



UNIVERSIDAD DEL AZUAY
FACULTAD DE CIENCIA Y TECNOLOGÍA
ESCUELA DE INGENIERÍA ELECTRÓNICA

**“Desarrollo de un Controlador
Centralizado de Seguimiento de
Trayectoria para un Sistema
Multi-Robot.”**

Trabajo de graduación previo a la obtención del título de:

INGENIERO ELECTRÓNICO

Autor:

CHRISTIAN EFRAÍN BELTRÁN TÚQUERREZ

Director:

MSC. ANDRÉS CABRERA FLOR

CUENCA, ECUADOR

2019

Desarrollo de un Controlador Centralizado de Seguimiento de Trayectoria para un Sistema Multi-Robot.

RESUMEN

En este trabajo se muestra el desarrollo e implementación de un sistema de seguimiento de trayectorias centralizado para sistemas multi-robot, el cual se basa en un controlador de trayectorias cinemático para robots unicycle y posteriormente transformarlo a cinemática diferencial, también se crea una red inalámbrica tipo estrella para la comunicación del maestro con los robots esclavos, adicionalmente se implementa un método de prevención y evasión de colisiones entre los robots.

Palabras clave: *Seguimiento de trayectoria, velocity obstacle, sistema centralizado, sistema multi-robot, cinemática diferencial, robot móvil.*



Ing. Daniel Iturralde Piedra. Ph.D
Coordinador de Carrera



MsC. Andrés Cabrera Flor
Director de Trabajo de Titulación



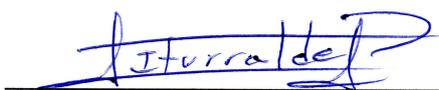
Christian Beltrán Túquerrez
Autor

Development of a Centralized Path Tracking Controller for a Multi-Robot System.

ABSTRACT

This work shows the development and implementation of a centralized trajectory tracking system for multi-robot systems. It is based on a kinematic trajectory controller for unicycle robots that subsequently transforms it to differential kinematics. A star-type wireless network was developed for the communication between the master and the slave robots. Additionally, a method of collision prevention and avoidance between robots was implemented.

Keywords: *Track tracking, velocity obstacle, centralized system, multi-robot system, differential kinematics, mobile robot.*



Ing. Daniel Iturralde Piedra. Ph.D
Faculty Coordinator



MsC. Andrés Cabrera Flor.
Thesis Director



Christian Beltrán Túquerrez
Author

Desarrollo de un Controlador Centralizado de Seguimiento de Trayectoria para un Sistema Multi-Robot

Beltrán Túquerrez Christian Efraín
Escuela de Ingeniería Electrónica
Universidad del Azuay
Cuenca, Ecuador
christian.bel.tuq@gmail.com

Resumen— En este trabajo se muestra el desarrollo e implementación de un sistema de seguimiento de trayectorias centralizado para sistemas multi-robot, el cual se basa en un controlador de trayectorias cinemático para robots unificado y posteriormente transformarlo a cinemática diferencial, también se crea una red inalámbrica tipo estrella para la comunicación del maestro con los robots esclavos, adicionalmente se implementa un método de prevención y evasión de colisiones entre los robots.

Palabras Clave— Seguimiento de trayectoria, velocity obstacle, sistema centralizado, sistema multi-robot, cinemática diferencial, robot móvil.

I. INTRODUCCIÓN

Los sistemas multi-robot (SMR) están conformados por un conjunto de robots, los cuales están relacionados de manera ordenada entre sí y obedecen ciertos parámetros para trabajar conjuntamente, de esta manera pueden alcanzar un objetivo o realizar una tarea la cual sería imposible para un único robot [1]. El objetivo se puede conseguir mediante la cooperación de los diferentes robots en una tarea grande o descomponiendo dicha tarea en subtarefas sencillas las cuales serán abordadas por los diferentes robots de manera individual. Esto resulta de mucho provecho en la industria ya que la relación costo beneficio suele ser menor aplicando SMR en comparación de un robot complejo que tenga la habilidad de llevar a cabo toda la tarea [2].

Dada la importancia de los SMR, se han hecho varias investigaciones que utilizan este sistema como solución a diversos problemas como los que se exponen a continuación:

Uno de los principales usos de los SMR es en el ámbito de la exploración como se presenta en el artículo “Collaborative Multi-Robot Exploration” realizado por Burgard Wolfram et al., en el cual exponen un método de base probabilística para optimizar el área de exploración cubierta por cada robot y evitar que los robots cubran áreas ya descubiertas por otros robots, de esta manera se minimiza el tiempo necesario para cubrir la totalidad del terreno a explorar [3].

Otra aplicación común de los SMR es en el campo de la vigilancia como se expone en “Collision Avoidance for Persistent Monitoring in Multi-Robot Systems with Intersecting Trajectories” realizado por Daniel E. Soltero et al., en el cual utilizan robots móviles para el seguimiento de trayectorias cerradas de manera continua. Cuando el robot pasa por una zona elimina la incertidumbre de la misma, dicha incertidumbre va aumentando en el tiempo si no es eliminada,

es así que los robots tienden a priorizar llegar a las zonas que poseen alta incertidumbre [4].

La manipulación mediante SMR también es posible como se expone en la publicación “Kinematic Multi-Robot Manipulation with no Communication Using Force Feedback” realizada por Wang Zijian y Schwager Mac, donde se expone un algoritmo descentralizado el cual controla a los robots para que se realice la manipulación de un objeto, el cual no puede ser movido por un solo robot. Los robots no tienen comunicación entre ellos, pero utilizan una retroalimentación de la fuerza aplicada y el movimiento que el objeto experimenta, lo cual permite realizar una sincronización de las fuerzas de tal manera que el objeto se mueva como se desea [5].

La investigación titulada “Flocking for multi-robot systems via the Null-Space-based Behavioral control” realizada por Antonelli Gianluca et al., presenta una arquitectura de control basado en comportamiento con base en espacio nulo. Este control ocasiona que los robots actúen como parte de un rebaño, dando así paso a un sistema mucho más complejo, conformado por múltiples robots. Dado que el procesamiento para este tipo de control se lo realiza de manera descentralizada, el robot requiere la información de los robots adyacentes o del entorno inmediato [6].

En el caso de mapeo utilizando SMR se puede mencionar la investigación llamada “Multiple Relative Pose Graphs for Robust Cooperative Mapping” realizado por Been Kim et al. en el cual se utilizan múltiples gráficos de posición obtenidos por diferentes robots para la generación de mapas de una zona, mediante la superposición de puntos de interés iguales en mapas obtenidos por diferentes robots [7].

II. DEFINICIÓN DEL PROBLEMA

Se tiene un grupo de n robots, cada uno denominado R_i donde, $i \in \{1, 2, \dots, n\}$. Cada uno está descrito por un radio r_{R_i} , el cual representa la distancia del centro del robot al borde físico del mismo. Sus movimientos están limitados por restricciones no holonómicas, así como también las restricciones de velocidad $v_{min} < v_{R_i} < v_{max}$ donde $v_{min} > 0$. Cada robot tiene asociada una trayectoria paramétrica $T_i(t)$, la cual pertenece a un espacio euclidiano discreto de 2 dimensiones \mathbb{R}^2 , donde la posición de cada robot en un instante de tiempo τ puede describirse como $P_{R_i}(\tau)$. El entorno permanece ajeno a los robots, dado que estos carecen de sensores, por lo que desconocen su posición, así como la de los otros robots. Sólo cuentan con una comunicación

simplex con el servidor, por la cual reciben las órdenes. El servidor conoce la posición y dirección de todos los robots en tiempo real, y con éstas calcula las velocidades necesarias de cada uno para que realicen su trayectoria con el mínimo error. En caso de que las trayectorias impliquen una colisión entre robots, el sistema cambia automáticamente la trayectoria por una con mínima variación que la inicial en la cual no exista colisión.

