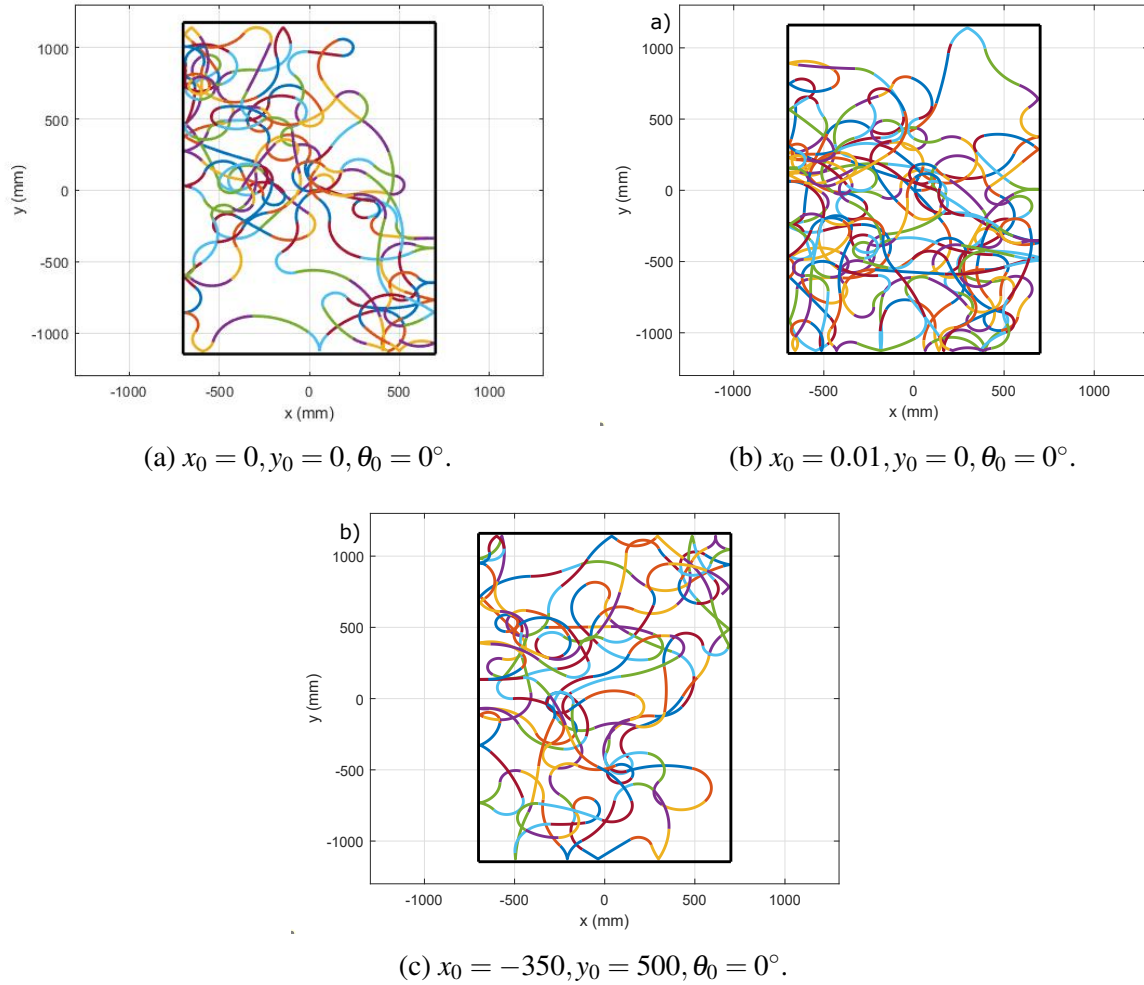


Figura 4.8 Trayectorias generadas por el robot diferencial para diferentes condiciones iniciales con 200 iteraciones.

Como se puede observar en la figura 4.8, las trayectorias cambian drásticamente a pesar de tener un cambio de únicamente 0.1 milímetros en el eje x , verificando así la propiedad de *sensibilidad a condiciones iniciales* en las trayectorias del robot, característica de un *sistema caótico*.

4.2.2. Porcentaje de cobertura del área de trabajo

Para poder estimar el área de trabajo cubierta por las trayectorias caóticas impuestas al robot móvil, se dividió el área de trabajo en una cuadrícula de 12×20 (240) cuadros con medidas

laterales de 76.66 mm, esto para tener una idea más precisa de la efectividad de cobertura del algoritmo propuesto en aplicaciones a tareas de exploración o búsqueda.

En la figura 4.9 se muestra el porcentaje de cobertura del área de prueba por el robot móvil diferencial para 100, 200, 300 y 400 iteraciones del algoritmo, cuando las condiciones iniciales son $x_0 = 0, y_0 = 0, \theta_0 = 0^\circ$. Podemos observar que apenas con 100 iteraciones el robot patrulla más de la mitad del área de prueba y con 300 iteraciones cubrió casi en su totalidad la superficie de trabajo.

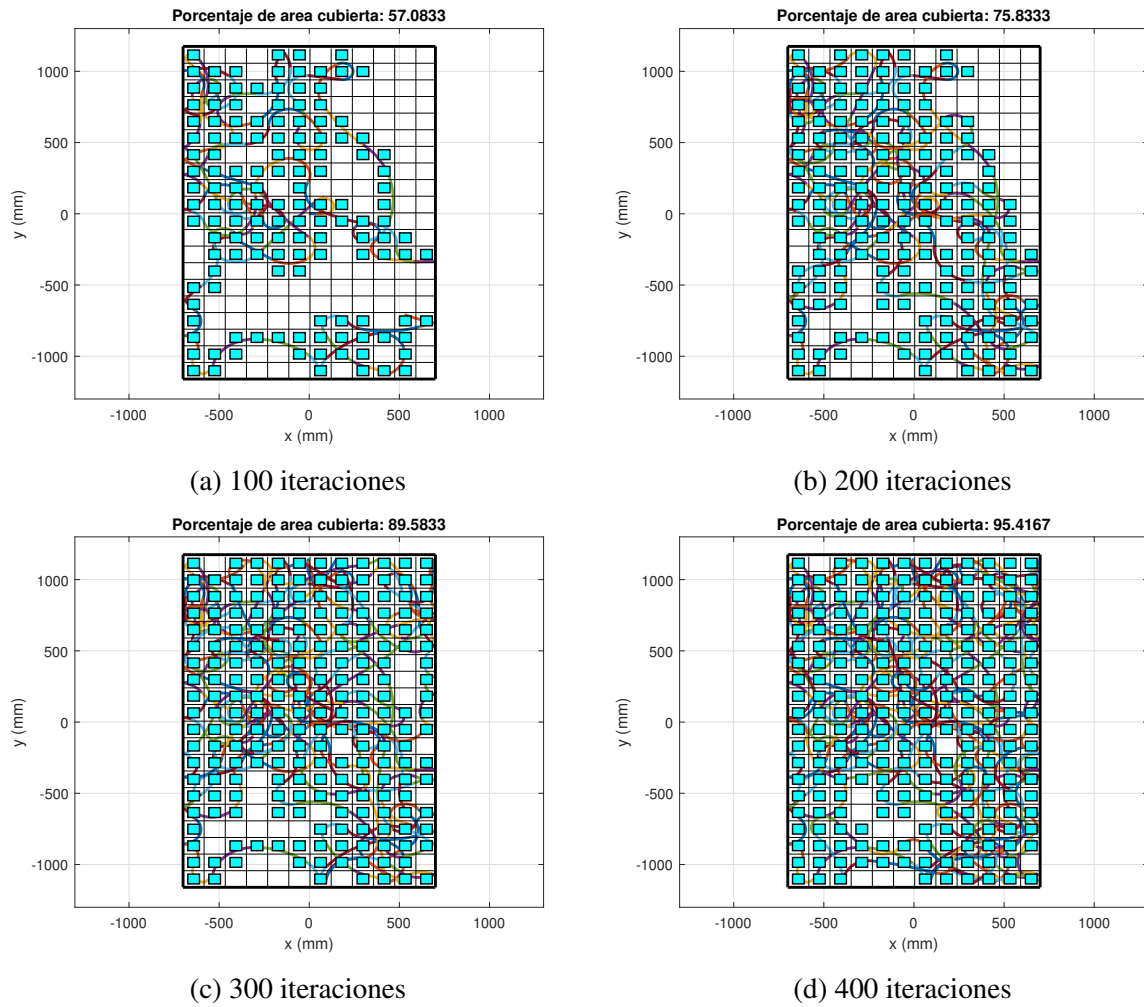


Figura 4.9 Porcentajes del área de trabajo cubiertas por el robot móvil diferencial para diferentes números iteraciones.

En la figura 4.10 se muestran los resultados numéricos obtenidos utilizando trayectorias generadas por 300 iteraciones del algoritmo con posiciones iniciales elegidas arbitrariamente para el robot móvil diferencial.

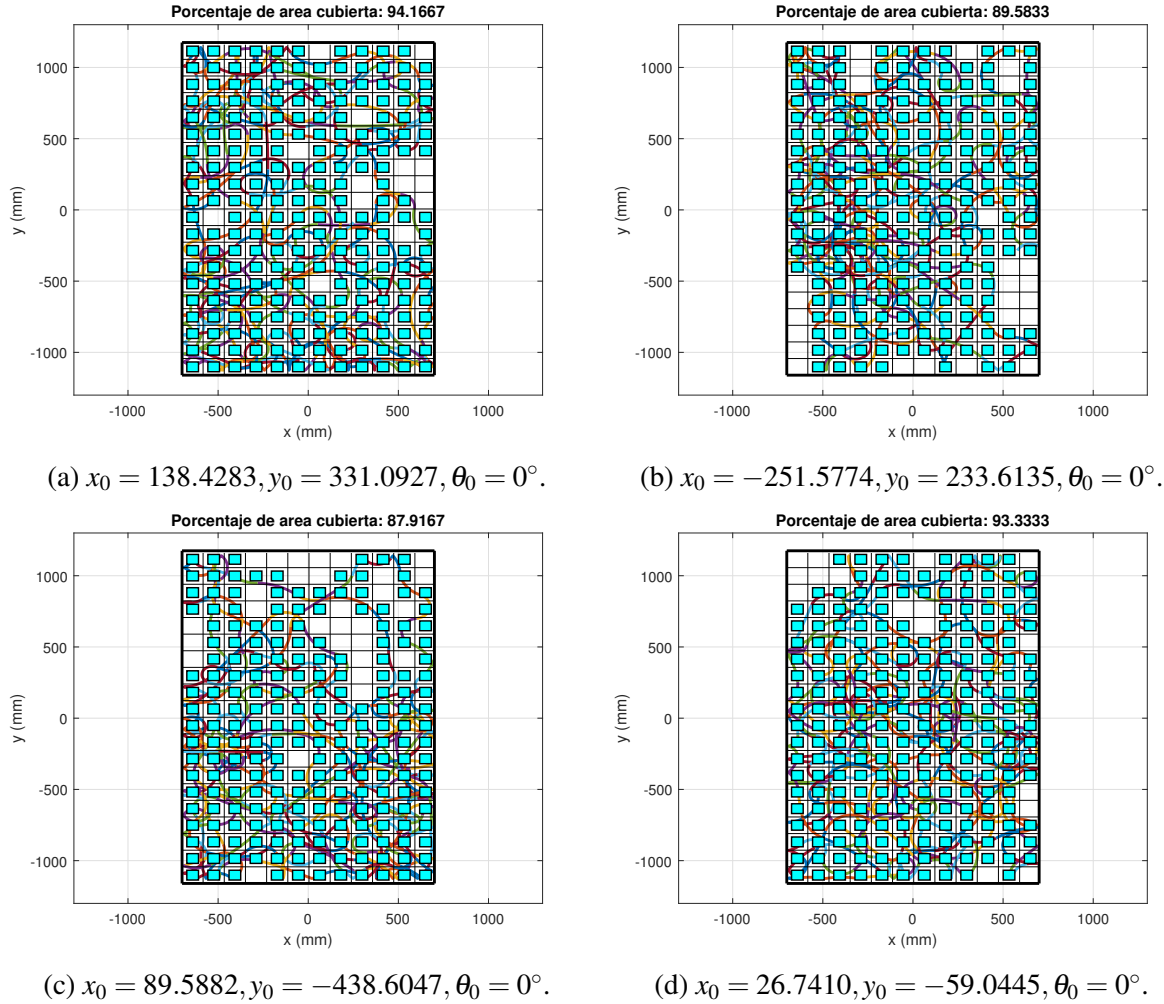


Figura 4.10 Porcentajes del área de trabajo cubiertas por el robot móvil diferencial para diferentes condiciones iniciales arbitrarias para 300 iteraciones.

En la tabla 4.1 se reportan los porcentajes de cobertura por el robot móvil diferencial obtenidos en 10 trayectorias diferentes generadas con condiciones iniciales elegidas arbitrariamente.

Repitiendo 1000 veces el proceso anterior, se obtuvo un promedio de 92.7083 % de área cubierta. Si tomáramos en consideración las medidas reales del robot Khepera III, el área cubierta por 300 iteraciones sería prácticamente del 100 %.

Tabla 4.1 Porcentajes de cobertura del área de prueba con posiciones iniciales arbitrarias para 300 iteraciones.

Porcentaje de cobertura del área de trabajo.		
x_0	y_0	% de cobertura de área
168.8192	-140.1049	89.1667
89.9301	-427.4573	92.9167
135.4415	-608.6173	97.0833
-250.7693	-589.9773	93.7500
-412.5814	420.4952	92.5000
125.3458	-237.7700	95.0000
355.1998	-605.2201	97.0833
-105.1301	-153.9740	94.1667
188.9651	383.7599	91.2500
-98.9724	190.2068	90.4167

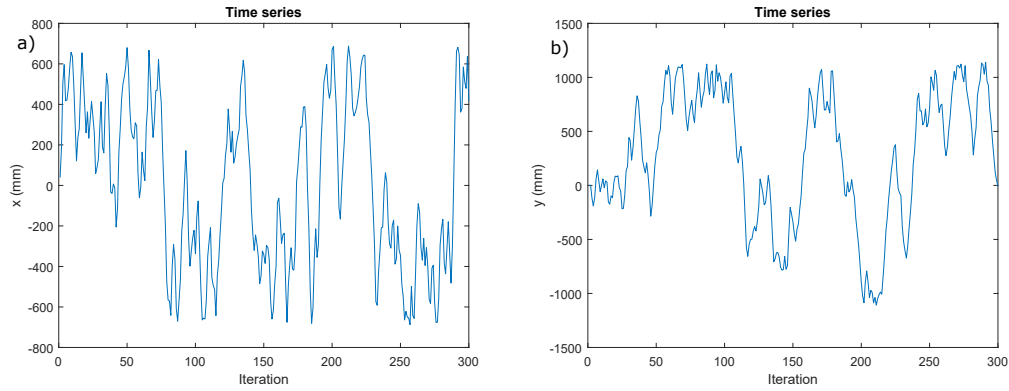
4.2.3. Prueba 0-1 Gottwald-Melbourne para determinación de caos de la trayectoria del robot móvil

Para comprobar que las trayectorias generadas por el algoritmo propuesto en esta tesis para el robot, son efectivamente caóticas, utilizamos la prueba 0-1 Gottwald-Melbourne [69]. Ellos describen una prueba para determinar si un sistema dinámico determinista discreto es caótico o no. Al contrario del método usual de calcular el máximo exponente de Lyapunov, este método se aplica directamente a la serie de tiempo y no requiere reconstrucción espacio-fase. Además, la dimensión del sistema dinámico y la forma de sus ecuaciones son irrelevantes.

La entrada son los datos de la serie de tiempo y la salida se encuentra en el rango entre 0 y 1, dependiendo si la dinámica es caótica o no, siendo 0 *no caótica* y 1 *totalmente caótica*. La prueba se aplica a cualquier sistema dinámico determinista, ver detalles de la prueba en [69].

Para realizar la prueba se ejecutan 1000 iteraciones del algoritmo generador de trayectorias caóticas al robot móvil con 3 condiciones iniciales diferentes: a) $x_0 = 0, y_0 = 0$; b) $x_0 = 0.01, y_0 = 0$ y c) $x_0 = -350, y_0 = 500$, que son las mismas que se mostraron en la figura 4.8. De igual manera, en los tres casos tenemos un ángulo inicial ($\theta_0 = 0^\circ$). Tomando en consideración los datos de las series de tiempo de los estados x y y del modelo del robot diferencial, obtenidos de los resultados de las simulaciones numéricas realizadas (mostrados en la figuras 4.11, 4.12

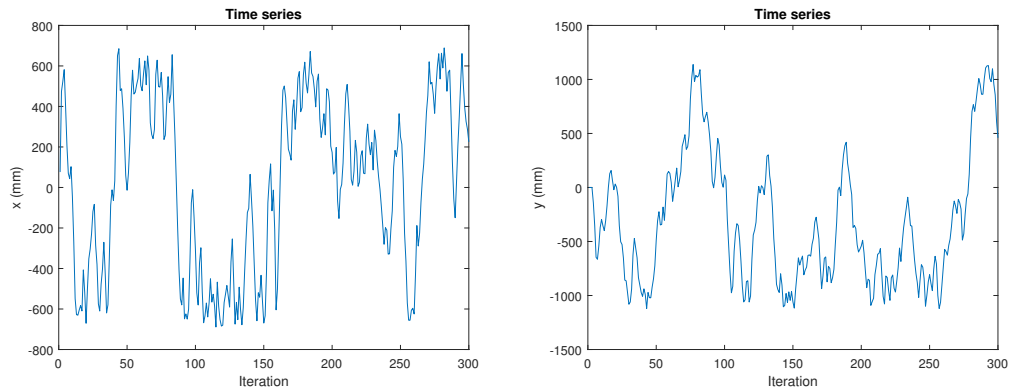
y 4.13) podemos realizar la prueba 0-1 para verificar evidencia de caos en las trayectorias del robot móvil generadas por el algoritmo.



(a) Serie de tiempo de x .

(b) Serie de tiempo de y .

Figura 4.11 Series de tiempo resultantes con condiciones iniciales $x_0 = 0, y_0 = 0$.



(a) Serie de tiempo de x .

(b) Serie de tiempo de y .

Figura 4.12 Series de tiempo resultantes con condiciones iniciales $x_0 = 0.01, y_0 = 0$.

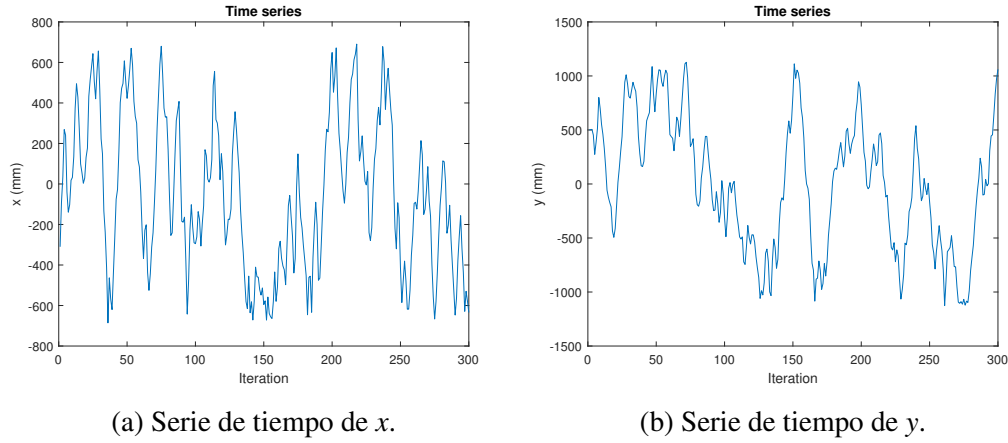


Figura 4.13 Series de tiempo resultantes con condiciones iniciales $x_0 = -350, y_0 = 500$.

Los resultados que arroja la prueba, considerando las series de tiempo a), b) y c), se reportan en la tabla 4.2. El número cercano a 1 es un indicio claro de que la prueba concluye que dichas series de tiempo son efectivamente caóticas.

Tabla 4.2 Resultados numéricos de la prueba 0-1 de Gottwald-Melbourne.

Resultados de la prueba 0-1 de Gottwald-Melbourne		
Condiciones iniciales	Resultado para x	Resultado para y
$x_0 = 0, y_0 = 0, \theta_0 = 0^\circ$	0.9852	0.9733
$x_0 = 0.01, y_0 = 0, \theta_0 = 0^\circ$	0.9865	0.9545
$x_0 = -350, y_0 = 500, \theta_0 = 0^\circ$	0.9978	0.9567

Al realizar un promedio de los resultados obtenidos de esta prueba para 100 trayectorias generadas por condiciones iniciales elegidas arbitrariamente del robot móvil diferencial, obtenemos valores para el eje x de 0.9982 y para el eje y de 0.9733. En la siguiente subsección se reportará otra prueba más de la existencia de caos en estas trayectorias.

