



**Coordinación de robots móviles mediante
un controlador-observador no lineal
basado en el error de posición de las
unidades móviles.**

Autor: Juan Ángel Hernández Suárez

Universidad Antonio Nariño
Facultad de Ingeniería Mecánica, Electrónica y Biomédica
Bogotá, Colombia

2020

Coordinación de robots móviles mediante un controlador-observador no lineal basado en el error de posición de las unidades móviles.

Juan Ángel Hernández Suárez

Proyecto de grado presentado como requisito para optar al título de:
Ingeniero Electrónico

Director(a):
Ing. Christian Erazo Ordoñez PhD.

Línea de investigación
Control.

Universidad Antonio Nariño
Facultad de Ingeniería Mecánica, Electrónica y Biomédica
Bogotá, Colombia
2020

DEDICATORIA

Las palabras son como escudos que me protegen de la demencia.

-Mario Mendoza

Pero antes de morir, quiero luchar por la vida

-Paulo Coelho

AGRADECIMIENTO

Quiero agradecerle a mi padre y madre, por todo su cuidado, apoyo incondicional y consejos que ayudaron a superar mis metas. Agradecer por su esfuerzo dado día a día en su labor de padres para yo lograr ser mejor persona y estar a un paso de poder culminar mi carrera como ingeniero electrónico.

A mi hermana heidy Lizbeth quiero agradecerle por toda su comprensión, compañía y por todo su amor de hermanos que me brindo.

Por último, quiero agradecerle al ingeniero Christian Erazo por su tiempo y enseñanza en mi camino para ser ingeniero electrónico.

ÍNDICE

| | |
|--|-----------|
| RESUMEN | 5 |
| ABSTRACT | 6 |
| 1. INTRODUCCIÓN | 7 |
| 1.1. Estado del arte | 7 |
| 1.2. Planteamiento del problema | 9 |
| 1.3. Justificación | 10 |
| 1.4. Objetivos | 11 |
| 1.4.1. General | 11 |
| 1.4.2. Específicos | 11 |
| 1.5. Alcance | 12 |
| 1.6. Metodología | 13 |
| 2. MODELAMIENTO DE UNIDADES ROBÓTICAS. | 14 |
| 2.1. Sistema de coordenadas | 15 |
| 2.2. Restricciones cinemáticas | 16 |
| 2.3. Modelo cinemático del robot movil | 17 |
| 2.4. Modelo cinemático del robot movil con punto de control desplazado | 20 |
| 2.5. Validación del sistema | 24 |
| 2.5.1. Algoritmo implementado en Matlab para el movimiento del robot movil | 24 |
| 2.5.2. Simulación para la orientación del robot movil. | 28 |
| 3. DISEÑO DE OBSERVADOR DE ESTADOS | 32 |
| 3.1. Motor de corriente directa. | 33 |
| 3.2. Estimador de velocidad angular. | 35 |
| 3.3. Implementación del estimador de velocidad angular en el motor GM25-13CPR. | 38 |
| 3.4. Validación del observador de estados. | 40 |
| 3.4.1. Etapas de implementación en Simulink | 40 |
| 3.4.2. Simulaciones realizadas. | 45 |

| | |
|--|-----------|
| 4. COORDINACIÓN DE ROBOTS MOVILES DIFERENCIALES. | 48 |
| 4.1. Llevar al robot de un punto inicial a un punto final en el marco global. | 49 |
| 4.1.1. Objetivo de control | 49 |
| 4.1.2. Estructura de controlador | 50 |
| 4.2. Ajuste de ganancia k | 54 |
| 4.3. Seguimiento de ruta con curva parametrizada en el robot movil. | 59 |
| 4.4. Coordinación de robots moviles en estructura líder - seguidor | 61 |
| 4.4.1. Ajuste de ganancia k para el seguimiento de una trayectoria | 66 |
| 4.5. Validación del sistema | 69 |
| 4.5.1. Algoritmos implementados. | 69 |
| 4.5.2. Simulaciones para posicionar el robot diferencial a un punto de interés con coordenadas rectangulares. | 75 |
| 4.5.3. Simulación de coordinación de robots líder - seguidor. | 78 |
| 4.5.4. Ajuste de ganancia k en seguimiento de ruta realizada por robot líder . | 80 |
| 5. CONCLUSIONES | 82 |
| 5.1. Consideraciones practicas | 83 |
| 5.2. Trabajos futuros. | 85 |
| REFERENCIAS | 88 |