III. MARCO TEÓRICO

A. Robot Uniciclo.

El robot uniciclo, como se muestra en la Fig. 1, es una representación ideal de un robot móvil, el cual lo describe como un punto localizado en el centro del eje de las ruedas del robot. Este punto tiene restricciones no holonómicas, por lo que para describir su movimiento se requiere una velocidad lineal y una velocidad angular. Por su simplicidad, es uno de los modelos más usados al momento de diseñar controladores de posición, de seguimiento de trayectoria, etc. [8]

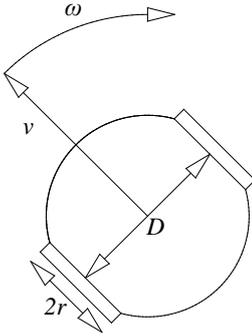


Fig. 1. Representación del Modelo uniciclo.

Las ecuaciones que describen el movimiento en un plano cartesiano, referenciado al origen, son:

$$\begin{aligned}\dot{x} &= v * \cos(\varphi) \\ \dot{y} &= v * \sin(\varphi) \\ \dot{\varphi} &= \omega\end{aligned}\quad (1)$$

En donde:

- φ es el ángulo entre el eje positivo x y la dirección del robot.
- \dot{x} es la razón de cambio del centro del robot de la coordenada x con respecto al tiempo.
- \dot{y} es la razón de cambio del centro del robot de la coordenada y con respecto al tiempo.
- $\dot{\varphi}$ es la razón de cambio de φ con respecto al tiempo.
- v es la velocidad lineal del centro del robot.
- ω la velocidad angular del centro del robot.

Muchas veces se requiere que el punto a controlar no se encuentre en el punto medio del eje de las ruedas, por lo cual se utiliza un modelo uniciclo modificado, el cual tiene el punto de control en la recta perpendicular al eje de las llantas, a una

distancia a del punto medio del mismo [9]. Para este caso, las ecuaciones que describen el movimiento de dicho punto son:

$$\begin{aligned}\dot{x}_c &= v * \cos(\varphi) - a * \omega * \sin(\varphi) \\ \dot{y}_c &= v * \sin(\varphi) + a * \omega * \cos(\varphi) \\ \dot{\varphi} &= \omega\end{aligned}\quad (2)$$

En donde:

- φ es el ángulo entre el eje positivo x y la dirección del robot.
- a es la distancia entre el centro del eje de las llantas al punto de control.
- \dot{x}_c es la razón de cambio de la posición del punto de control en x con respecto al tiempo.
- \dot{y}_c es la razón de cambio de la posición del punto de control en y con respecto al tiempo.
- $\dot{\varphi}$ es la razón de cambio de φ con respecto al tiempo.
- v es la velocidad lineal del punto de control.
- ω la velocidad angular del punto de control.

B. Robot Diferencial.

El robot móvil tipo diferencial, mostrado en la Fig. 2, es el más utilizado de todos los robots móviles, y consta de 2 ruedas activas independientes y una o más ruedas pasivas para mejorar la estabilidad.

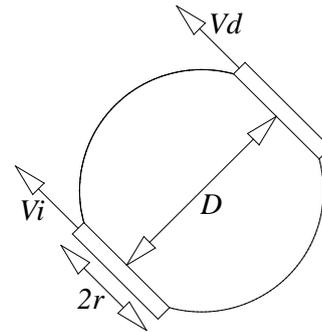


Fig. 2. Representación del Modelo Diferencial.

A diferencia del modelo uniciclo, el modelo diferencial requiere información adicional de la construcción del robot, como la distancia entre ruedas, radio de las llantas, etc. Las ecuaciones que describen el movimiento del robot diferencial en un plano cartesiano [10] son:

$$\begin{aligned}\dot{x} &= \frac{r}{2} * (v_d + v_i) * \cos(\varphi) \\ \dot{y} &= \frac{r}{2} * (v_d + v_i) * \sin(\varphi) \\ \dot{\varphi} &= \frac{r}{D} * (v_d - v_i)\end{aligned}\quad (3)$$

En donde:

- φ es el ángulo entre el eje positivo x y la dirección del robot.
- \dot{x} es la razón de cambio del centro del robot en x con respecto al tiempo.
- \dot{y} es la razón de cambio del centro del robot en y con respecto al tiempo.
- $\dot{\varphi}$ es la razón de cambio de φ con respecto al tiempo.
- v_d es la velocidad angular de la llanta derecha.
- v_i es la velocidad angular de la llanta izquierda.
- r es el radio de las ruedas del robot.
- D es la distancia entre las ruedas.

C. Relación Uniciclo-Diferencial

Como se mencionó anteriormente, el modelo uniciclo es el modelo más aplicado a la hora de diseñar controladores, dado que su nivel de abstracción representa al robot de la manera más simple posible. Por otro lado, en el momento de aplicar los sistemas de control en un ambiente real, su implementación es compleja dado que es un modelo ideal. En la realidad los robots suelen tener diferentes actuadores para controlar su movimiento, como es el caso de los robots diferenciales, por lo cual se requiere una manera de convertir las velocidades del modelo uniciclo al modelo diferencial [10]. Esto se logra mediante un sistema de ecuaciones entre (1) y (3), de lo cual se obtiene:

$$\begin{aligned} v * \cos(\varphi) &= \frac{r}{2} * (v_d + v_i) * \cos(\varphi) \\ v * \sin(\varphi) &= \frac{r}{2} * (v_d + v_i) * \sin(\varphi) \\ \omega &= \frac{r}{D} * (v_d - v_i) \end{aligned} \quad (4)$$

Y resolviendo para v_d y v_i se llega a la relación:

$$\begin{aligned} v_d &= \frac{2v + \omega D}{2r} \\ v_i &= \frac{2v - \omega D}{2r} \end{aligned} \quad (5)$$

Las cuales son las transformaciones que se aplican para obtener las velocidades requeridas por cada uno de los motores del robot diferencial, partiendo de las velocidades del modelo uniciclo.

D. Seguimiento de Trayectoria

Partiendo de una trayectoria definida $T(t)$, se puede llegar a obtener las velocidades necesarias para que un punto recorra dicha trayectoria derivando la trayectoria con respecto al tiempo. Una vez obtenidas dichas velocidades, se procede a transformarlas a las velocidades requeridas para un robot tipo uniciclo, para esto se utiliza el jacobiano de (2), por lo que se obtiene:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} = \begin{bmatrix} \cos(\varphi) & -a * \sin(\varphi) \\ \sin(\varphi) & a * \cos(\varphi) \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} \quad (6)$$

Despejando de esta ecuación las velocidades requeridas para el control del robot se obtiene:

$$\begin{bmatrix} v \\ \omega \end{bmatrix} = \begin{bmatrix} \cos(\varphi) & -a * \sin(\varphi) \\ \sin(\varphi) & a * \cos(\varphi) \end{bmatrix}^{-1} \begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} \quad (7)$$

Dado que (7) representa las velocidades para una trayectoria ideal, ya que no depende del error, se requiere adicionar un término que cambie el comportamiento en función del error actual:

$$\begin{bmatrix} v \\ \omega \end{bmatrix} = \begin{bmatrix} \cos(\varphi) & -a * \sin(\varphi) \\ \sin(\varphi) & a * \cos(\varphi) \end{bmatrix}^{-1} \left(\begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} + \begin{bmatrix} g & 0 \\ 0 & g \end{bmatrix} \begin{bmatrix} x_d - x \\ y_d - y \end{bmatrix} \right) \quad (8)$$

En donde:

- g es la ganancia proporcional del error.
- x_d es el valor de x deseado en ese instante.
- y_d es el valor de y deseado en ese instante.
- x es la posición actual del robot en el eje x.
- y es la posición actual del robot en el eje y.