4.2.4. Prueba de conteo de cajas ("Box-Counting")

La geometría fractal provee un modelo matemático para muchos objetos complejos encontrados en la naturaleza [70], tales como costas, montañas y nubes. Estos objetos son demasiado

complejos para poseer medidas típicas y ser descritos por la geometría tradicional euclidiana. Como se mencionó en la sección 2.2, la autosimilaridad es una propiedad esencial de los fractales en la naturaleza y puede ser cuantificada por la *dimensión fraccionaria*.

Algunas técnicas utilizadas para estimar la dimensión fraccionaria (no entera) de un objeto geométrico fueron discutidas por Mandelbrot [70]. En este trabajo de tesis, el método de conteo de cajas ("box-counting") fue utilizado debido a que es sencillo, automáticamente calculable con el uso de computadora y aplicable para patrones con o sin autosimilaridad [71]. En este método de conteo de cajas, cada imagen u objeto geométrico es cubierto por una secuencia de cuadrículas de medidas descendientes y para cada una de las cuadrículas se almacenan dos valores: el número de cuadros (cajas) intersectadas por la imagen $N(l)$ y la longitud lateral de cada cuadro l . La pendiente de regresión D de la línea recta formada por la gráfica de $\log(N(l))$ contra $\log(\frac{1}{l})$ indica el grado de complejidad (o dimensión fraccionaria) entre 1 y 2; $1 \leq D \leq 2$ [70]. Una imagen con dimensión fraccionaria de 1 ó 2, se considera completamente diferenciable. La regresión lineal usada para estimar la dimensión fraccionaria [72] está dada por

$$\log(N(l)) = \log(K) + D \log\left(\frac{1}{l}\right) \quad (4.3)$$

Donde K es una constante y $N(s)$ es proporcional a $\frac{1}{l}^{-D}$ [70]. Despejando D de (4.3) obtenemos la siguiente expresión,

$$D = \frac{\log(N(l))}{\log \frac{1}{l}}. \quad (4.4)$$

Para realizar esta prueba, retomamos la cuadrícula realizada en la sección 4.2.2, la cual fue utilizada para calcular el porcentaje de área cubierta por las trayectorias del robot móvil generadas por el algoritmo diseñado en este trabajo de tesis. El tamaño de la cuadrícula con $l = 1$, se tomará para el caso en el que el largo del área de trabajo (recordamos que las dimensiones del área de prueba son $1400 \times 2500 \text{ mm}$) es igual a la longitud de un cuadro de la cuadrícula. Comenzamos con cuadros de tamaño $l = 0.111$, las cuales se irán reduciendo hasta obtener cuadrículas muy pequeñas de tamaño $l = 0.0200$.

Como primera prueba se toma la trayectoria del robot móvil generada por las condiciones iniciales $x_0 = 0$, $y_0 = 0$ y $\theta_0 = 0^\circ$. En la figura 4.14 se muestran los porcentajes de área cubierta por el robot móvil para 4 diferentes tamaños de cuadrícula para el área de trabajo propuesta: a) 135 cuadros de $l = 0.0667$, b) 375 cuadros de $l = 0.0400$, c) 735 cuadros de $l = 0.0286$ y d) 1500 cuadros de $l = 0.0200$.

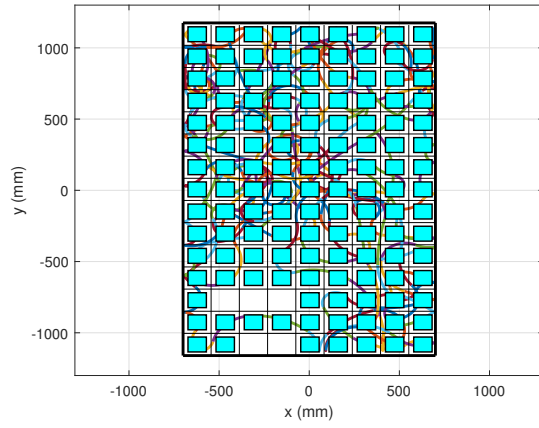
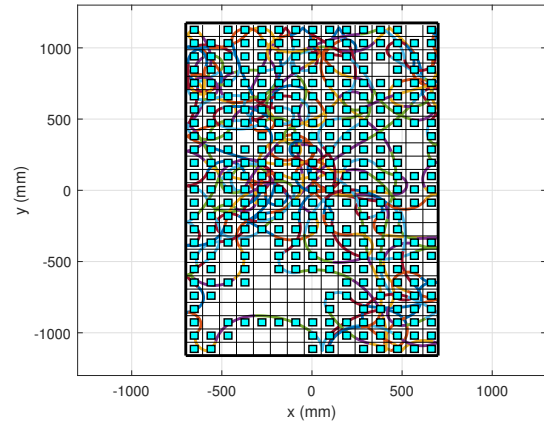
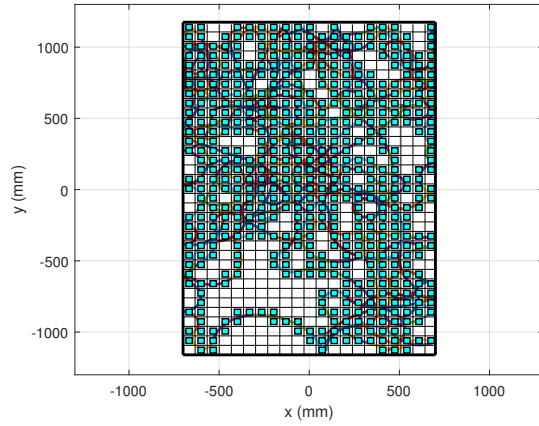
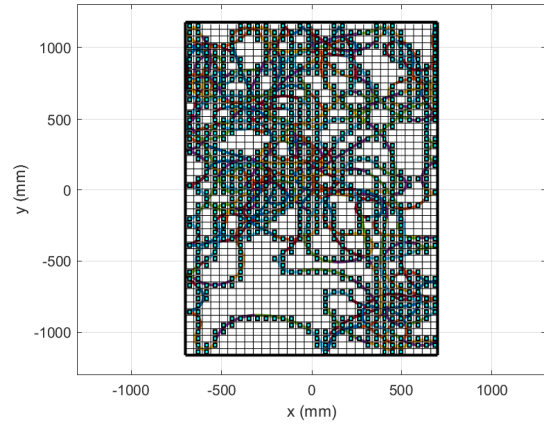
(a) 135 cuadros de $l = 0.0667$.(b) 375 cuadros de $l = 0.0400$.(c) 735 cuadros de $l = 0.0286$.(d) 1500 cuadros de $l = 0.0200$.

Figura 4.14 Porcentajes de cobertura para diferentes tamaños de cuadro.

En la tabla 4.3 se muestran los resultados obtenidos para 25 reducciones de la cuadrícula. La última columna muestra el resultado de D después de aplicar la expresión (4.4).

Tabla 4.3 Prueba de conteo de cajas para las condiciones iniciales del robot móvil $x_0 = 0, y_0 = 0$.

Datos para la prueba de conteo de cajas				
N	l	$\log(N)$	$\log(\frac{1}{l})$	D
59	0.1000	4.0775	2.3026	1.7709
81	0.0833	4.3944	2.4849	1.7685
108	0.0714	4.6821	2.6391	1.7742
129	0.0667	4.8598	2.7081	1.7946
164	0.0588	5.0999	2.8332	1.8000
194	0.0526	5.2679	2.9444	1.7891
223	0.0500	5.4072	2.9957	1.8050
262	0.0455	5.5683	3.0910	1.8014
303	0.0417	5.7137	3.1781	1.7979
337	0.0400	5.8201	3.2189	1.8081
378	0.0370	5.9349	3.2958	1.8007
421	0.0345	6.0426	3.3673	1.7945
457	0.0333	6.1247	3.4012	1.8007
510	0.0313	6.2344	3.4657	1.7989
557	0.0294	6.3226	3.5264	1.7929
601	0.0286	6.3986	3.5553	1.7997
643	0.0270	6.4661	3.6109	1.7907
700	0.0256	6.5511	3.6636	1.7882
740	0.0250	6.6067	3.6889	1.7910
792	0.0238	6.6746	3.7377	1.7858
848	0.0227	6.7429	3.7842	1.7819
885	0.0222	6.7856	3.8067	1.7826
943	0.0213	6.8491	3.8501	1.7789
993	0.0204	6.9007	3.8918	1.7731
1048	0.0200	6.9546	3.9120	1.7778

Al graficar $\log(N(s))$ contra $\log(\frac{1}{l})$ y realizar la regresión lineal a los datos mostrados obtenemos la figura 4.15. Se puede apreciar un comportamiento muy similar al de la recta dada por la ecuación $y = 1.7815x + 0.0276$.

Notamos que la pendiente de esta recta (1.7815) es muy similar a los valores de D obtenidos con la expresión (4.4). Lo anterior comprueba que la trayectoria descrita por el robot móvil tiene una *dimensión fraccionaria* y por lo tanto es *caótica*.

En el caso de la trayectoria del robot móvil generada por las condiciones iniciales $x_0 = 0.01$, $y_0 = 0$ y $\theta_0 = 0^\circ$, consideramos ahora 12 valores de l y medimos los valores correspondientes

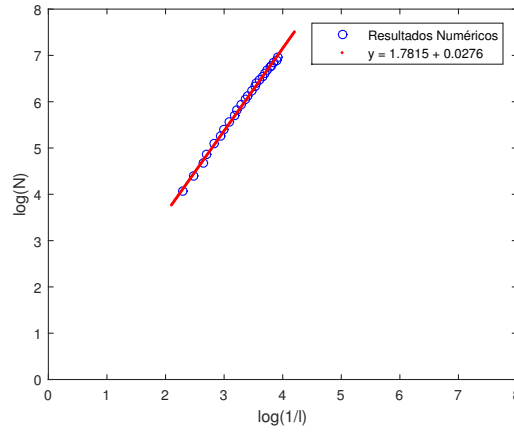


Figura 4.15 Resultado de la prueba de conteo de cajas para las condiciones iniciales $x_0 = 0$, $y_0 = 0$.

de $\log(N)$, $\log(\frac{1}{l})$ y D . Los valores obtenidos se reportan en la tabla 4.4.

Tabla 4.4 Prueba de conteo de cajas para condiciones iniciales del robot móvil $x_0 = 0.01$, $y_0 = 0$.

Datos para la prueba de conteo de cajas				
N	l	$\log(N)$	$\log(\frac{1}{l})$	D
70	0.0833	4.2485	2.4849	1.7097
117	0.0667	4.7622	2.7081	1.7585
172	0.0526	5.1475	2.9444	1.7482
231	0.0455	5.4424	3.0910	1.7607
291	0.0400	5.6733	3.2189	1.7625
369	0.0345	5.9108	3.3673	1.7554
445	0.0313	6.0981	3.4657	1.7595
515	0.0286	6.2442	3.5553	1.7563
610	0.0256	6.4135	3.6636	1.7506
702	0.0238	6.5539	3.7377	1.7535
780	0.0222	6.6593	3.8067	1.7494
880	0.0204	6.7799	3.8918	1.7421

La figura 4.16 nos muestra la gráfica de $\log(N(s))$ contra $\log(\frac{1}{l})$. Al realizar la regresión lineal a los datos mostrados obtenemos la ecuación de la recta $y = 1.7761x - 0.0831$, lo cual comprueba que la trayectoria descrita por el robot a partir de $x_0 = 0.01$, $y_0 = 0$ y $\theta_0 = 0^\circ$ tiene una *dimensión fractal* y por lo tanto es *caótica* también.

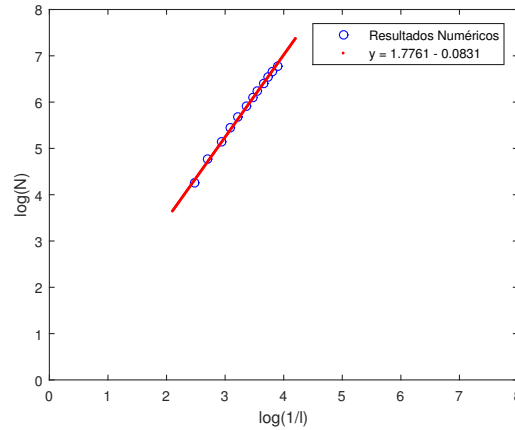


Figura 4.16 Resultado de la prueba de de conteo de cajas para condiciones iniciales del robot móvil $x_0 = 0.01$, $y_0 = 0$.

Por último, realizamos la prueba de conteo de cajas para la trayectoria descrita por el robot generada por las condiciones iniciales $x_0 = -350$, $y_0 = 500$ y $\theta_0 = 0^\circ$. los resultados obtenidos se muestran en la tabla 4.5

Finalmente, la gráfica de $\log(N(s))$ contra $\log(\frac{1}{l})$ se muestra en la figura 4.17. Al realizar la regresión lineal a los datos obtenemos la ecuación de la recta $y = 1.7729x - 0.0048$, lo cual comprueba que la trayectoria descrita por el robot móvil es *caótica* también.

Tabla 4.5 Prueba de conteo de cajas para condiciones iniciales del robot móvil $x_0 = -350$, $y_0 = 500$.

Datos para la prueba de conteo de cajas				
N	l	$\log(N)$	$\log(\frac{1}{l})$	D
83	0.0833	4.4188	2.4849	1.7783
127	0.0667	4.8442	2.7081	1.7888
187	0.0526	5.2311	2.9444	1.7766
247	0.0455	5.5094	3.0910	1.7824
318	0.0400	5.7621	3.2189	1.7901
391	0.0345	5.9687	3.3673	1.7726
473	0.0313	6.1591	3.4657	1.7771
566	0.0286	6.3386	3.5553	1.7828
651	0.0256	6.4785	3.6636	1.7684
723	0.0238	6.5958	3.7377	1.7647
833	0.0222	6.7250	3.8067	1.7666
915	0.0204	6.8189	3.8918	1.7521

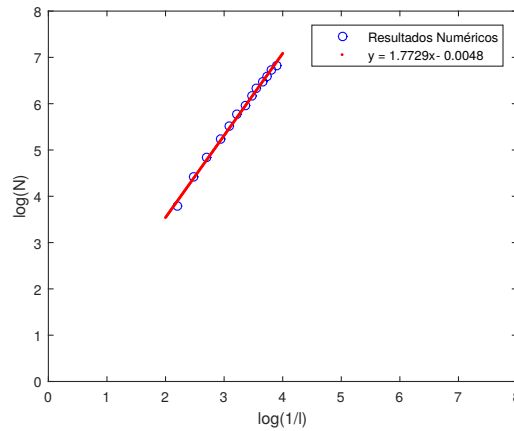


Figura 4.17 Resultado de la prueba de conteo de cajas para condiciones iniciales del robot móvil $x_0 = -350$, $y_0 = 500$.

Podemos afirmar que la prueba de conteo de cajas concluye que las trayectorias descritas por el robot móvil, generadas por el algoritmo propuesto en esta tesis, son efectivamente caóticas.

4.2.5. Simulación: Aplicación a búsqueda y detección de objetos

Adicionalmente, se simuló numéricamente una posible aplicación para las trayectorias caóticas generadas en robots móviles: la búsqueda y detección de objetos dentro de una superficie. Para alcanzar este objetivo, se colocó un objeto en el área de trabajo con ubicación en el plano (x_{obj}, y_{obj}) elegida de manera arbitraria y se almacenó la información referente al tiempo que tarda el robot móvil diferencial en encontrarlo (como se muestra en la figura 4.18). El robot inicia su curso normal en las mismas condiciones iniciales $(x_0 = 0, y_0 = 0, \theta_0 = 0^\circ)$ para todos los casos.

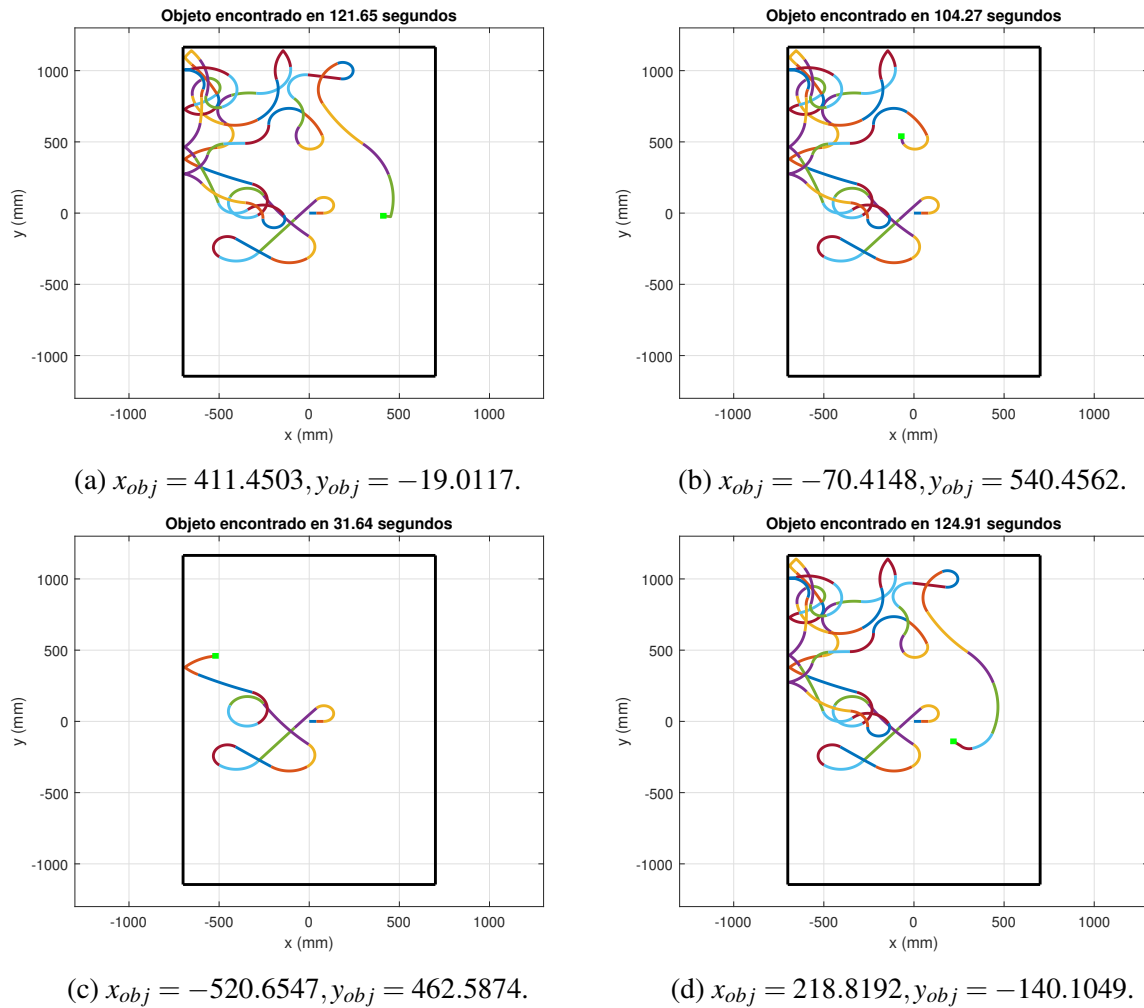


Figura 4.18 Resultados de búsqueda y detección de objetos para diferentes posiciones del objeto y condiciones iniciales del robot móvil: $x_0 = 0, y_0 = 0$.

En la tabla 4.6 se muestran los resultados obtenidos para 10 repeticiones del algoritmo generador de trayectorias caóticas.

Tabla 4.6 Resultados de búsqueda y detección de un objeto en diferentes posiciones y condiciones iniciales del robot móvil: $x_0 = 0, y_0 = 0$.

Resultados de búsqueda de objetos en el área de trabajo		
x_{obj}	y_{obj}	Tiempo de detección (s)
185.4415	-608.6173	131.79
-362.5814	420.4952	90.37
188.4283	331.0927	67.37
139.5882	-438.6027	126.85
-342.9021	-2.1267	48.06
413.7696	-207.4986	119.36
390.6096	-481.1219	134.85
-304.0359	382.5699	79.74
-243.9207	537.3386	81.25
285.5729	479.3031	98.07

Al realizar 1000 repeticiones del algoritmo se obtuvo que, en promedio el robot móvil diferencial tarda 96.94 segundos en encontrar el objeto ubicado en el área de trabajo.

De igual manera, se realizó la simulación para el caso en el que el robot inicia su recorrido con condiciones iniciales elegidas de manera arbitraria y el objeto a encontrar se mantiene en la misma posición ($x_{obj} = 0, y_{obj} = 0$) para todos los casos, como se muestra en la figura 4.19.

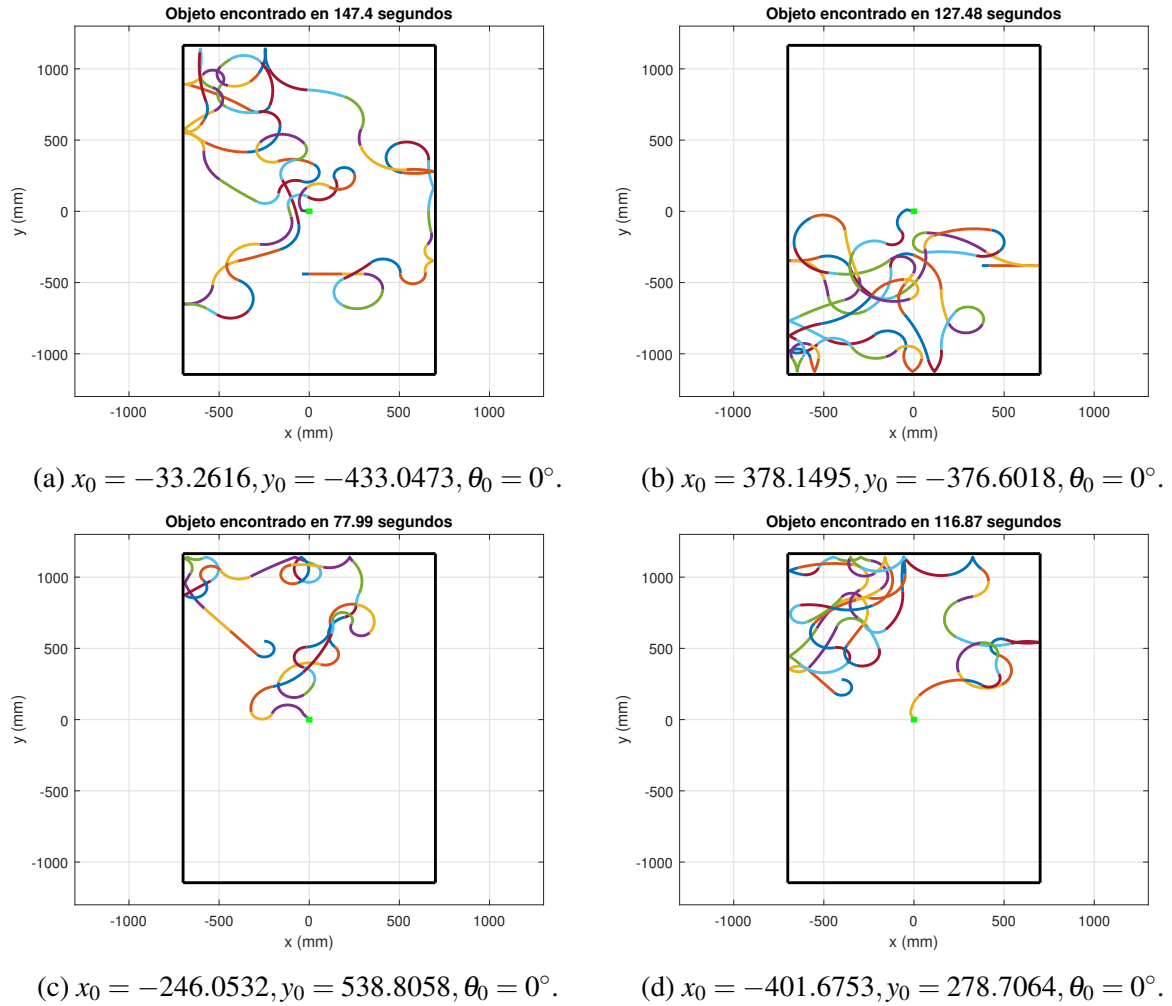


Figura 4.19 Resultado de búsqueda y detección de un objeto con ubicación fija ($x_{obj} = 0, y_{obj} = 0$) y condiciones iniciales arbitrarias del robot móvil diferencial en el área de trabajo.

En la tabla 4.7 se reportan los resultados obtenidos para 10 repeticiones del algoritmo generador de trayectorias caóticas del robot móvil diferencial.

Al repetir el proceso 1000 veces, se obtuvo un promedio de 94.41 segundos para encontrar el objeto por el robot móvil diferencial en el área de trabajo en este escenario.

Por último, en la figura 4.20 se muestran resultados de la simulación para el caso en que tanto el robot como el objeto a encontrar inician en posiciones arbitrarias.

Tabla 4.7 Resultados de búsqueda y detección de objeto con ubicación fija ($x_{obj} = 0, y_{obj} = 0$) y condiciones iniciales arbitrarias del robot móvil diferencial.

Resultados de búsqueda de objetos en el área de trabajo		
x_0	y_0	Tiempo de detección (s)
143.2002	-66.0764	60.86
28.1356	-55.7375	71.19
360.4146	478.5965	139.4
-264.2510	522.1088	134.33
188.2259	-478.4909	105.25
-67.3158	209.8807	52.8
-260.41767	-403.4963	84.11
-16.4184	-363.1242	38.76
-177.7508	-465.6737	70.1
-253.1433	543.8407	94.72

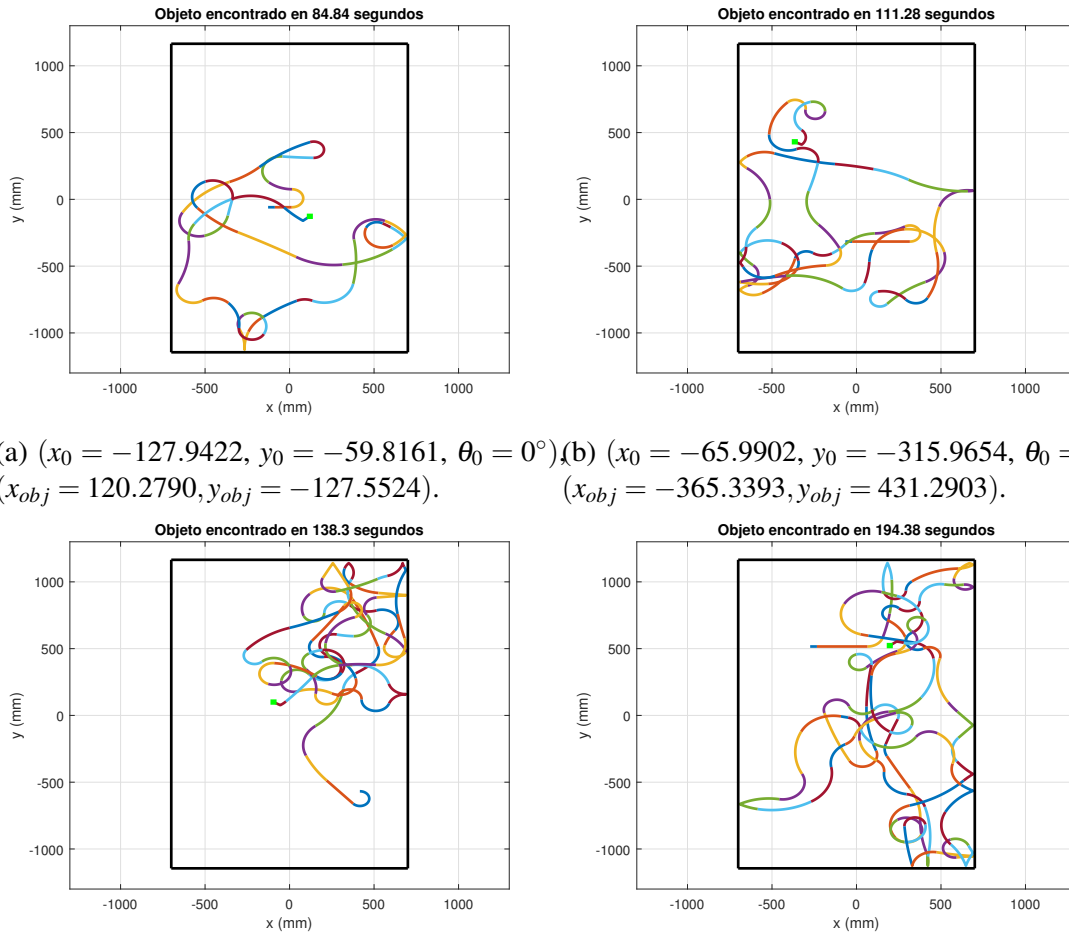


Figura 4.20 Resultado de búsqueda y detección de objeto con ubicación arbitraria y condiciones iniciales arbitrarias del robot móvil diferencial en el área de trabajo.

En la tabla 4.8 se muestran los resultados obtenidos para 10 repeticiones del algoritmo.

Tabla 4.8 Resultados de búsqueda de objeto con ubicación y condiciones iniciales arbitrarias del robot móvil diferencial.

Resultados de búsqueda de objetos en el área de trabajo				
$x_0 \text{ robot}$	$y_0 \text{ robot}$	$x_{obj} \text{ objeto}$	$y_{obj} \text{ objeto}$	Tiempo de detección (s)
-345.6779	631.4614	89.2485	491.3595	120.84
156.0968	-478.5885	-443.4993	-364.0416	39.52
-124.0528	1601.3787	183.9127	-129.6489	81.29
-78.6440	265.1067	-371.1880	-317.6409	97.93
243.1153	417.6995	-176.1055	580.0826	195.55
83.3384	-636.1013	-346.6957	-385.6962	68.1
269.0843	339.0313	179.8035	-352.1774	97.55
187.9034	-121.6191	-143.3225	-349.6585	81.28
-264.2920	653.5212	2177.7217	-509.8866	165.04
38.1595	-644.3030	-265.7645	-597.3148	104.53

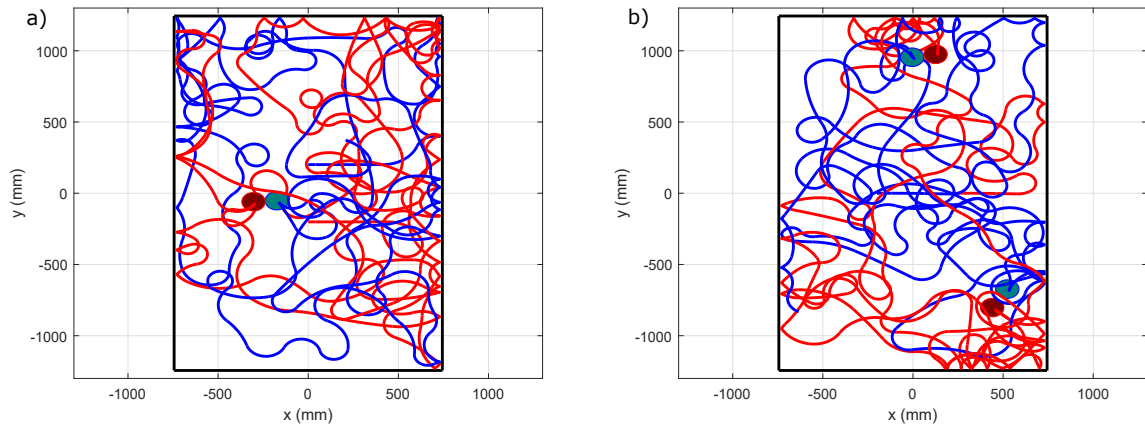
Finalmente, se obtuvo un promedio de tiempo de búsqueda de 96.17 segundos después de repetir el procedimiento 1000 veces. Se puede apreciar en todos los casos que el robot móvil diferencial Khepera III es capaz de encontrar el objeto en el área de trabajo propuesta en un período pequeño de tiempo.

4.3. Dos robots móviles en la misma área de trabajo

En esta sección analizamos el comportamiento obtenido cuando se aplica el algoritmo a múltiples robots móviles. En las siguientes secciones, mostramos los resultados obtenidos para los casos de 2, 3 y $n \leq 10$ robots que se mueven simultáneamente en el área de trabajo. Además, se muestran 3 casos en los que se divide el área de trabajo en dos secciones iguales de manera horizontal, vertical y diagonal, ocupadas cada una por un robot móvil diferencial.

La figura 4.21 muestra las trayectorias generadas por dos robots móviles diferenciales en la misma área de trabajo con diferentes condiciones iniciales. En este caso, cuando los dos

robots están cerca de chocar, se toma una estrategia reflexiva, de modo que ambos robots móviles puedan continuar su camino pero en dirección opuesta (dando un giro de 180°). Los círculos rojos y azules indican una colisión que se evita utilizando la estrategia reflectante. Cada simulación se realiza con 150 iteraciones del algoritmo propuesto. En ambos casos, se considera ($\theta_0 = 0^\circ$). La línea azul representa la trayectoria del robot 1 y la línea roja representa el robot 2.



(a) $x_{01} = 0, y_{01} = 200, x_{02} = 0, y_{02} = -200$.

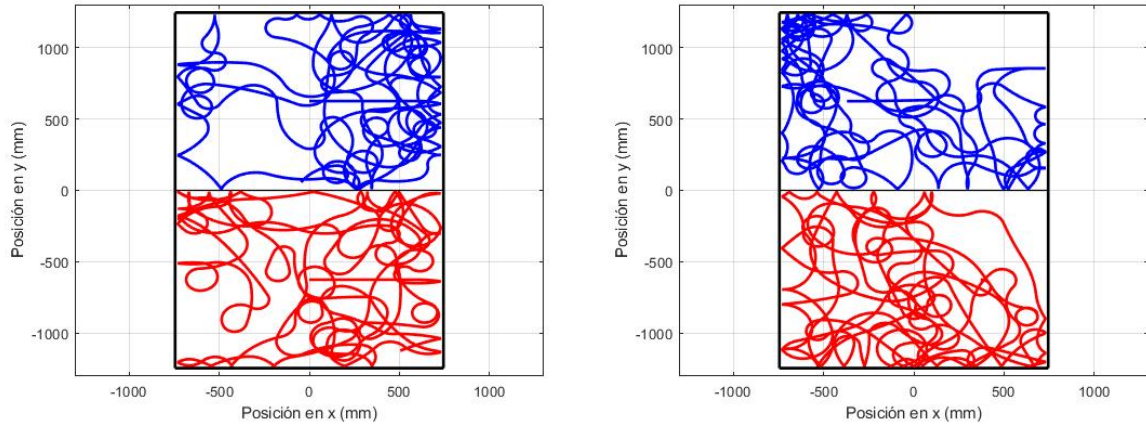
(b) $x_{01} = -200, y_{01} = 0, x_{02} = 200, y_{02} = 0$.

Figura 4.21 Trayectorias generadas para dos robots móviles.

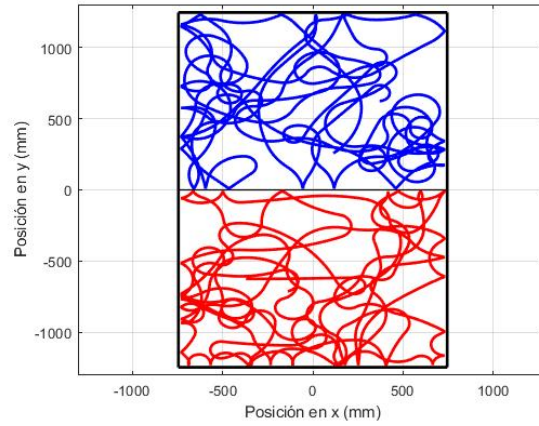
Repitiendo el procedimiento (Figura 4.21) 1000 veces con diferentes condiciones iniciales, se obtiene un porcentaje de cobertura promedio de 83.6429%. Es importante tener en cuenta que solo la mitad de las iteraciones son necesarias para cubrir un porcentaje del área de trabajo similar al caso cuando se usa un solo robot móvil.

4.3.1. División horizontal del área de trabajo

El siguiente caso es cuando se divide al área de trabajo en dos secciones de manera horizontal, es decir, cada robot cubre una sección distinta. En la figura 4.22 se muestran las trayectorias generadas para 2 condiciones iniciales diferentes.



(a) $x_{01} = 0, y_{01} = 625, \theta_{01} = 0^\circ, x_{02} = 0, y_{02} = -625, \theta_{02} = 0^\circ$.
 (b) $x_{01} = -370, y_{01} = 625, \theta_{01} = 0^\circ, x_{02} = 370, y_{02} = -625, \theta_{02} = 0^\circ$.



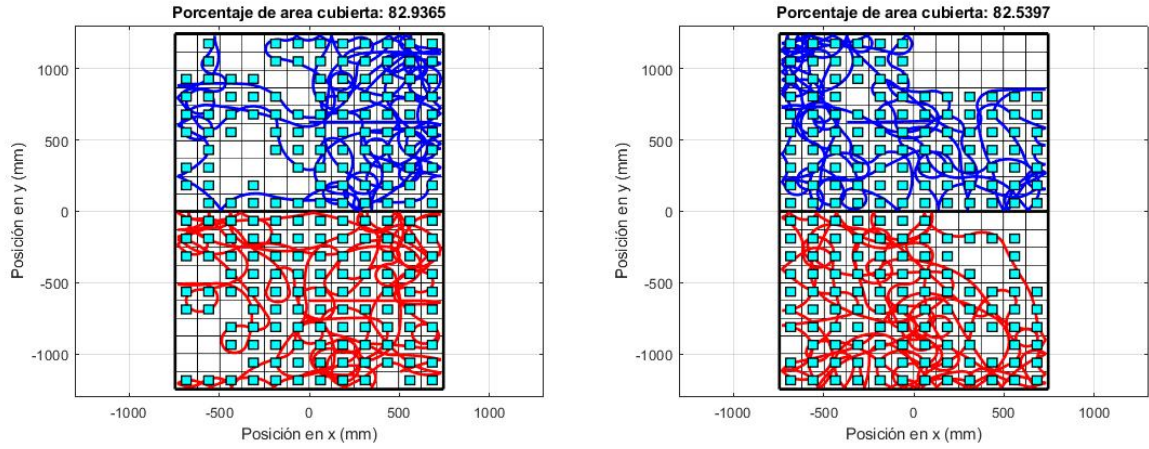
(c) $x_{01} = 370, y_{01} = 625, \theta_{01} = 0^\circ, x_{02} = -370, y_{02} = -625, \theta_{02} = 0^\circ$.

Figura 4.22 Trayectorias generadas con dos robots móviles diferenciales en el área de trabajo dividida horizontalmente.

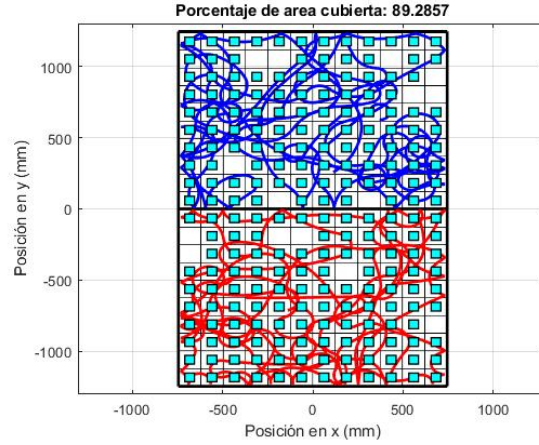
Porcentaje de cobertura

En la figura 4.23 se muestra el porcentaje de cobertura para las condiciones iniciales mostradas en la figura 4.22.

En la tabla 4.9 se muestran los porcentajes de cobertura para 5 trayectorias generadas para diferentes condiciones iniciales seleccionadas de manera arbitraria.



(a) $x_{01} = 0, y_{01} = 625, \theta_{01} = 0^\circ, x_{02} = 0, y_{02} = -625, \theta_{02} = 0^\circ$.
 (b) $x_{01} = -370, y_{01} = 625, \theta_{01} = 0^\circ, x_{02} = 370, y_{02} = -625, \theta_{02} = 0^\circ$.



(c) $x_{01} = 370, y_{01} = 625, \theta_{01} = 0^\circ, x_{02} = -370, y_{02} = -625, \theta_{02} = 0^\circ$.

Figura 4.23 Porcentajes del área de trabajo cubiertas por dos robots móviles diferenciales en el área de trabajo dividida horizontalmente.

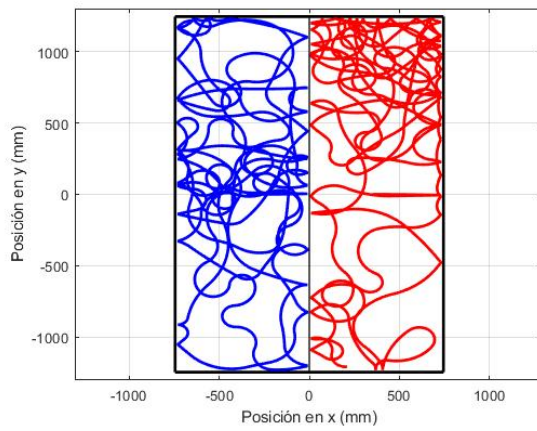
Tabla 4.9 Porcentaje de cobertura del área de trabajo.