E. Velocity Obstacle

Se define como "Velocity Obstacle" (VO), al conjunto de todas las posibles velocidades que puede tomar un objeto A, las cuales producirían una colisión con un objeto B en un instante futuro de tiempo. Geométricamente se puede entender a VO, como se muestra en la Fig. 3, partiendo de la suma Minkowski entre los objetos A y B, denotada como $A \oplus B$, tal que:

$$A \oplus B = \{a + b | a \in A, b \in B\} \quad (9)$$

Y definiendo $\lambda(p, v)$ como la línea que inicia en p y tiene una dirección v tal que:

$$\lambda(p, v) = \{p + tv | t \geq 0\} \quad (10)$$

Si se tiene una línea que se origina en el punto central del objeto A y una dirección dada por la velocidad relativa entre los objetos A y B, denotado $(v_A - v_B)$, y esta intercepta con $A \oplus B$ concéntrica con el objeto B, entonces v_A se encuentra en VO del objeto B y se define por:

$$VO_B^A(v_B) = \{v_A | \lambda(p_A, v_A - v_B) \cap A \oplus B \neq \emptyset\} \quad (11)$$

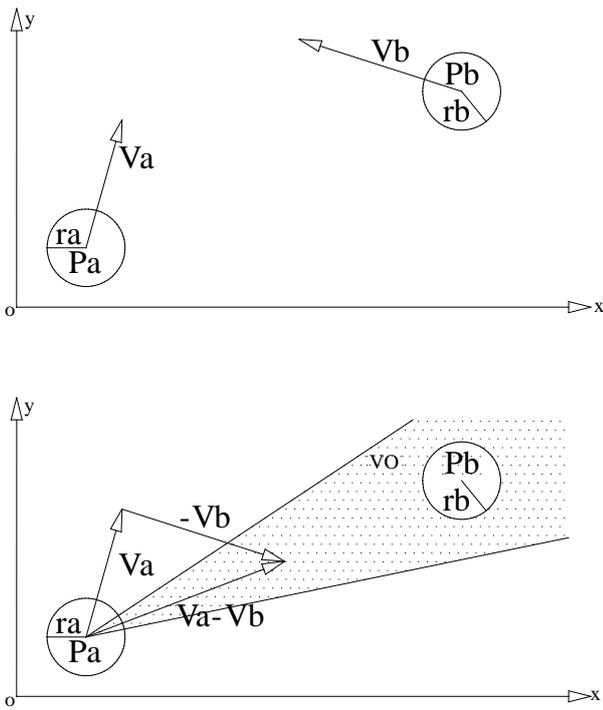


Fig. 3. Representación Geométrica de VO.

Esto significa que si $v_A \in VO_B^A(v_B)$, entonces el objeto A y el objeto B colisionarán en un futuro, si $\lambda(p_A, v_A - v_B)$ es tangencial a $A \oplus B$, entonces los objetos A y B se tocarán en un futuro y si $v_A \notin VO_B^A(v_B)$ entonces los objetos A y B nunca colisionarán [11].

IV. METODOLOGÍA

A. Funcionamiento del sistema

Inicialmente se planteó un sistema de control base y se lo dividió en sus partes principales como se muestra en la Fig. 4, partiendo de este concepto se procedió a desarrollar sus partes individualmente para unificarlo posteriormente.

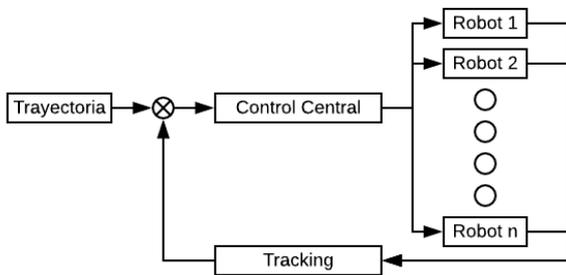


Fig. 4. Diagrama de Bloques del Sistema de Control.

La primera parte es el control del robot móvil, para esto se utilizó plataformas de robots diferenciales, los cuales se muestran en la Fig. 5.



Fig. 5. Robot Diferencial.

Estos robots constan con un chasis, 2 motores DC independientes en los laterales del robot, un controlador doble de motores DC, un microcontrolador con conexión inalámbrica Wi-Fi, una batería y un regulador de voltaje. Estos elementos se encuentran conectados como se muestra en la Fig. 6.

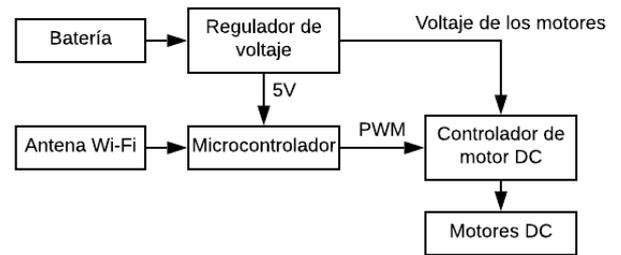


Fig. 6. Diagrama de Bloques de las Conexiones del Robot Diferencial.

Para la comunicación del sistema se seleccionó el protocolo de datagramas de usuario (UDP), dado que este protocolo prioriza la velocidad de envío de los datos. Se configuró a los robots como receptores UDP y se les dio una dirección de protocolo de Internet (IP) única en la red, para de esta manera el servidor pueda comunicarse con ellos.

El control del robot requiere las velocidades del modelo unicycle, provistas por el servidor, el cual las envía en una trama de datos mediante comunicación UDP. Posteriormente en el microcontrolador aplica las ecuaciones que relacionan el modelo unicycle y el modelo diferencial (5), las cuales proveen las velocidades de cada rueda. Una vez se tiene las velocidades requeridas por cada rueda, se procede a aplicar la función de transferencia de cada motor para obtener la modulación por ancho de pulso (PWM), necesaria para que el motor gire a la velocidad requerida.

La señal de PWM es enviada a un controlador de motores DC, el cual se encarga de controlar los voltajes y corrientes adecuados para los motores, así se logra que los motores operen a las velocidades requeridas para el movimiento adecuado del robot diferencial. En la Fig. 7 se muestra el diagrama de bloques del funcionamiento del robot diferencial.

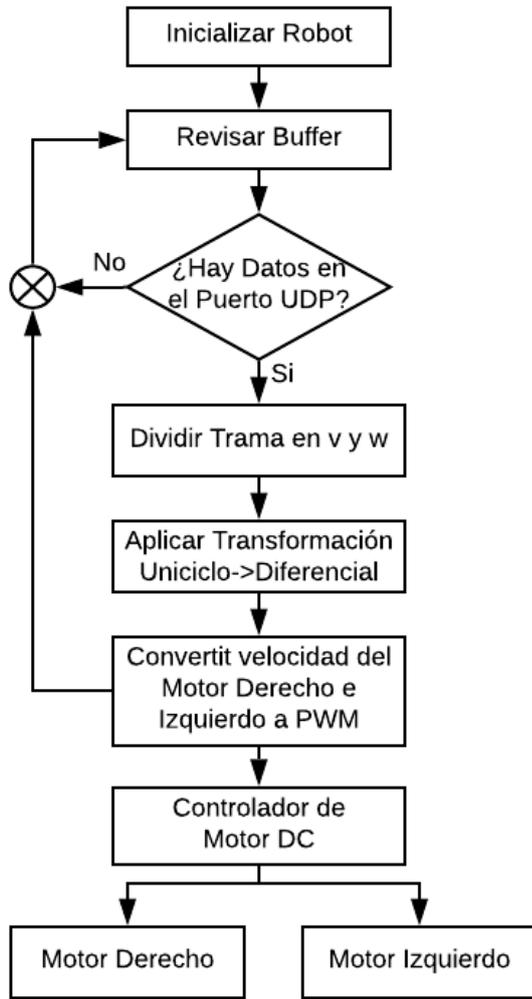


Fig. 7. Diagrama de bloques del programa implementado en el robot diferencial.