% de área cubierta				
x_{01}	y_{01}	x_{02}	y_{02}	Porcentaje de cobertura (%)
582.3143	872.4575	647.1898	-3.3981	88.4921
473.2158	52.1492	23.9173	-793.5963	85.4173
376.1083	354.3963	152.5156	-1083.2	82.1429
211.9865	-976.8579	-13.1110	-486.2563	82.9365
-598.5754	678.2396	527.2018	315.1578	81.3492

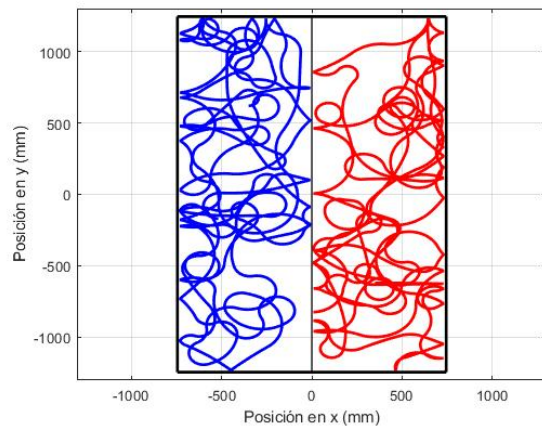
Después de repetir este experimento con 1000 condiciones iniciales diferentes, se obtiene un porcentaje promedio de cobertura del área de trabajo de 85.69%.

4.3.2. División vertical del área de trabajo

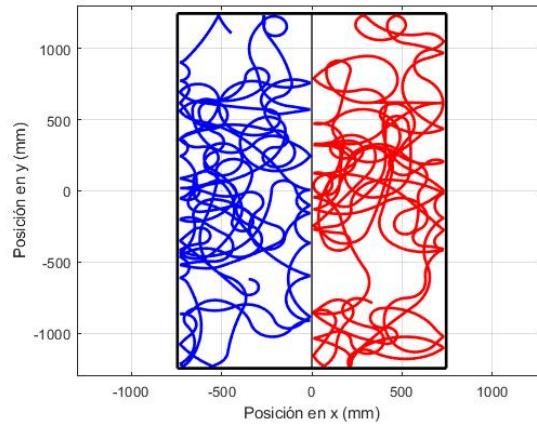
El siguiente caso es cuando se divide al área de trabajo en dos secciones de manera vertical, siguiendo las pautas de la sección anterior. En la figura 4.24 se muestran las trayectorias generadas para 3 condiciones iniciales diferentes.



(a) $x_{01} = -350, y_{01} = 0, \theta_{01} = 0^\circ, x_{02} = 350, y_{02} = 0, \theta_{02} = 0^\circ$.



(b) $x_{01} = -350, y_{01} = 620, \theta_{01} = 0^\circ, x_{02} = 350, y_{02} = -620, \theta_{02} = 0^\circ$.

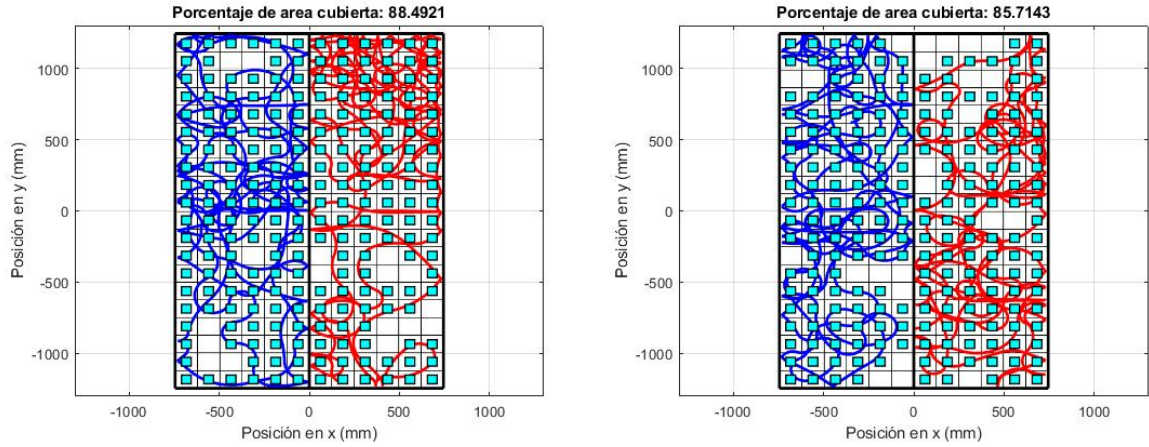


(c) $x_{01} = -350, y_{01} = -620, \theta_{01} = 0^\circ, x_{02} = 350, y_{02} = 620, \theta_{02} = 0^\circ$.

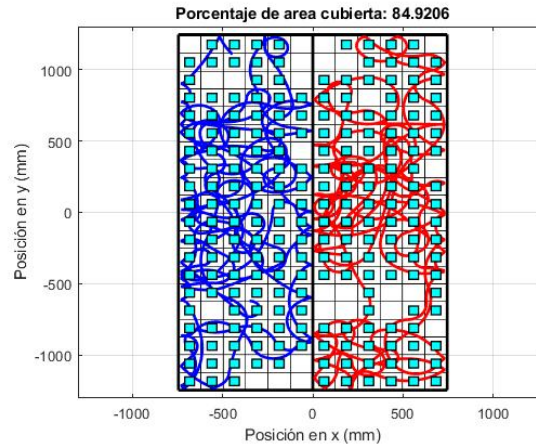
Figura 4.24 Trayectorias generadas con dos robots móviles diferenciales en el área de trabajo dividida verticalmente.

Porcentaje de cobertura

En la figura 4.25 se muestra el porcentaje de cobertura para las condiciones iniciales mostradas en la figura 4.24.



- (a) $x_{01} = -350, y_{01} = 0, \theta_{01} = 0^\circ, x_{02} = 350, y_{02} = 0, \theta_{02} = 0^\circ$.
 (b) $x_{01} = -350, y_{01} = 620, \theta_{01} = 0^\circ, x_{02} = 350, y_{02} = -620, \theta_{02} = 0^\circ$.



- (c) $x_{01} = -350, y_{01} = -620, \theta_{01} = 0^\circ, x_{02} = 350, y_{02} = 620, \theta_{02} = 0^\circ$.

Figura 4.25 Porcentajes del área de trabajo cubiertas por tres robots móviles diferenciales en el área de trabajo dividida verticalmente.

En la tabla 4.10 se muestran los porcentajes de cobertura para 5 trayectorias generadas para diferentes condiciones iniciales seleccionadas de manera arbitraria.

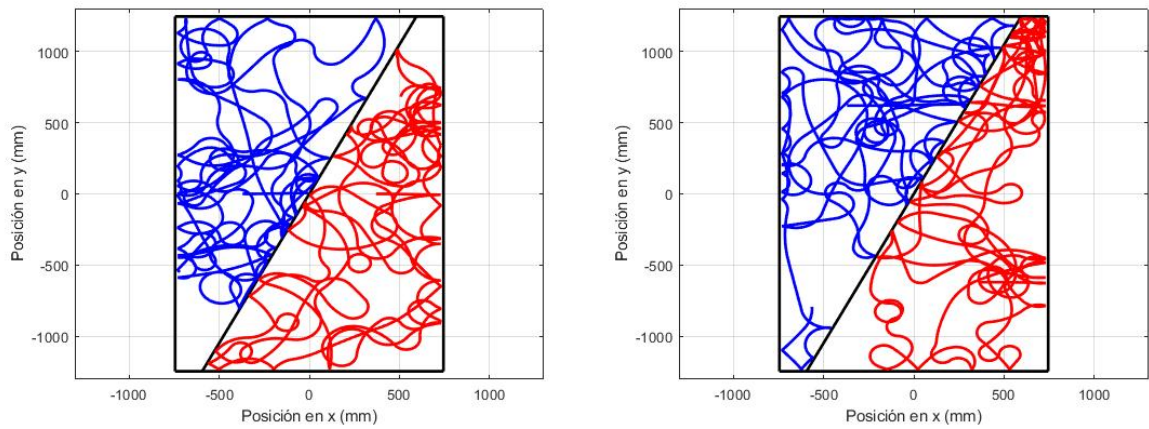
Tabla 4.10 Porcentaje de cobertura del área de trabajo.

% de área cubierta				
x_{01}	y_{01}	x_{02}	y_{02}	Porcentaje de cobertura (%)
-576.1495	342.0435	539.2936	978.7261	78.1746
-534.3689	-920.9641	171.1759	-115.1726	83.3333
-249.7197	-959.5202	573.4976	725.5550	86.5079
-148.9516	-255.9189	257.4315	1048.3	84.5238
-622.0339	321.2197	641.8082	-703.6018	81.7460

Después de repetir este experimento con 1000 condiciones iniciales diferentes, se obtiene un porcentaje promedio de cobertura del área de trabajo de 84.91 %.

4.3.3. División diagonal del área de trabajo

Similar a las dos secciones anteriores, dividimos el área de trabajo por medio de una diagonal que lo atraviesa, de manera que se formen dos secciones iguales. En la figura 4.26 se muestran las trayectorias generadas para 2 condiciones iniciales diferentes en este caso.

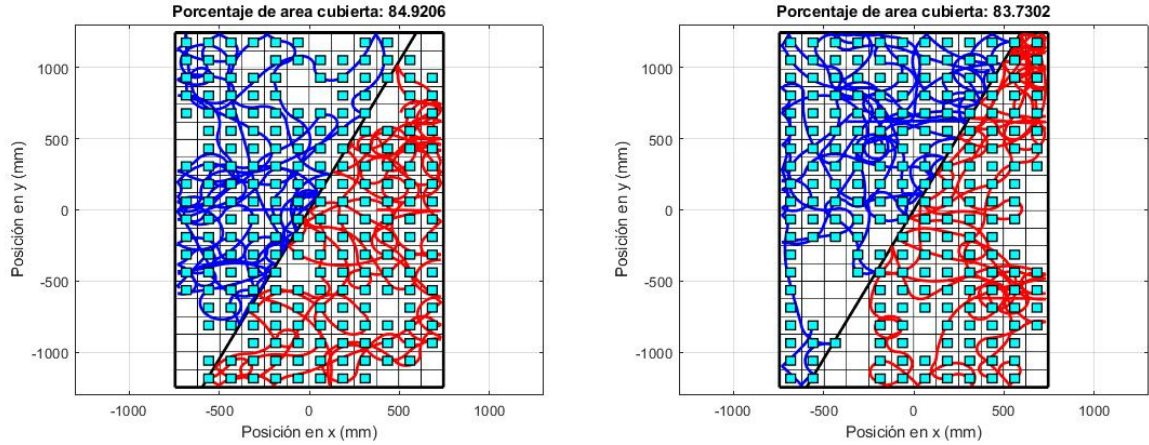


(a) $x_{01} = -370, y_{01} = 0, \theta_{01} = 0^\circ, x_{02} = 370, y_{02} = 0, \theta_{02} = 0^\circ$.
 (b) $x_{01} = -370, y_{01} = 620, \theta_{01} = 0^\circ, x_{02} = 370, y_{02} = -620, \theta_{02} = 0^\circ$.

Figura 4.26 Trayectorias generadas con dos robots móviles diferenciales en el área de trabajo dividida diagonalmente.

Porcentaje de cobertura

En la figura 4.27 se muestra el porcentaje de cobertura para las condiciones iniciales mostradas en la figura 4.26.



(a) $x_{01} = -370, y_{01} = 0, \theta_{01} = 0^\circ, x_{02} = 370, y_{02} = 0$
 (b) $x_{01} = -370, y_{01} = 620, \theta_{01} = 0^\circ, x_{02} = 370, y_{02} = -620, \theta_{02} = 0^\circ$.

Figura 4.27 Trayectorias generadas con dos robots móviles diferenciales en el área de trabajo dividida diagonalmente.

En la tabla 4.11 se muestran los porcentajes de cobertura para 5 trayectorias generadas para diferentes condiciones iniciales seleccionadas de manera arbitraria.

Tabla 4.11 Porcentaje de cobertura del área de trabajo.

% de área cubierta				
x_{01}	y_{01}	x_{02}	y_{02}	Porcentaje de cobertura (%)
117.3307	1079.1	352.2201	-87.0981	81.7460
-250.7105	-430.4872	-131.44333	-635.2320	80.5556
-660.7948	-1150.8	495.1749	-244.2492	81.7460
-373.6148	83.3545	619.1805	923.0007	82.1429
-645.2653	1094.3	648.0748	1173.9	83.3333

Después de repetir este experimento con 1000 condiciones iniciales diferentes, se obtiene un porcentaje promedio de cobertura del área de trabajo de 84.14%.

4.3.4. Comparativo

Como se puede apreciar en los resultados mostrados, el dividir el área de trabajo no influye significativamente en el porcentaje de cobertura del mismo. Esto nos indica que si tenemos un área de trabajo que, por ejemplo, consta de dos pasillos similares, el porcentaje esperado de cobertura se va a mantener al utilizar un robot móvil diferencial para cada uno.

Es importante destacar que, al utilizar dos robots móviles, únicamente se necesitan la mitad de las iteraciones para cubrir un porcentaje de cobertura similar al obtenido utilizando solamente un robot móvil.

En las siguientes secciones se analizan los casos para mas robots móviles simultáneamente (con un máximo de 10).

4.4. Tres robots móviles en la misma área de trabajo

La figura 4.28 muestra las trayectorias generadas por tres robots móviles diferenciales en movimiento simultáneamente en la misma zona de trabajo con diferentes condiciones iniciales para cada uno. De forma similar a lo explicado en la sección anterior, cuando dos o tres de los robots van a colisionar se toma la estrategia reflexiva. en la Figura 4.28, los círculos rojo, verde y azul indican una colisión que se evita usando la estrategia reflectiva. Cada simulación se realiza con 100 iteraciones del algoritmo. La línea azul representa la trayectoria del robot 1, la línea roja representa el robot 2 y la línea verde representa la trayectoria del robot 3.

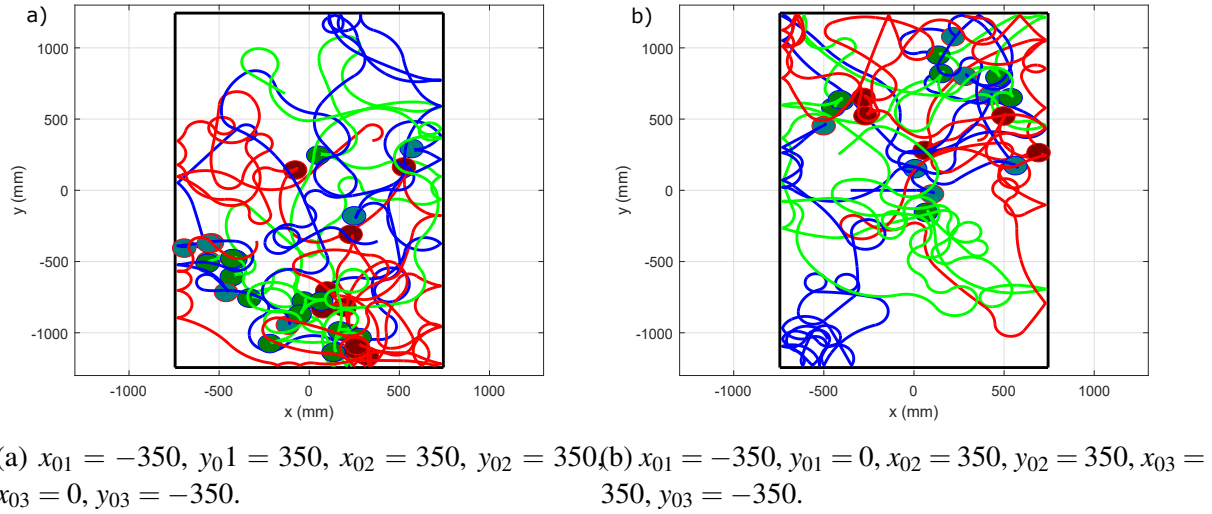


Figura 4.28 Trayectorias generadas para tres robots móviles.

Repitiendo el procedimiento 1000 veces con diferentes condiciones iniciales, se obtiene un porcentaje de cobertura promedio de 84.1826%. Podemos ver que se necesitan menos iteraciones para cubrir un porcentaje del área de trabajo similar al caso cuando se usa un solo robot móvil.

4.5. n robots móviles en la misma área de trabajo

De manera similar a antes, repetimos el proceso para n robots móviles ($n \leq 10$). No se muestra ninguna figura ya que las trayectorias no se visualizarían fácilmente debido a la cantidad de robots. Sin embargo, en el siguiente sección ofrecemos los resultados obtenidos para estos casos.

4.6. Presencia de caos en las trayectorias generadas para múltiples robots móviles

Siguiendo el procedimiento mostrado anteriormente en la sección 4.2.1, se analiza la cobertura y la presencia de caos para múltiples robots móviles.

Porcentaje de cobertura del área de trabajo

En la Tabla 4.12 se muestra el porcentaje de cobertura promedio obtenido para $1 \leq n \leq 10$ robots móviles. Las trayectorias se calculan utilizando 1000 condiciones iniciales diferentes para cada robot. Usamos 25 iteraciones del algoritmo para cada trayectoria, de manera que no se alcance la cobertura total (100 %).

Tabla 4.12 Porcentaje promedio de cobertura para n robots simultáneamente.

Numero de robots	Cobertura (%)
1	18.1250
2	33.7500
3	46.4583
4	55.8333
5	63.3333
6	70.2083
7	75.8333
8	79.3750
9	82.0833
10	85.8300

Se puede observar que el incremento de cobertura no sigue un comportamiento lineal. Esto se debe en parte al tamaño del robot que ocupa espacio en el área de trabajo y también podría implicar que existe un número óptimo de robots para un área de trabajo determinada. Planeamos investigar más a fondo este resultado como parte de nuestro trabajo futuro.

Prueba de Gottwald — Melbourne

Aplicamos la prueba de Gottwald - Melbourne para verificar la presencia de caos en las trayectorias generadas. Usamos 300 iteraciones y diferentes condiciones iniciales para cada trayectoria. La tabla 4.13 muestra los resultados promedio para x y y después de 1000 repeticiones, y cada uno es un promedio de los resultados del número de robots móviles utilizados en cada caso.

Tabla 4.13 Resultados de la prueba Gottwald-Melbourne para 10 robots simultáneamente.

Numero de robots	Resultados para x	Resultados para y
1	0.9837	0.9254
2	0.9461	0.9107
3	0.9540	0.9307
4	0.9863	0.9633
5	0.9554	0.9417
6	0.9899	0.9601
7	0.9919	0.9645
8	0.9621	0.9774
9	0.9619	0.9445
10	0.9769	0.9552

Por ejemplo, en caso de utilizar tres robots, el valor que se muestra para el eje x es un promedio del resultado de la prueba para los tres robots, 100 veces. Podemos concluir que, para $1 \leq n \leq 10$ robots móviles que se mueven simultáneamente en el área de trabajo, el caos está presente en sus trayectorias.

Prueba de Conteo de Cajas

Aplicando la prueba de Conteo de Cajas (ver sección 4.2.4) a $1 \leq n \leq 10$ robots móviles, se obtienen los promedios de dimension fraccionaria (1000 repeticiones para cada caso) que se muestran en la Tabla 4.14.

Tabla 4.14 Resultado de Conteo de Cajas para n robots simultáneamente.

Numero de robots	D
1	1.7762
2	1.7863
3	1.7634
4	1.7787
5	1.7723
6	1.7636
7	1.7864
8	1.7741
9	1.7679
10	1.7769

Podemos observar que una dimensión fractal está presente en las trayectorias generadas incluso con múltiples robots móviles simultáneos, la cual es una propiedad caótica. Con base en el resultado obtenido de las pruebas aplicadas, podemos concluir que toda trayectoria generada mediante el algoritmo propuesto presentado en este trabajo (algoritmo 1) es caótica.

4.7. Diferentes tamaños de área de trabajo

En esta sección se analiza el caso donde el área de trabajo es variable, siendo recorrido por $1 \leq n \leq 10$ robots móviles diferenciales.

En la tabla 4.15 se muestran los resultados de porcentaje de cobertura para 10 tamaños de área de trabajo (en mm^2). Se comienza con un área de $1000 mm^2$ y se realizan incrementos de $100 mm$ cada vez. Estos son los resultados de promediar los valores obtenidos en 100 repeticiones para cada caso, incrementando el número de robots móviles que se mueven en ella de manera simultánea. Cada trayectoria fue generada con 25 iteraciones del algoritmo 1.