Posteriormente para el sistema de retroalimentación se utilizó el sistema desarrollado por Fernando Arbito en “Implementación de un Sistema de Posicionamiento Óptico para un Robot Móvil dentro de un Entorno de Trabajo”, el cual utiliza visión artificial para captar el movimiento de múltiples robots en un espacio de trabajo determinado y proporciona una trama de datos con las coordenadas x , y, φ de los diferentes robots en tiempo real [12], obteniendo así el sistema final mostrado en la Fig. 8.

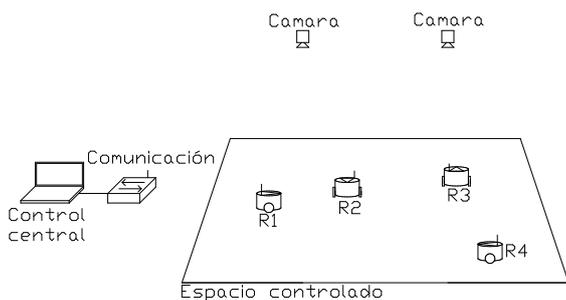


Fig. 8. Esquema del Sistema Final

Para el seguimiento de trayectoria se plantea una trayectoria paramétrica y posteriormente se discretiza la misma. Con los datos discretizados se aplica la ecuación (8), obteniendo así las velocidades necesarias para que el robot realice el recorrido en condiciones óptimas (las velocidades obtenidas son las del punto de control, el cual se encuentra a una distancia a del punto medio del eje de las ruedas). Las ecuaciones que relacionan las velocidades del modelo uniciclo con las velocidades del modelo diferencial hacen referencia al punto medio del eje de las llantas, lo cual se consigue reemplazando (1) en (2):

$$\dot{x}_c = \dot{x} - a * \omega * \sin(\varphi) \quad (12)$$

$$\dot{y}_c = \dot{y} + a * \omega * \cos(\varphi)$$

Y despejando las velocidades del centro del robot se obtiene:

$$\dot{x} = \dot{x}_c + a * \omega * \sin(\varphi) \quad (13)$$

$$\dot{y} = \dot{y}_c - a * \omega * \cos(\varphi)$$

Una vez se tiene el seguimiento de trayectoria aplicado a un robot diferencial con el punto de control a , se debe aplicar un criterio para la evasión de colisiones. Para esto se aplicó VO, el cual requiere verificar si la velocidad relativa entre los 2 objetos se encuentra en el cono de colisión. El criterio de evasión se muestra en la Fig. 9, de esta manera si $\text{límite inferior} < \angle(v_a - v_b) < \text{límite superior}$ la velocidad relativa se encuentra en el cono de colisión.

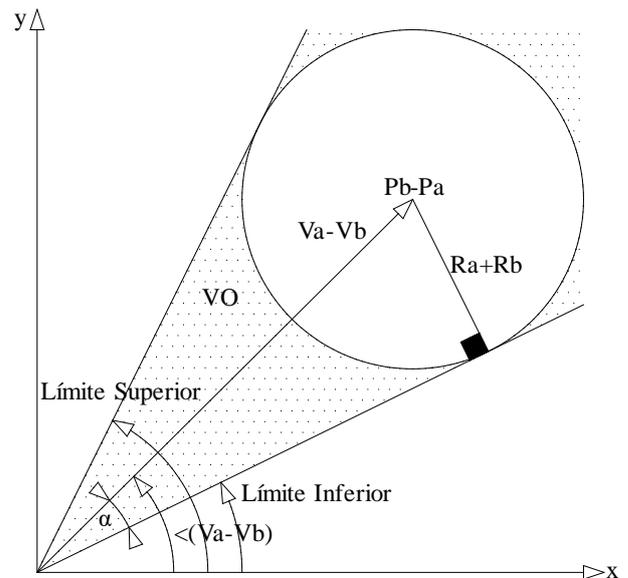


Fig. 9. Resolución geométrica de VO.

Para verificar si $\text{límite inferior} < \angle(v_a - v_b) < \text{límite superior}$ se sigue el Algoritmo 1:

Algoritmo 1: Velocity Obstacle (VO)

```
1 distancia ← √((xa - xb)2 + (ya - yb)2);
2 r ← ra + rb;
3 vr ← va - vb;
4 vφ ← tan-1(vy/vx);
5 α ← sin-1(r/distancia);
6 límitesuperior ← vφ + α;
7 límiteinferior ← vφ - α;
8 si (límiteinferior < vφ < límitesuperior):
9 entonces vr ← límitesuperior + vb;
```

Una vez se tiene todas las partes del sistema se las une en el sistema final, el cual se muestra en el Algoritmo 2.

Algoritmo 2: Programa Final

```
1 R Lista de robots
2 T Lista de trayectorias
3 para todo (Ti ∈ T) hacer:
4     para todo (Rj ∈ R) hacer:
5         Obtener posición x, y, φ;
6         Comparar posiciones con Ti;
7         Calcular nuevas velocidades;
8         para todo (Rk ∈ R) ≠ Rj hacer:
9             Aplicar VO;
10            Actualizar velocidades;
11            Enviar datos;
```

B. Experimentos

Se han planteado diferentes experimentos para comprobar el funcionamiento del sistema en diferentes ámbitos, los cuales son simulados y aplicados en el sistema real. Primero se diseñó una trayectoria cerrada la cual contiene rectas, curvas y ángulos, para probar la capacidad de reacción del sistema a diferentes recorridos. La trayectoria planteada es:

$$T_x(t) = 650 + \frac{(360\sin(0.37t) - 120\sin(1.11t))}{4}$$
$$T_y(t) = 150 + \frac{(390\cos(0.37t) - 150\cos(0.74t) - 60\cos(1.11t) - 30\cos(1.48t))}{4}$$

La segunda prueba consiste en comprobar el seguimiento de trayectoria de múltiples robots, las cuales no tienen colisiones y están en los parámetros admisibles para los robots, las cuales son:

$$T1_x = 363 + 91\sin\left(\frac{\pi t}{17}\right)$$
$$T1_y = 60 + 18t$$
$$T2_x = 545 + 91\sin\left(\frac{\pi t}{17}\right)$$
$$T2_y = 60 + 18t$$
$$T3_x = 721 + 91\sin\left(\frac{\pi t}{17}\right)$$
$$T3_y = 60 + 18t$$

La tercera prueba está diseñada para probar la detección y evasión de colisiones. En este caso la variable que se observa es la distancia entre robots, para este caso se utilizaron las trayectorias:

$$T1_x = 571 + 300\sin\left(\frac{\pi t}{17} - \frac{4\pi}{9}\right)$$
$$T1_y = 60 + 18t$$
$$T2_x = 571 + 300\sin\left(\frac{\pi t}{17} + \frac{2\pi}{9}\right)$$
$$T2_y = 60 + 18t$$
$$T3_x = 571 + 300\sin\left(\frac{\pi t}{17} + \frac{8\pi}{9}\right)$$
$$T3_y = 60 + 18t$$

V. RESULTADOS

A. Simulación

1) Prueba de simulación de seguimiento de trayectoria de un robot.