Tabla 4.15 Porcentajes de cobertura para diferentes tamaños de área de trabajo (mm^2).

Área	1	2	3	4	5	6	7	8	9	10
1	21.11	38.88	52.22	62.22	71.11	77.77	82.22	87.77	91.11	94.44
2	18.18	32.72	44.54	54.54	62.72	69.09	74.54	80	83.63	87.27
3	15.15	28.03	39.39	47.72	55.30	62.12	67.42	71.96	76.51	79.54
4	13.46	25	33.97	42.94	49.35	55.12	60.89	65.38	69.87	73.07
5	11.53	21.42	30.21	37.91	43.95	50	55.21	59.34	63.18	67.03
6	10	19.04	26.66	33.80	39.76	45.23	50	53.33	58.09	60.95
7	9.16	17.08	23.75	30	35.41	40.83	45	49.16	52.91	55.83
8	8.08	15.07	21.32	26.83	32.35	37.13	40.80	44.85	48.16	51.10
9	7.18	13.72	19.28	24.83	29.08	33.33	37.58	40.84	44.11	46.89
10	6.43	12.28	17.54	22.22	26.60	30.70	34.21	37.71	40.35	43.27

En la figura 4.29 se muestra el comportamiento descrito por la tabla TABLA. Cada línea corresponde a un tamaño del área de trabajo.

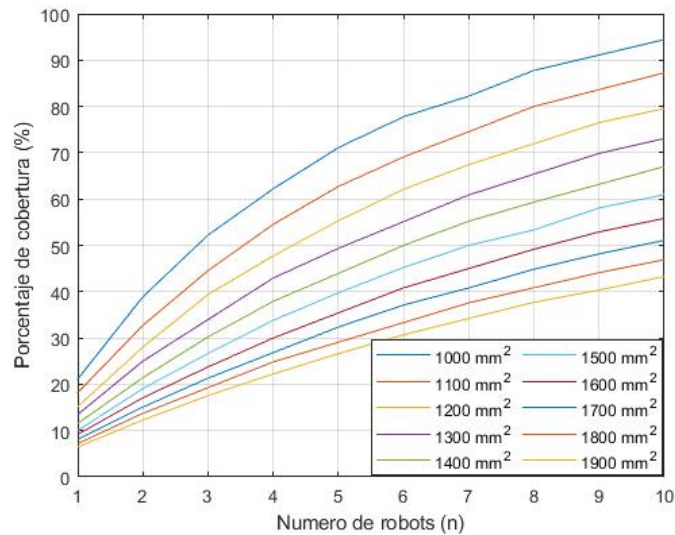


Figura 4.29 Porcentajes de cobertura para diferentes tamaños de área de trabajo.

Se puede observar que, para un área de cobertura menor, el impacto de incrementar la cantidad de robots es mayor; sin embargo se puede notar que, entre más robots se utilicen, es menor el incremento en la cobertura.

Por ejemplo, en el caso del área de 1000 mm^2 , incrementar el número de robots de 1 a 2 propicia un incremento de 17.77%; pero incrementar de 9 a 10 robots únicamente proporciona un incremento de 3.33%.

4.8. Conclusiones del capítulo

En este capítulo se propuso y describió un algoritmo para la generación de trayectorias caóticas en un robot móvil diferencial. Con ayuda del software Matlab R2015b®, se realizaron simulaciones numéricas que nos permitieron visualizar las propiedades del caos en dichas trayectorias. Se realizaron también algunas pruebas que nos permitieron determinar numéricamente la presencia de caos en ellas. Se concluyó que, efectivamente, las trayectorias obtenidas en diferentes condiciones iniciales son caóticas y que son capaces de cubrir prácticamente en su totalidad el área de trabajo propuesta, confirmando que las trayectorias del robot son impredecibles y, por tanto, son altamente atractivas para ser aplicadas a tareas de patrullaje.

Se realizaron simulaciones numéricas para el caso de la aplicación de búsqueda y detección de objetos, obteniendo resultados satisfactorios ya que el robot móvil diferencial es capaz de encontrar el objeto en cuestión en un plazo corto de tiempo.

Además, se analizaron los casos en los que múltiples robots móviles diferenciales fueron utilizados para cubrir el área de trabajo. Con los resultados obtenidos podemos concluir que las trayectorias generadas para cada robot móvil son caóticas. También podemos concluir, considerando los resultados obtenidos, que mas robots no equivale necesariamente a una mejor eficiencia de cobertura. Se debe realizar un análisis de costo-beneficio para poder determinar si es realmente conveniente utilizar cierta cantidad de robots móviles según sea el caso.

Capítulo 5

Implementación experimental

En este capítulo se presentan resultados experimentales de la implementación del algoritmo de generación de trayectorias caóticas a un robot móvil propuesto en este trabajo de tesis. Se habla en particular de la implementación del algoritmo en el robot móvil diferencial Khepera III. Además, se describe la ejecución experimental del algoritmo propuesto en un robot móvil aéreo no tripulado (cuadracóptero).

5.1. Resultados experimentales

Para la fase experimental, se necesita una manera de conocer la posición del robot móvil diferencial Khepera III en todo momento. En este trabajo de tesis se utilizaron las cámaras Flex 13 con el software Motive[®] (figura 5.1) de la marca Optitrack. Estas cámaras y software son especializados en la ubicación de marcadores reflectivos de infrarrojos en el espacio. Se utilizan en numerosas universidades y centros de investigación para diferentes áreas que necesitan estudio de movimientos como en robótica, movimiento del cuerpo humano, estudio de expresiones faciales, etc.

Se utilizó el programa MATLAB R2015b[®] para hacer los cálculos de los estados del mapa caótico de Hénon, calcular las velocidades lineal y angular y las velocidades de las ruedas izquierda y derecha. Esta información la envía al robot Khepera III por medio de bluetooth. La retroalimentación la recibe Matlab a través del sistema de cámaras. Esta información se gráfica en tiempo real para su presentación.

Se colocaron marcadores reflectivos en el robot Khepera III para conocer su posición en todo momento. Además, se colocaron marcadores en los extremos de la mesa de trabajo para

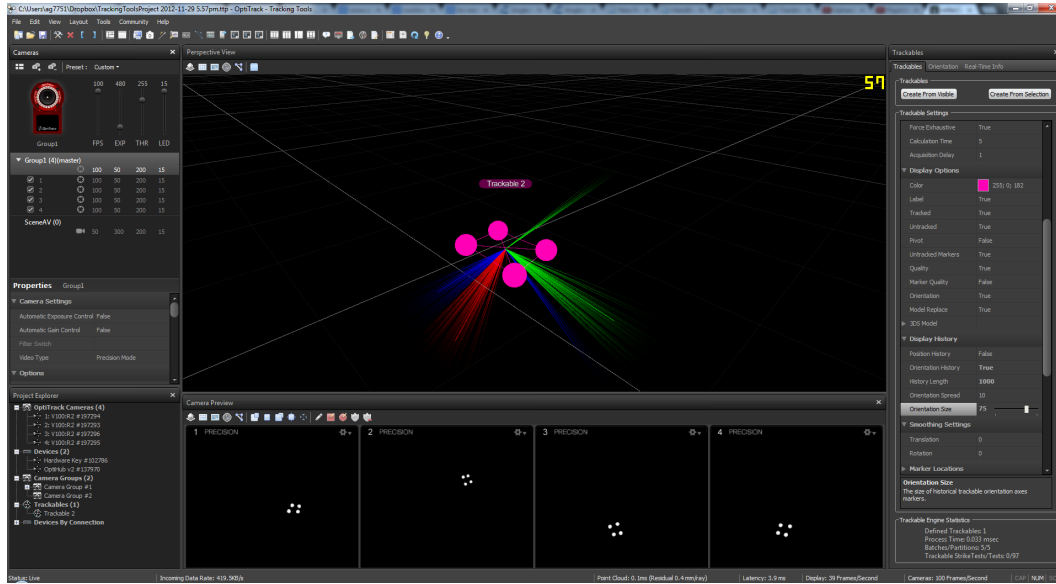


Figura 5.1 Software Motive®.

identificar los límites de la misma, representados por el cuadro negro visible en las figuras. Para la estrategia de movimiento reflectivo para la evasión de límites del área de trabajo, el programa detecta el momento en el que el robot diferencial se aproxima a los bordes de la mesa gracias a que las cámaras infrarrojas detectan los marcadores colocados en la misma y se toman acciones evasivas descritas en la sección 4.1. Gracias a esto, se obtuvieron gráficas en tiempo real de la trayectoria seguida por el robot móvil Khepera III al emplear el algoritmo generador de trayectorias caóticas descrito anteriormente.

Se realizaron tres experimentos con el robot diferencial Khepera III, utilizando diferentes condiciones iniciales (obtenidas directamente de la posición real del robot) con 100 iteraciones para cada uno de los siguientes casos:

- a) $x_0 = 1.1, y_0 = -0.9, \theta_0 = 1.8^\circ$,
- b) $x_0 = -0.5, y_0 = -1.3, \theta_0 = 359.1^\circ$,
- c) $x_0 = -31.4, y_0 = -254.8, \theta_0 = 238.6^\circ$.

Los resultados experimentales obtenidos se muestran en la figura 5.2. Se puede observar que un pequeño cambio en milímetros en las condiciones iniciales produce en el robot, trayectorias completamente diferentes e impredecibles, demostrando experimentalmente que la *propiedad*

caótica de sensibilidad a las condiciones iniciales se cumple. Se puede ver también, que con 100 iteraciones queda cubierta - por el patrullaje del robot Khepera III- la mayor parte del área de trabajo.

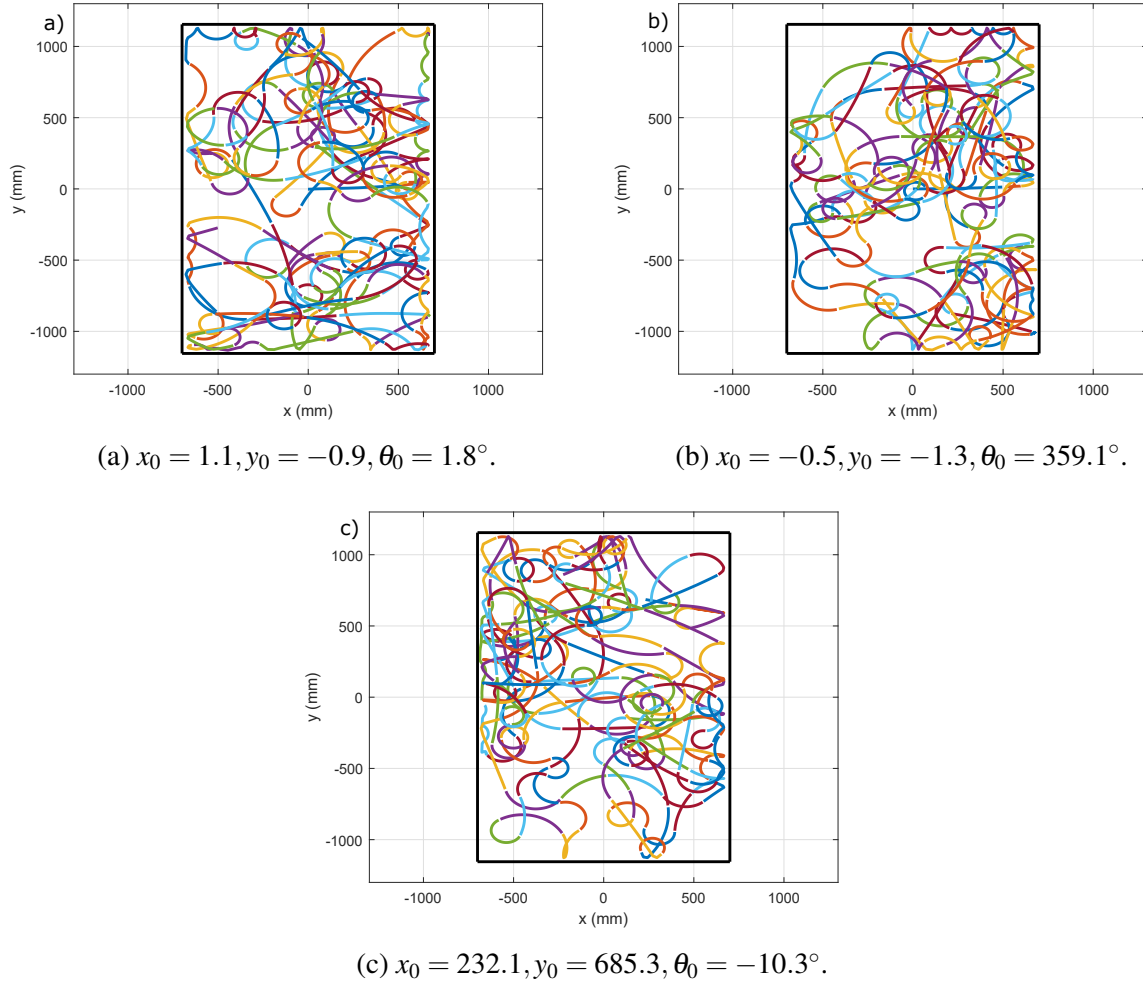


Figura 5.2 Resultados experimentales: 3 diferentes trayectorias generadas por 3 diferentes condiciones iniciales del robot Khepera III.

Cabe mencionar que el recorrido marcado por las líneas corresponde al camino seguido por el centro del robot. El cuadro negro que delimita el área de trabajo es visiblemente más pequeña que la mostrada en las figuras correspondientes a las simulaciones. Esto es debido a la anchura de los bordes de la mesa y a la distancia existente entre la periferia del robot Khepera III y el centro de sus ruedas.

En las siguientes subsecciones se reportan los resultados de las pruebas correspondientes para comprobar la presencia de caos en las trayectorias generadas por el algoritmo propuesto en este trabajo, en aplicaciones experimentales con datos obtenidos en tiempo real.

5.1.1. Porcentaje de cobertura experimental del área de trabajo

Para verificar el porcentaje de cobertura de las trayectorias caóticas del Khepera III, se repite el procedimiento utilizado en la sección 4.2.2, el cual consiste en dividir el área de trabajo en cuadros y midiendo la cantidad de éstos por los que el robot Khepera III pasa durante el recorrido de su trayectoria.

En la figura 5.3 se muestran los resultados obtenidos para los 3 recorridos del robot Khepera III, obtenidos experimentalmente mostrados en la sección anterior.

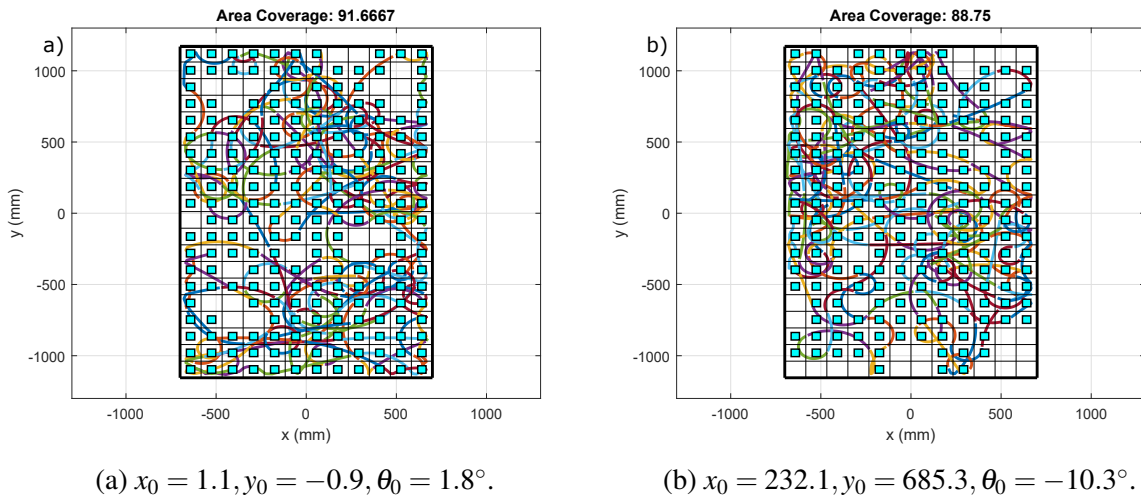


Figura 5.3 Porcentajes del área de trabajo cubiertas por trayectorias del robot Khepera III generadas con diferentes condiciones iniciales.

Podemos apreciar que, con tan solo 200 iteraciones del algoritmo generador de trayectorias caóticas, se cubre casi en su totalidad el área de prueba de manera experimental. Cabe recordar que no se toma en cuenta el área que ocupa el robot ni el alcance de sus sensores.

5.1.2. Prueba 0-1 Gottwald-Melbourne para resultados experimentales

Repitiendo el proceso seguido en la sección 4.2.3, utilizamos los valores obtenidos experimentalmente para realizar la prueba Gottwald-Melbourne para determinar la presencia de caos

en la trayectoria generada. En las figuras 5.4, 5.5 y 5.6 se muestran las series de tiempo de las coordenadas x y y para cada uno de los casos experimentales obtenidos.

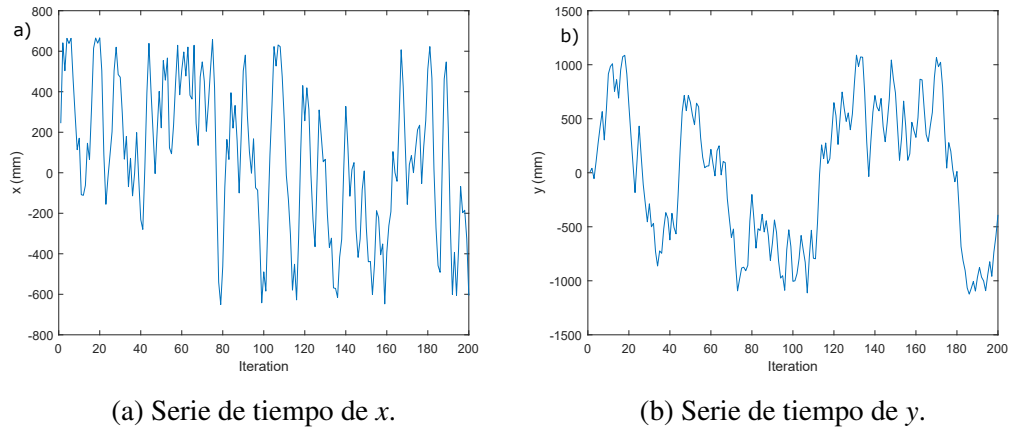


Figura 5.4 Series de tiempo resultantes con condiciones iniciales del inciso a).

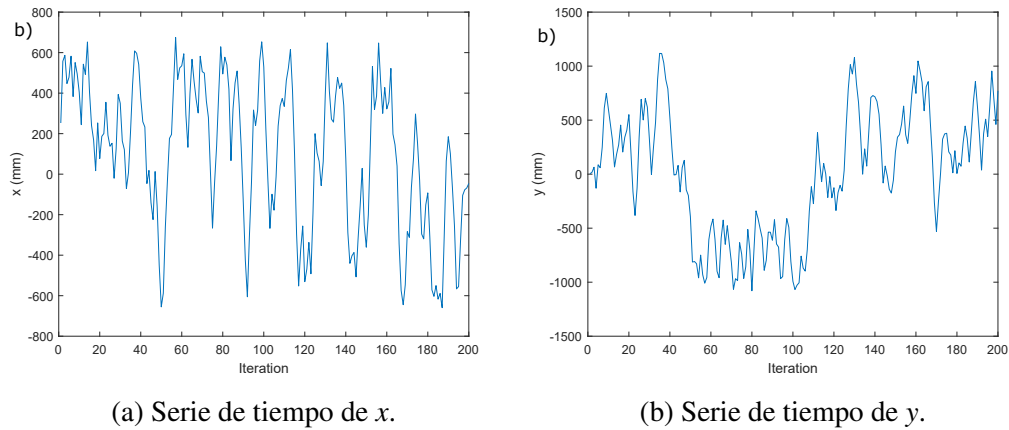


Figura 5.5 Series de tiempo resultantes con condiciones iniciales del inciso b).

En la tabla 5.1 se presentan los valores resultantes de la prueba. Se aprecia en estos resultados experimentales la presencia de *caos*.

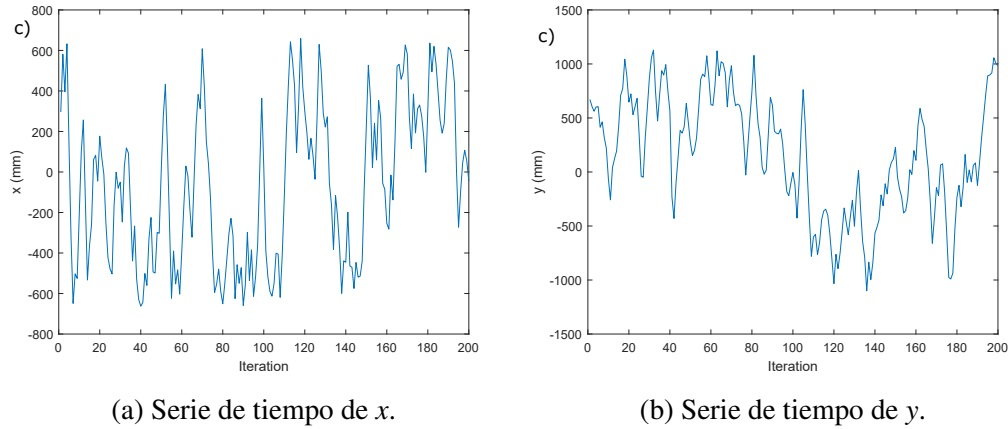


Figura 5.6 Series de tiempo resultantes con condiciones iniciales del inciso c).

Tabla 5.1 Resultados de la prueba 0-1 de Gottwald-Melbourne para resultados experimentales.

Resultados experimentales de la prueba 0-1 de Gottwald-Melbourne			
x_0	y_0	Resultado obtenido para x	Resultado obtenido para y
1.1	-0.9	0.9968	0.9703
-0.5	-1.3	0.9963	0.9665
232.1	685.3	0.9952	0.9430

5.1.3. Prueba de conteo de cajas ("Box-Counting") para resultados experimentales

Siguiendo el procedimiento descrito en la sección 4.2.4, tomamos en primer lugar el recorrido del robot Khepera III generado con las condiciones iniciales $x_0 = 1.1, y_0 = -0.9, \theta_0 = 1.8^\circ$. En la tabla 5.2 se muestran los valores medidos de N , l , $\log(N)$, $\log(\frac{1}{l})$ y D para este caso.

Al graficar $\log(N(s))$ contra $\log(\frac{1}{l})$ y realizar la regresión lineal a los datos mostrados obtenemos la figura 5.7. Se puede apreciar un comportamiento muy similar al de la recta dada por la ecuación $y = 1.7837x - 0.0462$.

Notamos que la pendiente de esta recta (1.7837) es muy similar a los valores de D obtenidos con la expresión (4.4). Lo anterior comprueba que la trayectoria tiene una *dimensión fraccional* y, por lo tanto, es *caótica*. En el caso de la trayectoria del robot móvil generada por las condiciones iniciales $x_0 = -0.5, y_0 = -1.3, \theta_0 = 359.1^\circ$ tomamos ahora 11 valores de l y medimos los

Tabla 5.2 Prueba de conteo de cajas para experimento de las condiciones iniciales del inciso a).

Datos experimentales para la prueba de conteo de cajas				
N	l	$\log(N)$	$\log(\frac{1}{l})$	D
194	0.0500	5.2679	2.9957	1.7585
274	0.0417	5.6131	3.1781	1.7662
369	0.0367	5.9108	3.3322	1.7738
469	0.0313	6.1506	3.4657	1.7747
579	0.0278	6.3613	3.5835	1.7752
694	0.0250	6.5425	3.6889	1.7736
796	0.0233	6.6796	3.7612	1.7759
935	0.0213	6.8405	3.8501	1.7767
1063	0.0196	6.9689	3.9318	1.7724
1200	0.0182	7.0901	4.0074	1.7693
1319	0.0169	7.1846	4.0775	1.7620

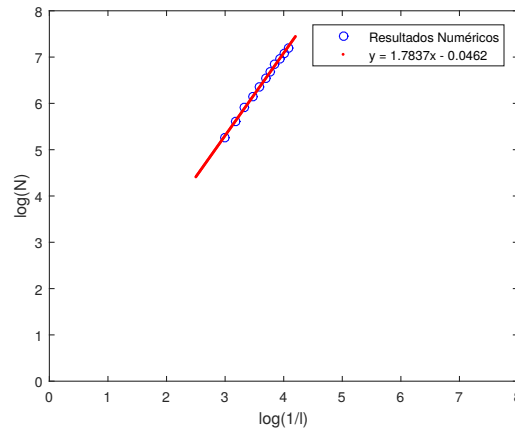


Figura 5.7 Resultado experimental de la prueba de conteo de cajas del inciso a).

valores correspondientes de $\log(N)$, $\log(\frac{1}{l})$ y D . Los valores obtenidos se muestran en la tabla 5.3.

La figura 5.8 nos muestra la gráfica de $\log(N(s))$ contra $\log(\frac{1}{l})$. Al realizar la regresión lineal a los datos mostrados obtenemos la ecuación de la recta $y = 1.7750x + 0.0037$, comprueba que la trayectoria tiene *dimensión fraccionaria* y, por lo tanto, es *caótica* también.

Tabla 5.3 Prueba de conteo de cajas para experimento de las condiciones iniciales del inciso b)).

Datos experimentales para la prueba Box-Counting				
N	l	$\log(N)$	$\log(\frac{1}{l})$	D
198	0.0500	5.2883	2.9957	1.7653
281	0.0417	5.6384	3.1781	1.7742
377	0.0367	5.9322	3.3322	1.7803
479	0.0313	6.1717	3.4657	1.7808
587	0.0278	6.3750	3.5835	1.7790
704	0.0250	6.5568	3.6889	1.7774
828	0.0233	6.7190	3.7612	1.7864
948	0.0213	6.8554	3.8501	1.7803
1073	0.0196	6.9782	3.9318	1.7748
1206	0.0182	7.0951	4.0074	1.7704
1347	0.0169	7.256	4.0775	1.7672

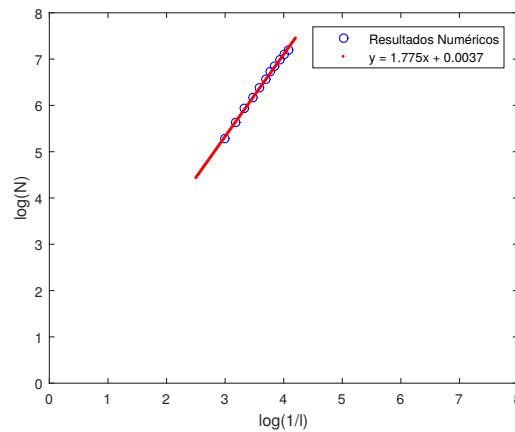


Figura 5.8 Resultado experimental de la prueba de conteo de cajas del inciso b).

Por último, realizamos la prueba para la trayectoria generada por las condiciones iniciales $x_0 = -31.4, y_0 = -254.8, \theta_0 = 238.6^\circ$. los datos obtenidos se muestran en la tabla 5.4

Finalmente, la gráfica de $\log(N(s))$ contra $\log(\frac{1}{l})$ se muestra en la figura 5.9. Al realizar la regresión lineal a los datos obtenemos la ecuación de la recta $y = 1.7975x - 0.0656$, lo cual comprueba que la *trayectoria es caótica* también.

Podemos afirmar que la prueba de conteo de cajas concluye que las trayectorias generadas por el algoritmo son efectivamente caóticas.

Tabla 5.4 Prueba de conteo de cajas para experimento de las condiciones iniciales del inciso c).

Datos experimentales para la prueba Box-Counting				
N	l	$\log(N)$	$\log(\frac{1}{l})$	D
199	0.0500	5.2933	2.9957	1.7669
284	0.0417	5.6490	3.1781	1.7775
377	0.0367	5.9322	3.3322	1.7803
479	0.0313	6.1717	3.4657	1.7808
592	0.0278	6.3835	3.5835	1.7814
708	0.0250	6.5624	3.6889	1.7790
829	0.0233	6.7202	3.7612	1.7867
970	0.0213	6.8774	3.8501	1.7862
1093	0.0196	6.9967	3.9318	1.7795
1249	0.0182	7.1301	4.0074	1.7793
1381	0.0169	7.2306	4.0775	1.7733

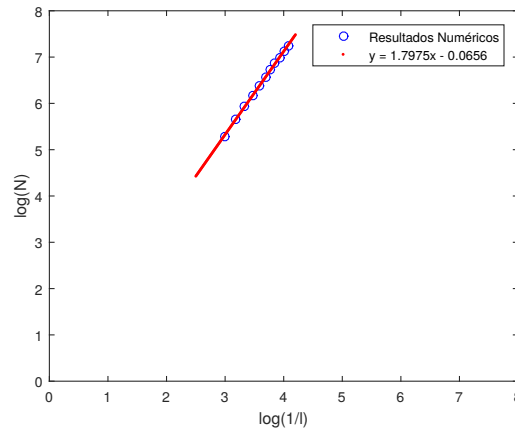


Figura 5.9 Resultado experimental de la prueba de conteo de cajas del inciso c).

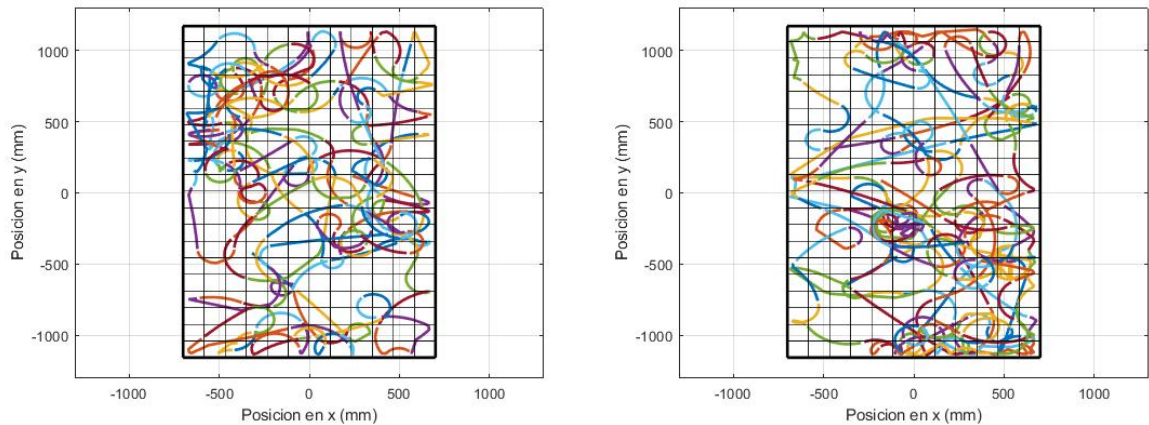
Es importante mencionar, que en las gráficas no se consideran las dimensiones reales del robot móvil diferencial Khepera III ni el alcance de sus sensores, por lo que en realidad el área cubierta por éste es mucho mayor que la mostrada.

5.2. Aplicación del algoritmo a dos robots Khepera III

Siguiendo la metodología de la sección anterior (5.1), se aplicó el algoritmo generador de trayectorias caóticas a dos robots móviles diferenciales Khepera III simultáneamente en el área de trabajo. Se realizaron dos experimentos con condiciones iniciales diferentes para cada uno, seleccionados de manera arbitraria.

- $x_{01} = 185.3, y_{01} = 147.7, \theta_{01} = 336.1^\circ, x_{02} = -317.4, y_{02} = -420.9, \theta_{02} = 313.3^\circ$.
- $x_{01} = 351.8, y_{01} = 612.7, \theta_{01} = 216.5^\circ, x_{02} = -398.3, y_{02} = -663.9, \theta_{02} = 34.9^\circ$.

En la figura 5.10 se muestran los resultados obtenidos para los tres casos. Nuevamente, cada sección de un color representa la trayectoria seguida por alguno de los robots móviles durante cada iteración del algoritmo. Cada trayectoria fue elaborada con 200 iteraciones del algoritmo propuesto.



(a) $x_{01} = 185.3, y_{01} = 147.7, \theta_{01} = 336.1^\circ, x_{02} = -317.4, y_{02} = -420.9, \theta_{02} = 313.3^\circ$.
 (b) $x_{01} = 351.8, y_{01} = 612.7, \theta_{01} = 216.5^\circ, x_{02} = -398.3, y_{02} = -663.9, \theta_{02} = 34.9^\circ$.

Figura 5.10 Resultados experimentales para 2 robots móviles simultáneos.

5.2.1. Porcentaje de cobertura

De la misma manera que en la sección 5.1.1, se calculó el porcentaje de área cubierta por las trayectorias generadas en ambos experimentos. La figura 5.11 muestra los resultados obtenidos.

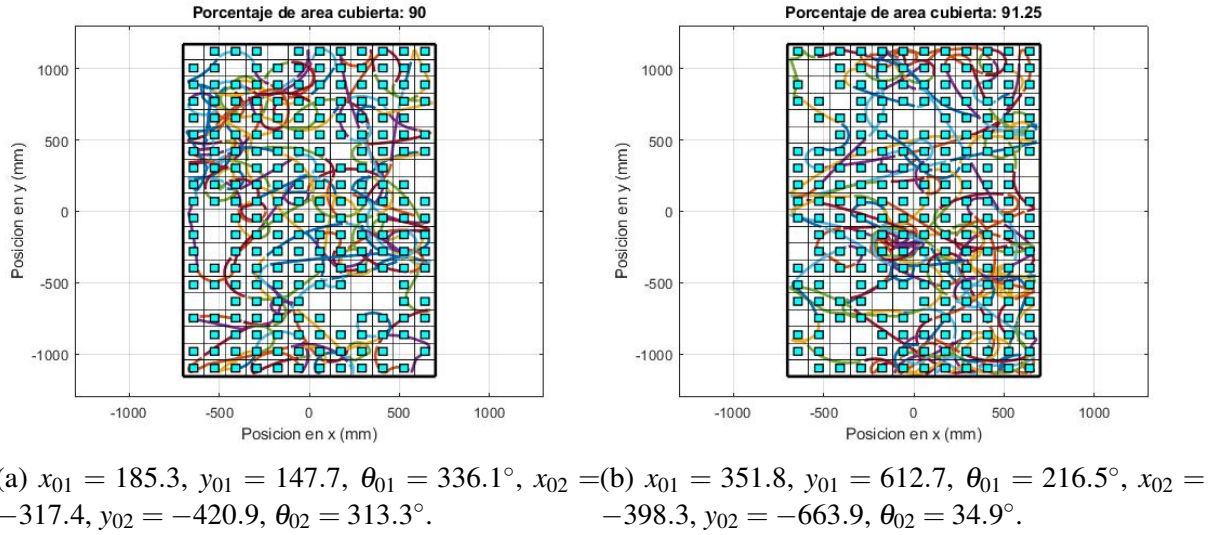


Figura 5.11 Resultados experimentales de cobertura para 2 robots móviles simultáneos.

Recordando que las trayectorias mostradas se generaron utilizando 200 iteraciones, podemos observar que los resultados de la cobertura de ambas trayectorias son similares a los reportados en la sección 4.3. Esto nos indica que los resultados experimentales cumplen con lo calculado en las simulaciones.

5.2.2. Conteo de Cajas

Siguiendo el procedimiento descrito en la sección 4.2.4, tomamos las trayectorias generadas para dos robots móviles simultáneos y calculamos sus dimensiones fraccionarias. Se realizaron 20 reducciones de cuadros para cada trayectoria. En la tabla 5.5 y 5.6 se muestran los valores medidos de ocho valores de N , l , $\log(N)$, $\log(\frac{1}{l})$ y D para cada caso.

El promedio de D para la tabla 5.5 es de 1.7748 y de la tabla 5.6 es de 1.7770. Estos resultados demuestran que ambas trayectorias son caóticas, y que cumplen muy cercanamente con los resultados obtenidos en las simulaciones.

5.2.3. Prueba 0-1 Gottwald-Melbourne

Repetiendo el proceso seguido en la sección 4.2.3, utilizamos los valores obtenidos experimentalmente para realizar la prueba Gottwald-Melbourne para determinar la presencia de caos

Tabla 5.5 Prueba de conteo de cajas para el primer experimento.

Datos experimentales para la prueba de conteo de cajas				
N	l	$\log(N)$	$\log(\frac{1}{l})$	D
216	0.0500	5.3753	2.9957	1.7943
272	0.0417	5.6058	3.1781	1.7639
349	0.0370	5.8551	3.2958	1.7765
381	0.0357	5.9428	3.3322	1.7834
421	0.0333	6.0426	3.4012	1.7766
462	0.0312	6.1356	3.4657	1.7703
469	0.0303	6.2066	3.4965	1.7751
572	0.0270	6.3491	3.6109	1.7583

Tabla 5.6 Prueba de conteo de cajas para el segundo experimento.

Datos experimentales para la prueba de conteo de cajas				
N	l	$\log(N)$	$\log(\frac{1}{l})$	D
219	0.0500	5.3891	2.9957	1.7989
274	0.0417	5.6131	3.1781	1.7662
359	0.0370	5.8833	3.2958	1.7851
386	0.0357	5.9558	3.3322	1.7874
424	0.0333	6.0497	3.4012	1.7787
450	0.0312	6.1092	3.4657	1.7628
499	0.0303	6.2126	3.4965	1.7768
577	0.0270	6.3578	3.6109	1.7607

en las trayectorias generadas. En la tabla 5.7 se muestran los resultados obtenidos los ejes x y y de ambos robots para las dos trayectorias.

Con estos resultados podemos concluir que, al utilizar dos robots móviles diferenciales Khepera III de manera experimental, las trayectorias generadas por el algoritmo propuesto también son caóticas

Tabla 5.7 Resultados de la prueba 0-1 para dos robots móviles.

Resultados experimentales de la prueba de Gottwald-Melbourne.			
x_0	y_0	Resultado obtenido para x	Resultado obtenido para y
185.3	147.7	0.9443	0.9128
-317.4	-420.9	0.9853	0.8808
351.8	612.7	0.9889	0.8902
-398.3	-663.9	0.9831	0.8889

5.3. Aplicación del algoritmo a un cuadracoptero

En esta sección se muestra el procedimiento utilizado y los resultados obtenidos para el caso en el que se utiliza un robot móvil volador (cuadracoptero) para demostrar que el algoritmo diseñado puede ser utilizado en otros tipos de robots móviles (además del diferencial).

5.3.1. Cuadracoptero

Los Vehículos aéreos no tripulados, más conocidos como drones (5.12), son aparatos de alta tecnología que funcionan gracias a unas hélices o rotores que les permite mantenerse y propulsarse en el aire. Para que el dron se mantenga estable, la mitad de estas hélices rotan en un sentido y la otra mitad lo hacen en sentido contrario (5.13). Cada una de las hélices se impulsa mediante un pequeño motor propulsado electrónicamente.



Figura 5.12 Dron.

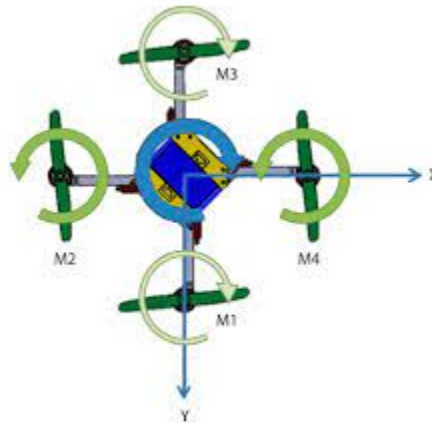


Figura 5.13 Sentido de rotación de las hélices de un dron.

Los drones se pilotan por control remoto y pueden realizar diferentes movimientos: pueden girar hacia la derecha o hacia la izquierda de su eje vertical y de su eje longitudinal (Yaw y Roll), pueden elevarse en vertical (Thrust) o rotar hacia delante o hacia atrás en relación a su eje transversal (Pitch) (5.14). Todos estos movimientos se controlan ajustando la propulsión de cada hélice y esto se hace con un aparato denominado controlador central de vuelo, cuya función principal es que el dron mantenga un vuelo lo más estable posible.

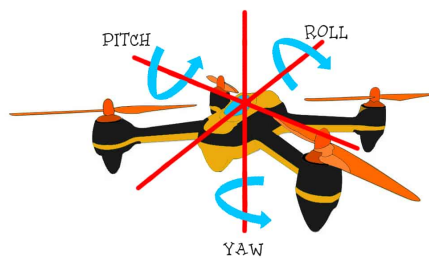


Figura 5.14 Ejes de movimiento de un dron.

El mando o transmisor de radio es el que permite dirigir el vuelo del dron y realizar ajustes a distancia. Normalmente se usa un aparato de control remoto pero actualmente también se puede pilotar un dron con un teléfono móvil. En algunos casos es posible programar el controlador central de vuelo para permitir un manejo autónomo del dron.

Existen diversos tipos de drones, los cuales se clasifican segun el numero de hélices. Un cuadraoptero es un caso particular de dron, cuyo vuelo es permitido gracias a cuatro motores que hacen girar cuatro hélices en total.