En la Fig. 10 se muestra la representación de los datos obtenidos mediante la simulación de la primera prueba.

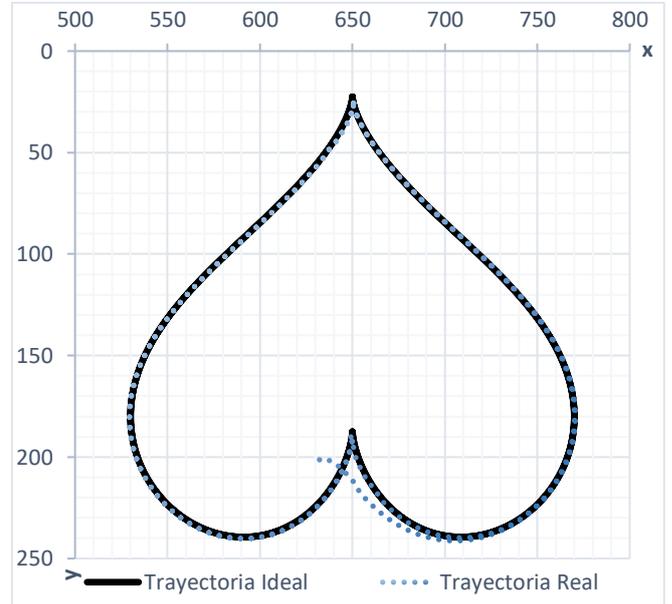


Fig. 10. Gráfica de los datos obtenidos en la simulación de la primera prueba.

En la Fig. 11 se muestra el error que existió entre la trayectoria propuesta y la trayectoria realizada en la simulación.

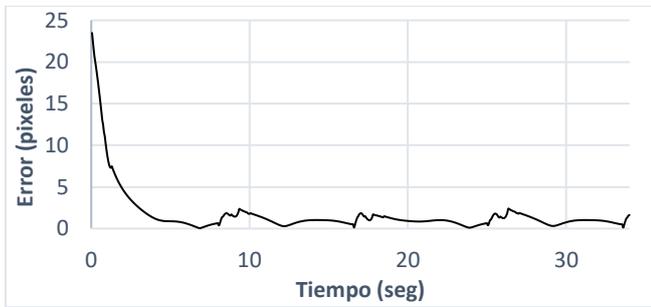


Fig. 11. Error con respecto al tiempo de la simulación de la primera prueba.

En la Tabla I se muestra el error medio y error cuadrático medio de la trayectoria, así también el error medio y error cuadrático medio de la trayectoria una vez se pasó el estado de transición inicial.

TABLA I. TABLA ESTADÍSTICA DE LOS ERRORES OBTENIDOS EN LA SIMULACIÓN DE LA PRIMERA PRUEBA.

Toda la Trayectoria		Percentil 10-100	
Error Medio	Error Cuadrático Medio	Error Medio	Error Cuadrático Medio
1.66	2.83	0.88	0.48

2) Prueba de simulación de seguimiento de trayectoria de múltiples robots.

En la Fig. 12 se muestra la representación de los datos obtenidos mediante la simulación de la segunda prueba.

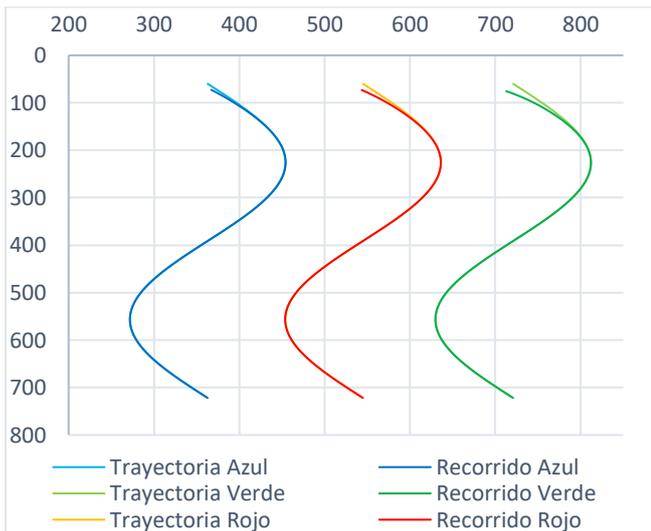


Fig. 12. Gráfica de los datos obtenidos en la simulación de la segunda prueba.

En la Fig. 13 se muestra el error que existió entre la trayectoria propuesta y la trayectoria realizada en la simulación de la segunda prueba.

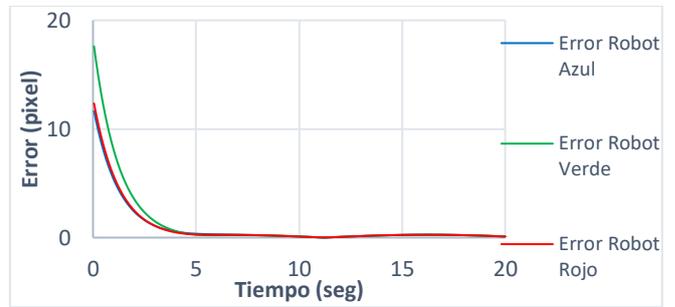


Fig. 13. Error con respecto al tiempo de la simulación de la segunda prueba.

En la Tabla II se muestra el error medio y error cuadrático medio de la trayectoria recorrida en la segunda prueba, así como también se muestra el error medio y error cuadrático medio de la trayectoria una vez se pasó el estado de transición inicial.

TABLA II. TABLA ESTADÍSTICA DE LOS ERRORES OBTENIDOS EN LA SIMULACIÓN DE LA SEGUNDA PRUEBA.

	Toda la Trayectoria		Percentil 10-100	
	Error Medio	Error Cuadrático Medio	Error Medio	Error Cuadrático Medio
Robot Azul	0.864	1.89	0.319	0.397
Robot Verde	1.195	2.89	0.360	0.419
Robot Rojo	0.889	2.01	0.309	0.577

3) Prueba de simulación de seguimiento de trayectoria de múltiples robots con colisión.

En la Fig. 14 se muestra la representación de los datos obtenidos mediante la simulación de la tercera prueba.

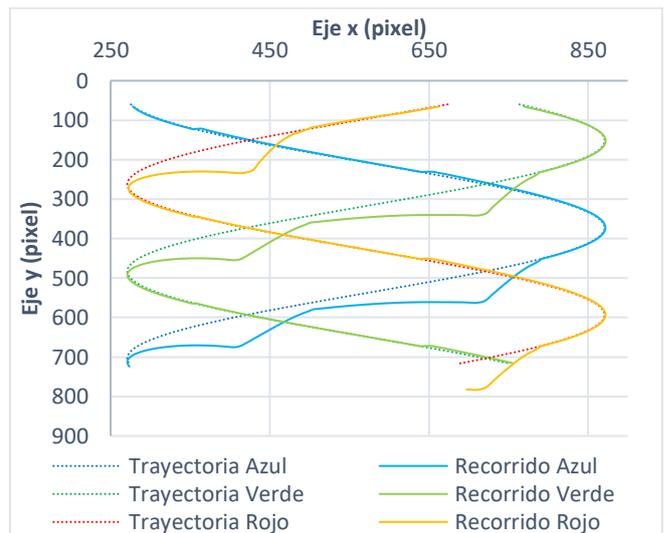


Fig. 14. Gráfica de los datos obtenidos en la simulación de la tercera prueba.