En la actualidad los vehículos aéreos no tripulados tienen muchos usos, ya que pueden cumplir funciones comerciales, militares e incluso recreativas. Por ejemplo, un dron puede usarse para realizar labores de búsqueda y rescate, de vigilancia, control del tráfico, para hacer mediciones y observaciones climatológicas, para la extinción de incendios o para el transporte de mercancías.

5.3.2. CrazyFlie

El Crazyflie (5.15) es un cuadraóptero del tamaño de la palma de la mano que pesa 27 gramos y admite control inalámbrico por radiofrecuencia y Bluetooth. Tiene un tiempo de vuelo de 7 minutos y un tiempo de carga de 40. Es una plataforma de desarrollo versátil que está diseñada para ayudar a las personas a que aprender sobre robótica y cuadraópteros. Ayuda adquirir conocimientos en áreas como algoritmos de control, sistemas integrados, aeronáutica o robótica verificando sus conocimientos en un cuadraóptero real. El diseño de hardware y código fuente está abierto y disponible para permitir la exploración y modificación.

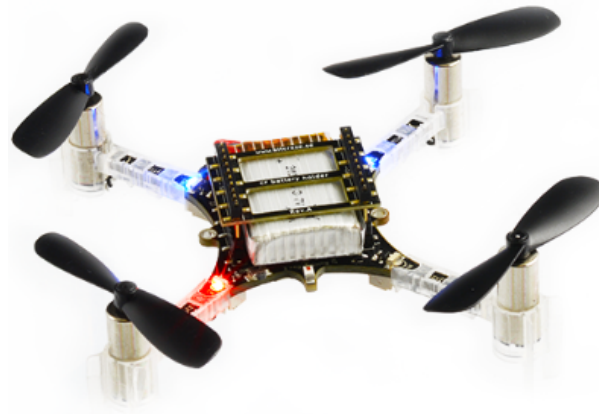


Figura 5.15 Crazyflie.

Está equipado con una interfaz expansible a la que se pueden conectar una variedad de plataformas de expansión. Dado que el diseño de la fuente y el hardware está abierto, también es posible agregar sus propios sensores o conectar otro hardware. Las plataformas de expansión

agregan una nueva funcionalidad a Crazyflie que amplía el alcance de las áreas de interés potenciales y hace que Crazyflie sea una gran herramienta para la educación donde tanto los profesores como los estudiantes pueden confiar en una plataforma de evaluación.

5.3.3. Ambiente de programación

EL firmware del Crazyflie está programado en C++ y permite ser modificado para poder realizar estudios y pruebas de controladores diseñados para cuadraópteros, así como modificar algunas de sus funciones para poder ser controlado de manera autónoma.

Ademas, cuenta con una maquina virtual con Ubuntu que contiene todas las librerías para poder modificar el firmware. Cuenta también con todo lo necesario para realizar la programación el Crazyflie utilizando el lenguaje Python 3, el cual es el que será utilizado en esta aplicación. Las librerías de Python permiten el control del Roll, Pitch, Yaw y Thrust, así como de su altura, permitiendo controlar la velocidad a la que el dron se mueve en ellos.

5.3.4. LOCO Positioning System

LOCO Positioning System es un sistema de posicionamiento local, basado en radio de banda ultra ancha que se utiliza para encontrar la posición absoluta en 3D de objetos en el espacio. Es en muchos aspectos similar a un sistema GPS en miniatura (5.16).

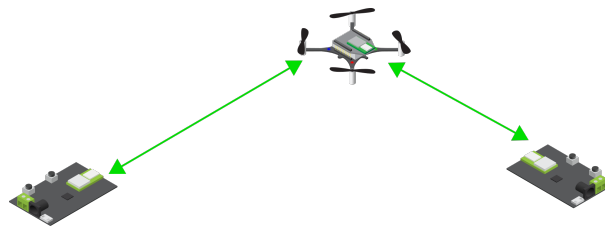


Figura 5.16 LOCO Positioning System.

La base del sistema es un conjunto de Anclas (Anchors, ver 5.17) que se colocan en la habitación y sirven de referencia (similar a los satélites de GPS). La otra parte del sistema es una o más Etiquetas (Tags, ver 5.18) (similar al receptor de GPS) que se fijan a los objetos que se van a localizar. Al enviar mensajes cortos de radio de alta frecuencia entre las Anclas y las Etiquetas, el sistema mide la distancia desde cada Ancla a las Etiquetas y calcula la posición de estas últimas a partir de esa información.

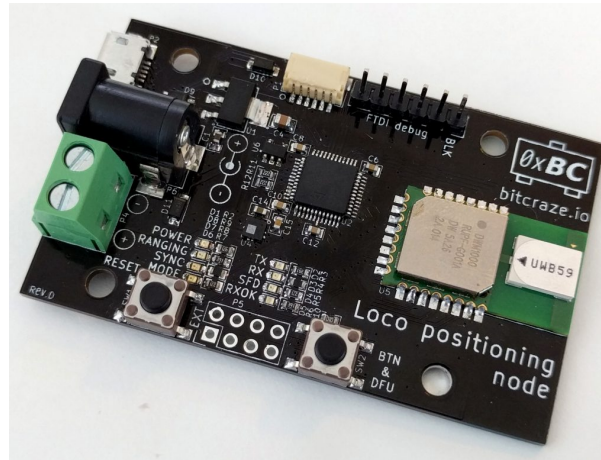


Figura 5.17 Ancla utilizada por LOCO Positioning System.



Figura 5.18 Etiqueta utilizada por LOCO Positioning System.

Toda la información necesaria para calcular la posición está disponible en la Etiqueta que permite la estimación de la posición a bordo del Crazyflie, a diferencia de muchos otros sistemas de posicionamiento donde la posición se calcula en una computadora externa y se envía al Crazyflie.

Al tener conocimiento de la posición del dron, es capaz de volarse de forma autónoma sin control manual. Esto abre una variedad de aplicaciones y casos de uso.

5.3.5. Metodología

En esta subsección se describirá el procedimiento utilizado para la generación de trayectorias caóticas en un cradracóptero Crazyflie, utilizando el algoritmo propuesto.

Sistema caótico MACM

El sistema caótico MACM CITA fue desarrollado en el CICESE por parte del equipo de trabajo del Laboratorio de Sincronización y Sistemas Complejos. Es un sistema sencillo que cuenta con 3 estados mostrados en la ecuación 5.1. Se ha demostrado exitosamente su uso en sistemas discretos CITA.

$$\begin{aligned}\dot{x} &= -ax - byz, \\ \dot{y} &= -x + cy, \\ \dot{z} &= d - y^2 - z.\end{aligned}\tag{5.1}$$

Este sistema tiene 7 términos, dos no-linealidades cuadráticas y cuatro parámetros. El sistema es caótico para $a = 2$, $b = 3$, $c = 0.5$ y $d = 4$. En la figura 5.19 se puede apreciar el atractor caótico en tres dimensiones formado por el sistema MACM.

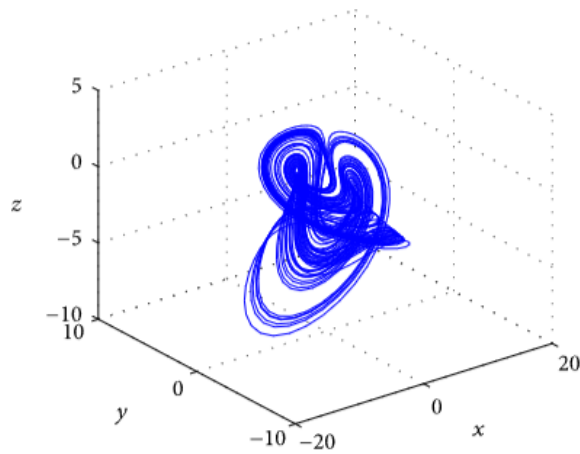


Figura 5.19 Atractor caótico en tres dimensiones del sistema MACM.

Algoritmo

Se realizaron algunas pequeñas modificaciones al algoritmo propuesto en este trabajo para su implementación experimental en un cuadracóptero Crazyflie. En primer lugar, en lugar de asignar los resultados obtenidos del mapa caótico utilizado a las velocidades de dos ruedas, se asignan a dos de los ejes de movimiento y uno al desplazamiento vertical, como se describió en la sección 5.3. Este es el principal motivo para utilizar el sistema caótico MACM en lugar del mapa de Hénon, ya que cuenta con tres estados. Estos se asignan a las velocidades de los ejes Roll, Pitch y al desplazamiento vertical Thrust respectivamente. El eje Yaw se mantiene fijo, de manera que el cuadracóptero no rota sobre si mismo, si no que su orientación permanece constante (mirando siempre al frente). En segundo lugar, se elimina el paso donde se despejan las velocidades independientes de las ruedas izquierda y derecha ya que no es necesario. El resto del algoritmo se mantiene igual.

El área de trabajo es un cubo libre de obstáculos de $800 \times 800 \times 1100$ mm, y se aplica la misma estrategia reflectiva mostrada en la sección 4.1, pero para tres dimensiones.

Trayectoria generada

En la figura 5.20 se muestran los resultados obtenidos al aplicar el algoritmo ligeramente modificado al cuadracóptero Crazyflie. La línea negra denota los límites de área de trabajo, la línea azul indica la trayectoria seguida por el dron y los círculos rojos muestran las lecturas de posición realizadas por el sistema de localización (sección 5.3.4). Cabe mencionar que la altura del área de trabajo no comienza al nivel del suelo, si no que a una altura conveniente para que el dron despegue y pueda mantenerse estable en vuelo. Los puntos fuera del área de trabajo son las posiciones del cuadracóptero al despegar.

Para la trayectoria mostrada se utilizaron 100 iteraciones del algoritmo. Se puede apreciar de manera visual el comportamiento impredecible que la trayectoria describe. En la siguiente subsección se analizará la presencia de caos en ésta trayectoria.

5.3.6. Presencia de Caos en la trayectoria generada para un cuadcoptero

En esta sección se mostraran los resultados obtenidos al realizar las pruebas de porcentaje de cobertura del área de trabajo, de conteo de cajas y de Gottwald-Melbourne (aplicadas previamente en los diferentes casos presentados con el robot móvil diferencial Khepera 3) a la trayectoria generada por el algoritmo propuesto en un cuadracóptero Crazyflie.

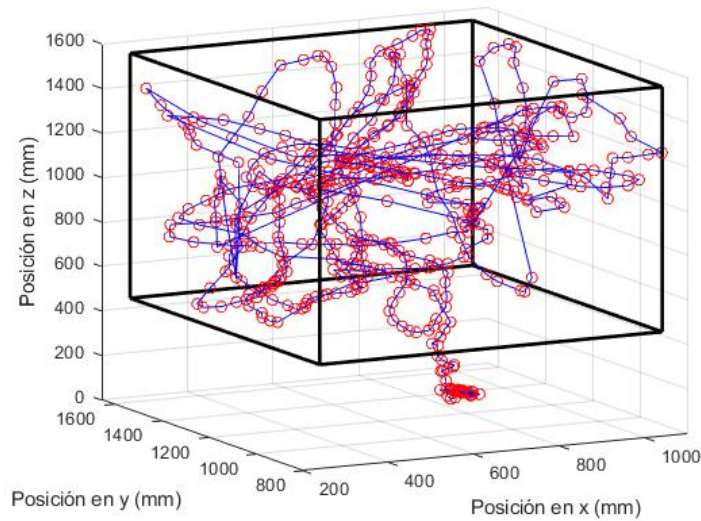


Figura 5.20 Trayectoria generada en un cuadracóptero Crazyflie.

Porcentaje de cobertura

Para determinar el porcentaje de cobertura del área de trabajo, se dividió en $5 \times 5 \times 5$ cubos de 160 mm por lado cada uno, se anotó la cantidad de cuadros visitados por la trayectoria generada y se calculó el porcentaje (similar a lo realizado en secciones anteriores). Este tamaño de cuadro fue elegido debido su similitud con las dimensiones del Crazyflie. En la figura 5.21 se muestra el resultado obtenido por este experimento.

Los cubos azules denotan los cubos visitados por la trayectoria. Es importante notar que estos indican los cuadros visitados por las mediciones de posición realizadas por LOCO Positioning System y no considera la trayectoria recorrida entre mediciones.

Con un total de 125 cubos que conforman el área de trabajo, y 88 de ellos visitados, se tiene un porcentaje de cobertura de 70.4% . Cabe mencionar que, al ser un espacio en 3 dimensiones, es difícil obtener un porcentaje de cobertura cercano al 100% .

Conteo de Cajas

Para aplicar la prueba de Conteo de Cajas a la trayectoria generada con el cuadracoptero, se sigue un procedimiento similar al de la sección 4.2.4. Se toman los cubos generados en la sección anterior (5.3.6) y se va reduciendo su tamaño. Se toma nota de la longitud de los cubos

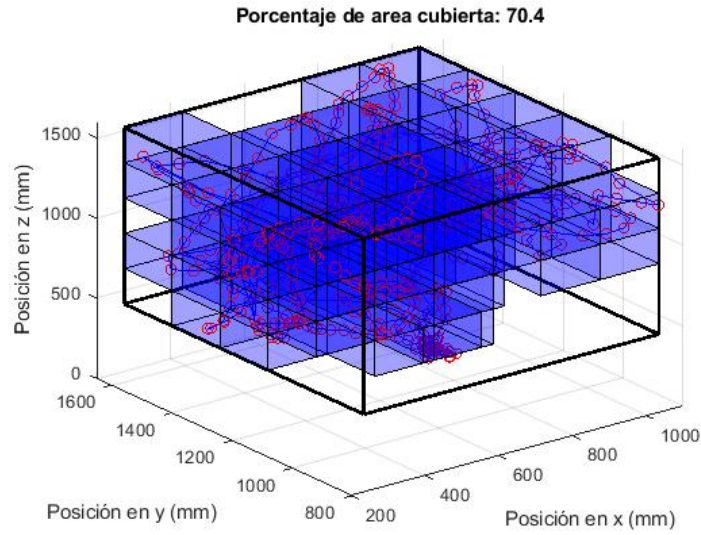


Figura 5.21 Porcentaje de cobertura del área de trabajo por la trayectoria generada por el cuadracóptero Crazyflie.

y de la cantidad de cubos visitados. En este caso se realizan diez reducciones de cuadros. En la figura 5.22 se muestran cuatro de estas reducciones de manera visual.

En la tabla 5.8 se muestran los resultados del conteo de cajas obtenidos para las diez reducciones de cubos.

Tabla 5.8 Prueba de conteo de cajas para la trayectoria del Crazyflie.

Porcentaje de cobertura del área de trabajo.				
N	l	$\log(N)$	$\log(\frac{1}{l})$	D
127	0.166	4.8442	1.7918	2.7036
167	0.143	5.1180	1.9459	2.6301
207	0.125	5.3327	2.0794	2.5645
254	0.111	5.5013	2.1972	2.5037
362	0.1	5.5683	2.3026	2.4183
304	0.091	5.7170	2.3979	2.3842
320	0.083	5.7683	2.4849	2.3213
352	0.077	5.8636	2.5649	2.2861
368	0.071	5.9081	2.6391	2.2387
373	0.0667	5.9081	2.7081	2.1851
377	0.0625	5.9322	2.7726	2.1396

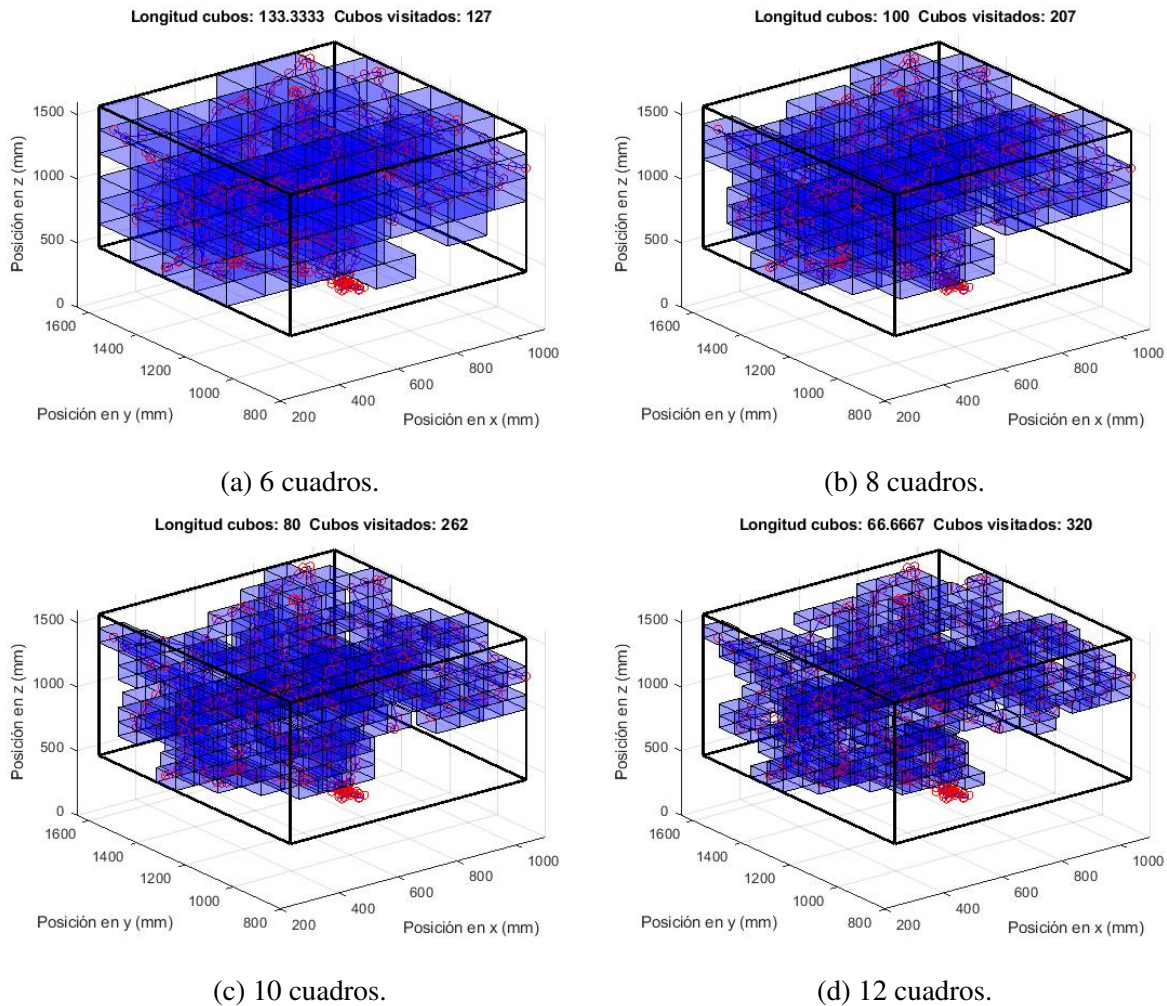


Figura 5.22 Diferentes tamaños de cubo para el conteo de cajas de la trayectoria generada.

Estos resultados no son los deseados para esta prueba. Se esperaría que las dimensiones fraccionarias fuesen mas similares para cada caso. Esto se debe principalmente a que en muchos casos, la distancia entre dos muestras continuas tomadas por el sistema de posicionamiento es muy grande, dejando espacios vacíos entre una y otra. Al ir reduciendo el tamaño de los cubos, no se cubre la totalidad de la trayectoria generada (es decir, se pierden cubos).

Sin embargo, se puede observar también que en ninguno de los casos la dimension fraccionaria es entera, lo cual nos da un indicio de que la trayectoria generada es caótica.

Prueba 0-1 Gottwald-Melbourne

Para la prueba 0-1 de Gottwald-Melbourne se utilizan las series de tiempo generadas para los ejes x , y , y z . En la tabla 5.9 se muestran los resultados obtenidos para el cuadracóptero.

Tabla 5.9 Resultados de la prueba 0-1 de Gottwald-Melbourne para la trayectoria del Crazyflie.

Resultados experimentales de la prueba 0-1 de Gottwald-Melbourne		
Resultado obtenido para x	Resultado obtenido para y	Resultado obtenido para z
0.9391	0.8804	0.8492

Con estos resultados podemos concluir que la trayectoria generada por el algoritmo propuesto, aplicada a un cuadracóptero, es caótica.

5.4. Conclusiones del capítulo

En este capítulo se reportaron los resultados experimentales obtenidos por la implementación del algoritmo de generación de trayectorias caóticas propuesto en este trabajo de tesis. Se utilizó un robot móvil diferencial Khepera III y la mesa de trabajo descrita en la sección 4.1. Se presentaron las trayectorias caóticas generadas para diferentes condiciones iniciales del robot móvil Khepera III y se realizaron pruebas descritas en el capítulo anterior para determinar la presencia de caos en ellas. Se llegó a la conclusión de que las trayectorias generadas con el algoritmo propuesto para el robot Khepera III son caóticas, cumpliendo el objetivo de este trabajo de tesis.

Cabe mencionar que estos resultados experimentales son preliminares, es necesario mejorar el algoritmo de evasión de bordes para que el ángulo de reflexión sea más preciso y utilizar otros sistemas caóticos para comparar su desempeño. De igual manera, es necesario ajustar algunos detalles del área de trabajo y del programa generador de trayectorias caóticas. Sin embargo, los resultados experimentales obtenidos son satisfactorios y congruentes con los numéricos. Entre las condiciones que afectan los resultados experimentales se encuentran las limitaciones físicas intrínsecas del robot Khepera III, así como algunas características del área de trabajo.

Se implementó también, de manera experimental, el algoritmo a dos robots móviles diferenciales Khepera III simultáneamente en el área de trabajo, concluyendo que las trayectorias

generadas son caóticas, y que los resultados experimentales son bastante similares a los obtenidos en las simulaciones.

Se analizó también el caso en el que el algoritmo propuesto es aplicado a un cuadracóptero Crazyflie para generar en él trayectorias caóticas, concluyendo que es aplicable para mas de un tipo de robot móvil, incluyendo drones. También se demostró que la trayectoria generada es, ciertamente, caótica. Sin embargo, es necesario realizar estos experimentos de manera mas rigurosa, reduciendo el tiempo de muestreo de posición del sistema de posicionamiento e incrementando el número de iteraciones realizadas.

Capítulo 6

Conclusiones

En este capítulo final se presentan las conclusiones acerca del presente trabajo de tesis realizado, así como una mención del trabajo a futuro que se genera a partir de los resultados obtenidos.

En este trabajo de tesis se empleó el mapa caótico de Hénon para inducir comportamientos caóticos en las velocidades lineales y angulares de un modelo matemático de un robot móvil diferencial, con aplicaciones de patrullaje, búsqueda y detección de objetos en un área de trabajo, aprovechando las propiedades del caos.

Como contribución de esta tesis, se diseñó un algoritmo que produce una trayectoria caótica en un robot diferencial, tanto en simulaciones numéricas como en aplicación práctica empleando un robot móvil tipo Khepera III y se realizaron simulaciones y experimentos prácticos que demuestran su validez.

Las trayectorias generadas cumplen los objetivos de ser *impredecibles* para los observadores externos y cumplen la misión de patrullar un área de trabajo de *manera eficiente*. Además, se demostró que dichas trayectorias son altamente sensibles a las condiciones iniciales y que tienen un comportamiento determinista. Todas estas características son propiedades intrínsecas de los sistemas caóticos.

Se realizaron pruebas numéricas para determinar la existencia de caos en las trayectorias generadas en el robot móvil Khepera III. La prueba de Gottwald-Melbourne concluye que las series de tiempo obtenidas son efectivamente caóticas; mientras que la prueba de conteo de cajas muestra que la trayectoria final como tal, también lo es.

Se realizó una aplicación de búsqueda de objetos en el área de trabajo para diferentes casos, y se comprobó la eficacia de las trayectorias caóticas para esta tarea. También, con

las pruebas de cobertura se demuestra su utilidad para tareas de patrullaje. Con este fin, se implementó el algoritmo a múltiples robots móviles con diferentes configuraciones y se analizó el comportamiento obtenido para este caso.

Por último, el algoritmo generador de trayectorias caóticas se aplicó exitosamente para los casos de uno y dos robots móviles diferenciales Khepera III y para un cuadricóptero Crazyfly, demostrando numéricamente la presencia de caos en las trayectorias generadas.

Los resultados experimentales están en concordancia con los resultados numéricos, lo que implica que la metodología aplicada se encuentra bien encaminada.

Con los resultados numéricos y experimentales obtenidos y mostrados durante esta tesis, podemos concluir que los objetivos planteados se cumplieron en su totalidad y de manera exitosa. Además, se deja una base sólida para reproducir los resultados obtenidos y su posible aplicación en otros trabajos; con aplicaciones variadas como lo son: patrullaje, búsqueda de objetos peligrosos, rescate, exploración de terrenos desconocidos.

6.1. Trabajo futuro

Este trabajo de tesis se enfocó en el diseño de un algoritmo capaz de generar trayectorias caóticas en robots móviles diferenciales y en el análisis de dichas trayectorias para determinar la presencia de caos en ellas. Además, se realizó una aplicación de búsqueda de objetos por el robot móvil en un área de trabajo. Sin embargo, existen muchas otras tareas por realizar para profundizar en el tópico de este trabajo de tesis:

- Mejorar el algoritmo de evasión de bordes con ángulo de reflexión.
- Emplear otros sistemas caóticos en el algoritmo.
- Estudio e implementación en otras aplicaciones prácticas de las trayectorias caóticas del robot móvil diferencial, como búsqueda y rescate, exploración con obstáculos en el área de trabajo, etc.
- Implementación de otras pruebas para determinar de manera más rigurosa la presencia de caos en las trayectorias generadas.
- Evasión de obstáculos en el área de trabajo.

-
- Realizar trayectorias mas largas y con menor tiempo de muestreo para el cuadracóptero Crazyflie.
 - Analizar la existencia de condiciones iniciales optimas que permitan una alta cobertura del área de trabajo con pocas iteraciones para uno o varios robots móviles.
 - Aplicación en otros tipos de robots móviles.

Bibliografía

- [1] R. A. Española and E. Madrid, *Diccionario de la lengua española*, vol. 22. Real academia española Madrid, 2001.
- [2] C. Cruz-Hernández, “Synchronization of time-delay Chua’s oscillator with application to secure communication,” *Nonlinear Dynamics and Systems Theory*, vol. 4, no. 1, pp. 1–13, 2004.
- [3] C. Cruz-Hernández and N. Romero-Haros, “Communicating via synchronized time-delay Chua’s circuits,” *Communications in Nonlinear Science and Numerical Simulation*, vol. 13, no. 3, pp. 645–659, 2008.
- [4] G. Lazea and A. E. Lupu, “Aspects on path planning for mobile robots,” *TEMPUS M-JEP 11467: Intensive Course on Computer Aided Engineering in Flexible Manufacturing*, pp. 19–23, 1997.
- [5] L. Gámez-Guzmán, C. Cruz-Hernández, R. López-Gutiérrez, and E. García-Guerrero, “Synchronization of Chua’s circuits with multi-scroll attractors: application to communication,” *Communications in Nonlinear Science and Numerical Simulation*, vol. 14, no. 6, pp. 2765–2775, 2009.
- [6] L. Cardoza-Avendaño, R. López-Gutiérrez, C. Cruz-Hernández, V. Spirin, R. Chávez-Pérez, and A. Arellano-Delgado, “Encrypted audio transmission via synchronized chaotic Nd: Yag lasers,” *Revista Mexicana de Física*, vol. 58, no. 6, pp. 472–480, 2012.
- [7] H. K. Khalil and J. Grizzle, *Nonlinear systems*, vol. 3. Prentice Hall New Jersey, 1996.
- [8] L. Li, H. Peng, J. Kurths, Y. Yang, and H. J. Schellnhuber, “Chaos–order transition in foraging behavior of ants,” *Proceedings of the National Academy of Sciences*, vol. 111, no. 23, pp. 8392–8397, 2014.
- [9] O. Miramontes Vidal, “Orden y caos en la organización social de las hormigas,” *Departamento de Sistemas Complejos. Instituto de Física, UNAM*, no. 059, 2000.
- [10] U. Nehmzow and K. Walker, “The behaviour of a mobile robot is chaotic,” *AISB Journal*, vol. 1, no. 4, pp. 373–388, 2003.
- [11] Y. Nakamura and A. Sekiguchi, “The chaotic mobile robot,” *IEEE Transactions on Robotics and Automation*, vol. 17, no. 6, pp. 898–904, 2001.

- [12] Y. Bae, M. Lee, and T. M. Gatton, "An obstacle avoidance method for chaotic robots using angular degree limitations," in *International Conference on Computational Science and Its Applications*, pp. 244–250, Springer, 2006.
- [13] C. K. Volos, I. M. Kyprianidis, and I. N. Stouboulos, "A chaotic path planning generator for autonomous mobile robots," *Robotics and Autonomous Systems*, vol. 60, no. 4, pp. 651–656, 2012.
- [14] L. S. Martins-Filho, E. E. Macau, R. Rocha, R. F. Machado, and L. A. Hirano, "Kinematic control of mobile robots to produce chaotic trajectories," in *Proc. of the 18th Int. Congress of Mechanical Engineering, Ouro Preto*, 2005.
- [15] L. S. Martins-Filho and E. E. Macau, "Patrol mobile robots and chaotic trajectories," *Mathematical Problems in Engineering*, vol. 2007, 2007.
- [16] L. S. Martins-Filho, R. F. Machado, R. Rocha, and V. Vale, "Commanding mobile robots with chaos," in *ABCM Symposium Series in Mechatronics*, vol. 1, pp. 40–46, 2004.
- [17] D.-I. Curiac and C. Volosencu, "A 2d chaotic path planning for mobile robots accomplishing boundary surveillance missions in adversarial conditions," *Communications in Nonlinear Science and Numerical Simulation*, vol. 19, no. 10, pp. 3617–3627, 2014.
- [18] C. Li, F. Wang, L. Zhao, Y. Li, and Y. Song, "An improved chaotic motion path planner for autonomous mobile robots based on a logistic map," *International Journal of Advanced Robotic Systems*, vol. 10, no. 273, 2013.
- [19] J.-H. Lin, L.-R. Huang, *et al.*, "Chaotic bee swarm optimization algorithm for path planning of mobile robots," in *Proceedings of the 10th WSEAS international conference on evolutionary computing*, pp. 84–89, World Scientific and Engineering Academy and Society (WSEAS), 2009.
- [20] J. Pliego-Jiménez, R. Martínez-Clark, C. Cruz-Hernández, and A. Arellano-Delgado, "Trajectory tracking of wheeled mobile robots using only cartesian position measurements," *Automatica*, vol. 113, 2021.
- [21] R. Martínez-Clark, C. Cruz-Hernández, J. Pliego-Jiménez, and A. Arellano-Delgado, "Control algorithms for the emergence of self-organized behaviors in swarms of differential-traction wheeled mobile robots," *International Journal of Advanced Robotic Systems*, vol. 15(6), 2018.
- [22] S. Nasr, H. Mekki, and K. Bouallegue, "A multi-scroll chaotic system for a higher coverage path planning of a mobile robot using flatness controller," *Chaos, Solitons and Fractals*, vol. 118, 2018.
- [23] A. I. Majeed, "Mobile robot motion control based on chaotic trajectory generation," *Journal of Engineering and Sustainable Development*, vol. 24(2), 2019.

- [24] E. Petavratzis, C. Volos, L. Moysis, and I. Stouboulos, "An inverse pheromone approach in a chaotic mobile robot's path planning based on a modified logistic map," *Technologies*, vol. 7(4), 2019.
- [25] F. C. Moon, *Chaotic and Fractal Dynamics: Introduction for Applied Scientists and Engineers*. John Wiley & Sons, 2008.
- [26] G. P. Williams, *Chaos Theory Tamed*. Joseph Henry Press, 1997.
- [27] J. Gleick, *Caos: la creación de una ciencia*. Seix Barral, 1988.
- [28] C. S. Bertuglia and F. Vaio, *Nonlinearity, chaos, and complexity: the dynamics of natural and social systems*. Oxford University Press on Demand, 2005.
- [29] R. C. Hilborn, S. Coppersmith, A. J. Mallinckrodt, S. McKay, *et al.*, "Chaos and nonlinear dynamics: an introduction for scientists and engineers," *Computers in Physics*, vol. 8, no. 6, pp. 689–689, 1994.
- [30] R. Martínez-Clark, "Aplicaciones del caos.," *CICESE*, 2016.
- [31] M. Hénon, "A two-dimensional mapping with a strange attractor," *Communications in Mathematical Physics*, vol. 50, no. 1, pp. 69–77, 1976.
- [32] N. Torres, "Caos en sistemas biológicos.," *Matematicalia: Revista Digital de Divulgación Matemática de la Real Sociedad Matemática Española*, vol. 1, no. 3, 2005.
- [33] M. F. Barnsley, *Fractals everywhere*. Academic Press, 2014.
- [34] S. Días Brecia, "Primeras reflexiones de los fractales, la teoría del caos y su aplicación en el aula," *Revista Red Visual*, pp. 9–10, 2009.
- [35] F. Hausdorff, "Dimension und äußeres maß," *Mathematische Annalen*, vol. 79, no. 1-2, pp. 157–179, 1919.
- [36] A. Besicovitch, "On linear sets of points of fractional dimension," *Mathematische Annalen*, vol. 101, no. 1, pp. 161–193, 1929.
- [37] Z. Liu, X. Zhu, W. Hu, and F. Jiang, "Principles of chaotic signal radar," *International Journal of Bifurcation and Chaos*, vol. 17, no. 05, pp. 1735–1739, 2007.
- [38] F.-Y. Lin and J.-M. Liu, "Chaotic radar using nonlinear laser dynamics," *IEEE Journal of Quantum Electronics*, vol. 40, no. 6, pp. 815–820, 2004.
- [39] A. Zhu and H. Leung, "Cooperation random mobile robots based on chaos synchronization," in *2007 IEEE International Conference on Mechatronics*, pp. 1–5, IEEE, 2007.
- [40] R. Trejo-Guerra, E. Tlelo-Cuautle, C. Cruz-Hernández, and C. Sánchez-López, "Chaotic communication system using Chua's oscillators realized with CCII+ s," *International Journal of Bifurcation and Chaos*, vol. 19, no. 12, pp. 4217–4226, 2009.

- [41] S. Stavrinides, A. Anagnostopoulos, A. Miliou, A. Valaristos, L. Magafas, K. Kosmatopoulos, and S. Papaioannou, "Digital chaotic synchronized communication system," *J. Eng. Sci. Techn. Rev.*, vol. 2, pp. 82–86, 2009.
- [42] K. M. Cuomo, A. V. Oppenheim, and S. H. Strogatz, "Synchronization of Lorenz-based chaotic circuits with applications to communications," *IEEE Transactions on circuits and systems II: Analog and digital signal processing*, vol. 40, no. 10, pp. 626–633, 1993.
- [43] R. Batra, P. Zade, and D. Sagne, "Design and implementation of Sierpinski Carpet fractal antenna for wireless communication," *International Journal of Scientific Research Engineering & Technology (IJSRET)*, vol. 1, pp. 043–047, 2012.
- [44] R. Kelly and V. Santibáñez, "Control de movimiento de robots manipuladores," *Pearson educación*, 2003.
- [45] K. Čapek and I. Kallinikov, "Rossum's universal robots," *Fr. Borový*, 1940.
- [46] K. Cleary and C. Nguyen, "State of the art in surgical robotics: clinical applications and technology challenges," *Computer Aided Surgery*, 2001.
- [47] E. Appleton and W. D. J., "Industrial robot applications," *Springer Science and Business Media*, 2012.
- [48] J. A. N. Cocota, H. S. Fujita, and J. d. S. Itamar, "A low-cost robot manipulator for education," *Technologies Applied to Electronics Teaching (TAEET). IEEE*, 2012.
- [49] J. Griffith, V. K., and M. Cosad, "Long term autonomous ocean remote sensing utilizing the wave glider," *Proceedings of the AGU Fall Meeting*, 2012.
- [50] C. Chung and M. Nakashima, "Swimming humanoid robot "swumanoid" as an experimental platform for research of human swimming," *Journal of Robotics and Mechatronics*, 2014.
- [51] P. Radoglou-Grammatikis, P. Sarigiannidis, T. Lagkas, and I. Moscholios, "A compilation of uav applications for precision agriculture," *Computer Networks*, vol. 172, p. 107148, 2020.
- [52] C. D. Drummond, M. D. Harley, I. L. Turner, A. N. A. Matheen, W. C. Glamore, *et al.*, "Uav applications to coastal engineering," in *Australasian Coasts & Ports Conference*, vol. 2015, p. 22nd, 2015.
- [53] A. Restas *et al.*, "Drone applications for supporting disaster management," *World Journal of Engineering and Technology*, vol. 3, no. 03, p. 316, 2015.
- [54] C. D. Bellicoso, M. Bjelonic, L. Wellhausen, K. Holtmann, F. Günther, M. Tranzatto, P. Fankhauser, and M. Hutter, "Advances in real-world applications for legged robots," *Journal of Field Robotics*, vol. 35, no. 8, pp. 1311–1326, 2018.

- [55] R. Galati, G. Mantriota, and G. Reina, "Design and development of a tracked robot to increase bulk density of flax fibers," *Journal of Mechanisms and Robotics*, vol. 13, no. 5, 2021.
- [56] W. Budiharto, "Design of tracked robot with remote control for surveillance," in *Proceedings of the 2014 International Conference on Advanced Mechatronic Systems*, pp. 342–346, IEEE, 2014.
- [57] C. Ducros, G. Hauser, N. Mahjoubi, P. Girones, L. Boisset, A. Sorin, E. Jonquet, J. M. Falciola, and A. Benhamou, "Rica: A tracked robot for sampling and radiological characterization in the nuclear field," *Journal of Field Robotics*, vol. 34, no. 3, pp. 583–599, 2017.
- [58] Q. Qiu, Z. Fan, Z. Meng, Q. Zhang, Y. Cong, B. Li, N. Wang, and C. Zhao, "Extended ackerman steering principle for the coordinated movement control of a four wheel drive agricultural mobile robot," *Computers and Electronics in Agriculture*, vol. 152, pp. 40–50, 2018.
- [59] T. N. Mundhenk, C. Ackerman, D. Chung, N. Dhavale, B. Hudson, R. Hirata, E. Pichon, Z. Shi, A. Tsui, and L. Itti, "Low-cost high-performance mobile robot design utilizing off-the-shelf parts and the beowulf concept: the beobot project," in *Intelligent Robots and Computer Vision XXI: Algorithms, Techniques, and Active Vision*, vol. 5267, pp. 293–303, SPIE, 2003.
- [60] J. Ackerman and J. Seipel, "Energy efficiency of legged robot locomotion with elastically suspended loads," *IEEE Transactions on Robotics*, vol. 29, no. 2, pp. 321–330, 2013.
- [61] S. B. Saoud, L. Nciri, and M. Ghrissi, "Path-tracking and parking manoeuvre control of an industrial tricycle robot," *International Journal of Robotics and Automation*, vol. 20, no. 4, pp. 231–239, 2005.
- [62] R. Fernández, E. González, V. Feliú, and A. G. Rodríguez, "A wall climbing robot for tank inspection. an autonomous prototype," in *IECON 2010-36th Annual Conference on IEEE Industrial Electronics Society*, pp. 1424–1429, IEEE, 2010.
- [63] B. Widiawan, C. Triwidiarto, S. Kautsar, and R. Firgiyanto, "Wireless greenhouse monitoring system using tricycle mobile-robot based on raspberry pi," in *IOP Conference Series: Earth and Environmental Science*, vol. 411, p. 012058, IOP Publishing, 2020.
- [64] Z. Bi, S. Y. Lang, and L. Wang, "Improved control and simulation models of a tricycle collaborative robot," *Journal of Intelligent Manufacturing*, vol. 19, no. 6, pp. 715–722, 2008.
- [65] Y. Yagi, "Omnidirectional sensing and its applications," *IEICE transactions on information and systems*, vol. 82, no. 3, pp. 568–579, 1999.
- [66] I. Doroftei, V. Grosu, and V. Spinu, *Omnidirectional mobile robot-design and implementation*. INTECH Open Access Publisher, 2007.

-
- [67] J. Suster, “Neural tracking trajectory of the mobile robot Khepera II internal model control structure.,” *International Conference Process Control*, vol. 9, pp. C153a 1–13, 2010.
- [68] Kumar-Malu and Majumdar-Jharna, “Kinematics, location and control of differential drive mobile robot.,” *Global Journal of Researchers in Engineering: Robotics and Nano-Tech*, vol. 14, no. 1, 2014.
- [69] G. A. Gottwald and I. Melbourne, “A new test for chaos in deterministic systems,” in *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, vol. 460, pp. 603–611, The Royal Society, 2004.
- [70] B. B. Mandelbrot, *The fractal geometry of nature*, vol. 173. Macmillan, 1983.
- [71] H.-O. Peitgen, H. Jurgens, and D. Saupe, *Chaos and fractals: new frontiers of science*. Springer Science & Business Media, 2006.
- [72] K. Foroutan-Pour, P. Dutilleul, and D. Smith, “Advances in the implementation of the box-counting method of fractal dimension estimation,” *Applied Mathematics and Computation*, vol. 105, no. 2, pp. 195–210, 1999.