En la Fig. 15 se muestra la distancia que existió entre 2 robots distintos en la simulación de la prueba 3 realizada.

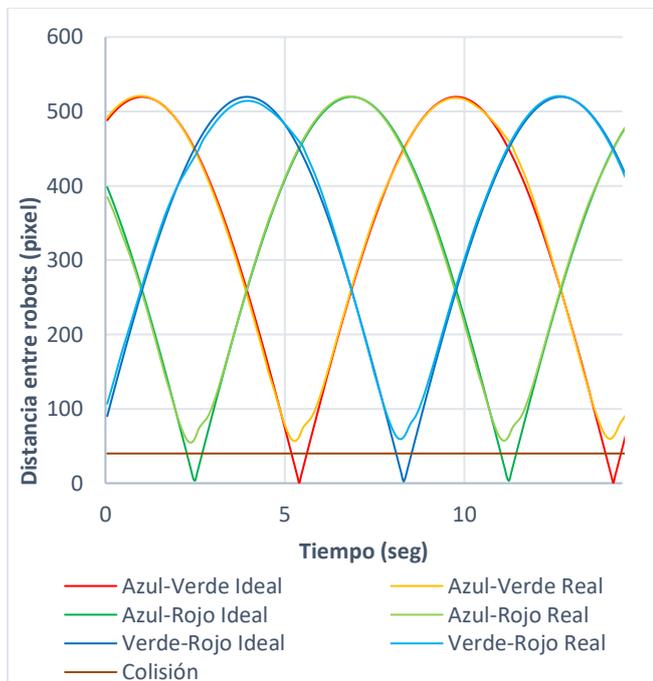


Fig. 15. Distancia entre los robots con respecto al tiempo en la simulación de la tercera prueba.

En la Tabla III se exponen los datos más relevantes de la simulación de la tercera prueba, la cual muestra la distancia mínima a la que llegan a estar los robots a lo largo de todo el recorrido.

TABLA III. DISTANCIAS MÍNIMAS ENTRE LOS ROBOTS EN LA SIMULACIÓN DE LA TERCERA PRUEBA.

	Distancia Mínima (pixeles)
Robot Azul-Verde	56.83
Robot Azul-Rojo	54.77
Robot Verde-Rojo	56.98

B. Experimentación

Luego de haber obtenido los datos de la simulación, se procede a la ejecución de las mismas pruebas en el sistema. Posteriormente se realizará el mismo análisis a los datos obtenidos, para contrastarlos con las simulaciones.

1) Prueba de seguimiento de trayectoria de un robot real.

Al igual que en las simulaciones, en la prueba 1 se presenta una trayectoria cerrada de un único robot. En la Fig. 16 se muestra la trayectoria propuesta, así como la trayectoria recorrida.

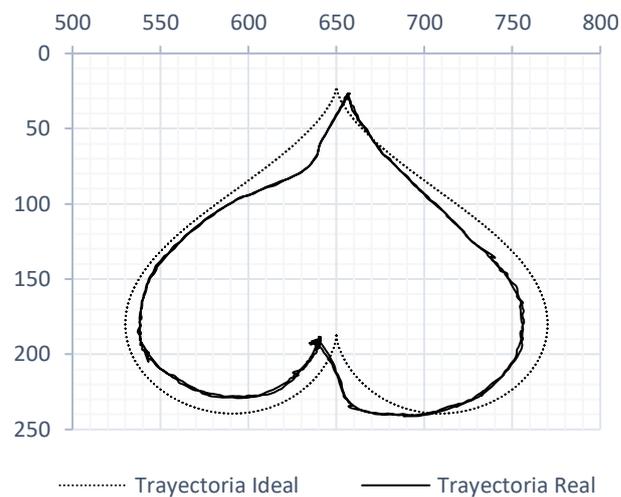


Fig. 16. Gráfica de los datos obtenidos en la primera prueba real.

En la Fig. 17 se puede observar el recorrido realizado en la prueba 1, representado por una estela dejada por el robot al moverse.

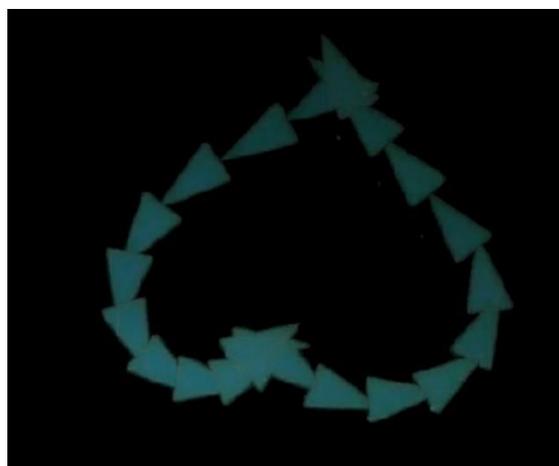


Fig. 17. Imagen captada por las cámaras en la primera prueba.

En la Fig. 18 se muestra el error instantáneo durante todo el recorrido.

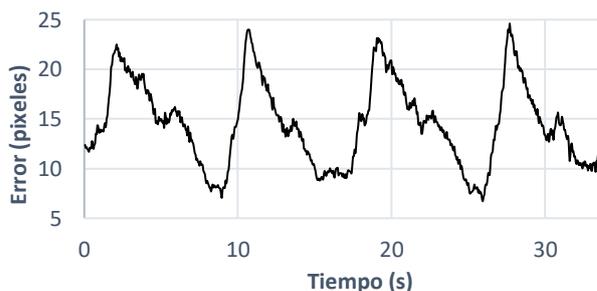


Fig. 18. Error con respecto al tiempo de la primera prueba.

TABLA IV. TABLA ESTADÍSTICA DE LOS ERRORES OBTENIDOS EN LA PRIMERA PRUEBA.

Toda la Trayectoria	
Error Medio	Error Cuadrático Medio
14.44	4.22

2) Prueba de seguimiento de trayectoria de múltiples robots reales.

De la misma manera en el experimento 2, se prueba el seguimiento de trayectoria de múltiples robots simultáneamente. En la Fig. 19 se muestra las trayectorias propuestas, así como las trayectorias recorridas.

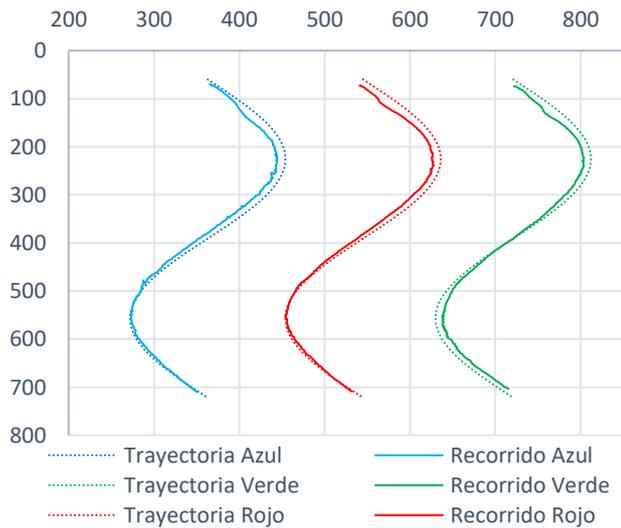


Fig. 19. Gráfica de los datos obtenidos en la segunda prueba real.

En la Fig. 20 se puede observar el recorrido realizado por los diferentes robots en la prueba 2, representados por una estela dejada por cada robot al moverse.

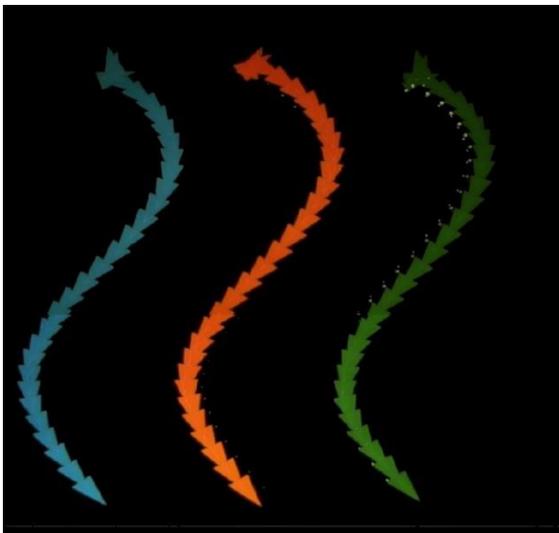


Fig. 20. Imagen captadas por las cámaras en la segunda prueba.

En la Fig. 21 se muestra el error instantáneo de cada robot a lo largo de todo el recorrido.

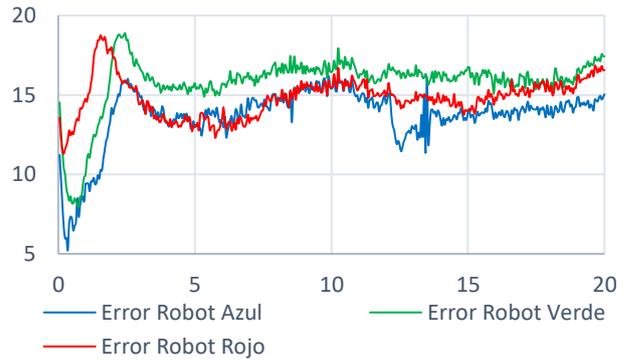


Fig. 21. Error con respecto al tiempo de la segunda prueba.

En la Tabla V se muestra el error medio y error cuadrático medio de la trayectoria recorrida en la segunda prueba, así como también se muestra el error medio y error cuadrático medio de la trayectoria una vez se pasó el estado de transición inicial.

TABLA V. TABLA ESTADÍSTICA DE LOS ERRORES OBTENIDOS EN LA SEGUNDA PRUEBA.

	Toda la Trayectoria		Percentil 10-100	
	Error Medio	Error Cuadrático Medio	Error Medio	Error Cuadrático Medio
Robot Azul	13.684	1.84	14.214	0.908
Robot Verde	15.787	1.78	16.309	0.838
Robot Rojo	14.814	1.22	14.820	0.992

3) Prueba de seguimiento de trayectoria de múltiples robots reales con colisión.

En la Fig. 22 se muestra la representación de los datos obtenidos mediante la ejecución de la tercera prueba.

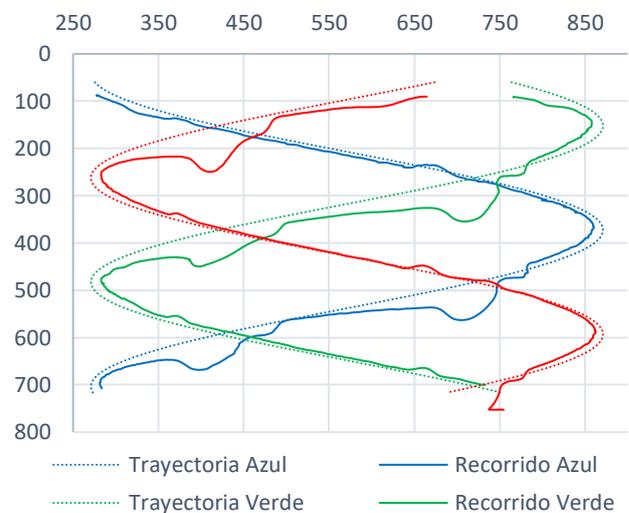


Fig. 22. Gráfica de los datos obtenidos en la tercera prueba real.

En la Fig. 23 se observa el recorrido realizado por los diferentes robots en la prueba 3, representados por una estela dejada por cada robot al moverse.

VI. DISCUSIÓN

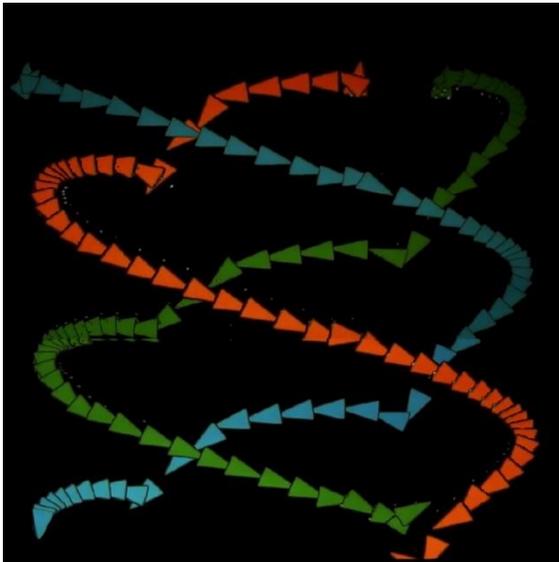


Fig. 23. Imagen captada por las cámaras en la tercera prueba.

En la Fig. 24 se muestra la distancia que existió entre 2 robots distintos en la ejecución de la prueba 3 realizada.

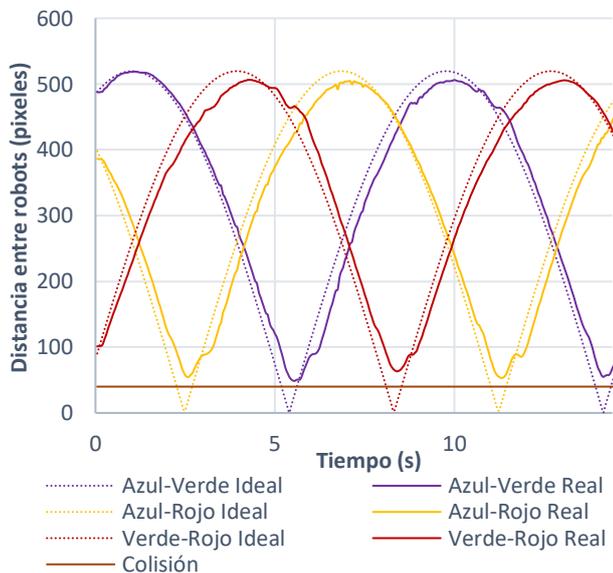


Fig. 24. Distancia entre los robots con respecto al tiempo en la tercera prueba.

En la Tabla VI se exponen los datos obtenidos mediante la ejecución de la tercera prueba, la cual muestra la distancia mínima a la que llegan a estar los robots a lo largo de todo el recorrido.

TABLA VI. DISTANCIAS MÍNIMAS ENTRE LOS ROBOTS EN LA TERCERA PRUEBA.

	Distancia Mínima (píxeles)
Robot Azul-Verde	48.64
Robot Azul-Rojo	53.28
Robot Verde-Rojo	57.13

En la primera prueba (seguimiento de trayectoria con un solo robot) de simulación se puede observar que, después de corto intervalo de tiempo inicial, el error tiende a reducirse en general. Sin embargo, existen ciertos puntos en los que el error aumenta, esto se debe a que la trayectoria posee cambios de dirección marcados, lo que se puede interpretar como una discontinuidad en la trayectoria y esto provoca que las velocidades requeridas estén fuera de los parámetros de funcionamiento del robot. También se observa que el error cuadrático medio es bajo, esto significa que los datos no están separados unos de los otros. Por otro lado, los datos obtenidos en la prueba real se presentan en la Fig. 25 en la cual se aprecia un error medio 14.44 píxeles, lo cual es una medida alta con respecto a la simulación. Por otro lado, su error cuadrático medio es de 4.22 píxeles, lo cual implica que, los puntos no se encuentran dispersos uno del otro, esto se puede constatar en la Fig. 25, dado que la caja, la cual representa el intervalo de los cuartiles 1-3, es más pequeña que los bigotes, los cuales tienen una longitud de 1.5 el rango intercuartílico, por lo que se observa que los datos están agrupados alrededor de la media, por esta razón se deduce que el error medio se debe a un desplazamiento de la trayectoria más que a una trayectoria inestable.

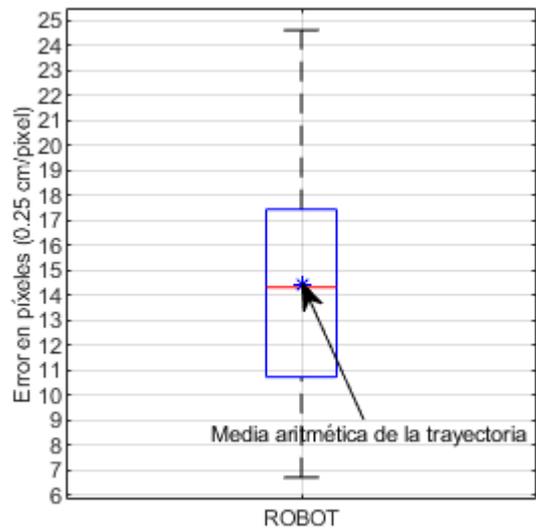


Fig. 25. Diagrama de cajas y vigotes de la primera prueba.

En la segunda prueba se planteó una trayectoria en la cual en todo momento las velocidades están dentro de los parámetros del robot. Al momento de ejecutar tanto la simulación como el experimento real, cuyos resultados se muestran en la Fig. 26, se observa que la mayoría de los datos no se encuentran dispersos, lo cual se traduce a una caja es pequeña, y esto a su vez se traduce en un error medio cuadrático casi nulo, a excepción del inicio de la trayectoria en el cual interviene el posicionamiento inicial del robot. Así mismo los bigotes son cortos, lo que significa que la mayoría de los datos que se encuentran fuera de la caja, no se alejan de los límites de esta. En este caso la principal diferencia entre la simulación y el experimento real recae en el error medio e

igual que en la prueba anterior, presenta un desplazamiento constante a lo largo de toda la trayectoria.

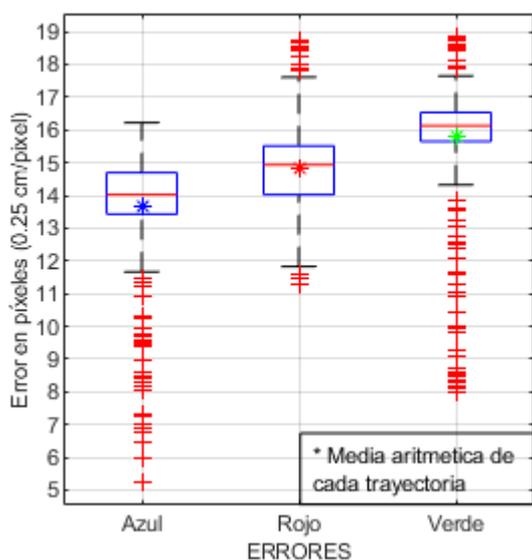


Fig. 26. Diagrama de cajas y vigotes de la primera prueba.

En la tercera prueba se omite el análisis de error en la trayectoria y se evalúa la capacidad de predicción y evasión de colisiones. Al conocer que la distancia del centro al extremo del robot es de 20 píxeles, se considera como colisión cuando la distancia entre los centros de dos robots es igual o menor a 40 píxeles. Tanto la simulación como el experimento real presentan resultados muy similares al cambiar la trayectoria evadiendo al otro robot exitosamente en todos los casos. Los resultados de esta prueba muestran una distancia mínima entre robots de 48.64 píxeles en experimento real y 54.77 píxeles en el caso de simulaciones.

Como se puede observar el sistema desarrollado cumple con su objetivo, dado que realiza el seguimiento de trayectoria de múltiples robots de manera centralizada, utilizando únicamente cámaras como retroalimentación. El principal problema encontrado se describe en las pruebas 1 y 2, en las que se demuestra un desplazamiento constante a la hora del seguimiento de trayectoria. Este desplazamiento se lo atribuye a un error sistemático dado que dicho error no se presenta en las simulaciones, por lo cual se lo puede atribuir a la calibración del sistema de seguimiento o problemas en el procesamiento de la imagen. Este error en las pruebas demostró ser de aproximadamente 16 píxeles lo cual representa un error sistemático del 1.7%.

VII. Referencias

- [1] A. Khamis, A. Hussein y A. Elmogy, Multi-robot Task Allocation: A Review, Cham: Springer, 2015.
- [2] A. Gautam y S. Mohan, A Review of Research in Multi-Robot Systems, Pilani: IEEE, 2012.
- [3] W. Burgard, M. Moors, D. Fox, R. Simmons y S. Thrun, «Collaborative Multi-Robot Exploration,» de *IEEE International Conference on Robotics and Automation*, San Francisco, 2000.
- [4] D. E. Soltero, S. L. Smith y D. Rus, «Collision Avoidance for Persistent Monitoring in Multi-Robot Systems with Intersecting Trajectories,» de *Intelligent Robots and Systems*, San Francisco, 2011.
- [5] Z. Wang y M. Schwager, «Kinematic Multi-Robot Manipulation with no Communication Using Force Feedback,» de *IEEE International Conference on Robotics and Automation*, Estocolmo, 2016.
- [6] An, G. Antonelli, F. Arrichiello y S. Chiaverini, Flocking for multi-robot systems via the Null-Space-based Behavioral control, Cassino: Springer US, 2010.
- [7] B. Kim, M. Kaess, L. Fletcher, J. Leona, A. Bachrach, N. Roy y S. Teller, Multiple Relative Pose Graphs for Robust Cooperative Mapping, Anchorage: IEEE International Conference on Robotics and Automation, 2010.
- [8] J. Plaskonka, The Path Following Control of a Unicycle Based on the Chained Form of a Kinematic Model Derived with Respect to the Serret-Frenet Frame, Wroclaw: IEEE, 2012.
- [9] D. Xie, S. Wang y Y. Wang, «Trajectory tracking Control of Differential Drive Mobile Robot based on improved Kinematics Controller algorithm,» de *Chinese Automation Congress (CAC)*, Xi'an, 2018.
- [10] L. Solaque, M. Molina y E. Rodriguez, «Seguimiento de Trayectorias con un Robot Movil de Configuracion Diferencial,» *USBMed*, vol. 5(1), pp. 26-34, 2014.
- [11] D. Wilkie, J. Van Den Berg y D. Manocha, «Generalized Velocity Obstacles,» de *RSJ International Conference on Intelligent Robots and Systems*, St. Louis, 2009.
- [12] F. Arbito, Implementacion de un Sistema de Posicionamiento Óptico para un Robot Móvil dentro de un Entorno de Trabajo, Cuenca, 2019.