Índice de Figuras

| | | |
|-------|--|----|
| 2.1. | Representación de coordenadas inerciales y del robot diferencial / Autor. . . . | 15 |
| 2.2. | Esquema de robot holonomo y no holonomo / Autor. | 16 |
| 2.3. | Esquema de robot triciclo, diferencial y triciclo con tracción trasera / Autor. . . | 17 |
| 2.4. | Esquema del modelamiento en el robot triciclo con tracción trasera / Autor. . . | 18 |
| 2.5. | Modelo gráfico de robot movil con punto de control desplazado/ Autor | 20 |
| 2.6. | distribución de N número de robots/ Autor | 23 |
| 2.7. | Movimiento del robot hacia adelante/ Autor. | 28 |
| 2.8. | Movimiento del robot hacia atras/ Autor. | 29 |
| 2.9. | Movimiento del robot girando hacia la izquierda/ Autor. | 29 |
| 2.10. | Movimiento del robot girando hacia la derecha./ Autor. | 30 |
| 2.11. | Movimiento del robot girando sobre su centro de masa/ Autor. | 31 |
| 3.1. | Esquema de motor./ Diseño disponible en: https://www.meca.cinvestav.mx/temarios/clase203204 | |
| 3.2. | Diagrama de bloques de planta y observador de estados / Autor | 37 |
| 3.3. | Esquema general del diagrama de bloques implementado en Simulink para el observador de estados. / Autor | 41 |
| 3.4. | Representacion de la planta y el observador de estados del sitema. | 42 |
| 3.5. | Esquema de velocidad lineal y angular del robot diferencial implementado en Simulink/ Autor. | 43 |
| 3.6. | Esquema para representar los movimientos del robot movil/ Autor. | 44 |
| 3.7. | Simulación de los estados del motor y los estados estimados/ Autor. | 45 |
| 3.8. | Movimiento del robot diferencial hacia la izquierda/ Autor. | 46 |
| 3.9. | Movimiento del robot diferencial hacia la izquierda/ Autor. | 47 |
| 4.1. | Diagrama del algoritmo de control. / Autor | 51 |
| 4.2. | Diagrama explicando coordenadas rectangulares y coordenadas mediante ecuación paramétrica / Autor | 52 |
| 4.3. | Diagrama de algoritmo de control en el motor. / Autor | 54 |
| 4.4. | Acción de control en la velocidad lineal del robot diferencial al posicionarse en los puntos $X_{ref} = 6, Y_{ref} = -4$ / Autor. | 55 |

| | |
|---|----|
| 4.5. Acción de control en la velocidad angular del robot diferencial al posicionarse en los puntos $X_{ref} = 6, Y_{ref} = -4$ / Autor. | 56 |
| 4.6. Simulación del robot diferencial para posicionarse en los puntos $X_{ref} = 6, Y_{ref} = -4$ / Autor | 58 |
| 4.7. Trayecto de circulo en el marco global/ Autor | 60 |
| 4.8. Acciones de control en el seguimiento de trayectoria / Autor. | 67 |
| 4.9. Trayectoria realizada con diferentes valores de ganancia k / Autor. | 68 |
| 4.10. Simulación del robot diferencial para posicionarse en los puntos $X_{ref} = 6, Y_{ref} = -4$ / Autor. | 75 |
| 4.11. Error de posición con posiciones de interés $X_{ref} = 6, Y_{ref} = -4$ / Autor | 76 |
| 4.12. Simulación del robot diferencial para posicionarse en los puntos $X_{ref} = 6, Y_{ref} = -4$ y tiempo de simulación $t = 0 : 0,1 : 30$ / Autor. | 76 |
| 4.13. Error de posición con posiciones de interés $X_{ref} = 6, Y_{ref} = -4$ y tiempo de simulación $t = 0 : 0,1 : 30$ / Autor | 77 |
| 4.14. Trayecto de circulo para líder-seguidor / Autor | 78 |
| 4.15. Trayecto senoidal para líder-seguidor / Autor. | 79 |
| 4.16. Trayecto cosenoidal para líder-seguidor / Autor. | 79 |
| 4.17. Seguimiento del robot líder con diferentes valores de ganancia k / Autor. | 81 |
| 5.1. Trayecto del robot diferencial al punto (6,-4). | 83 |
| 5.2. Trayecto circulo para cuatro robots moviles / Autor. | 84 |

RESUMEN

En este documento se presenta el diseño de un controlador no lineal basado en el error de posición y orientación, para coordinar el movimiento de dos robots móviles utilizando la estrategia líder-seguidor. Inicialmente se describe el modelo matemático del robot diferencial, expresado en términos de ecuaciones diferenciales lineales ordinarias, el cual se validó haciendo uso del software Matlab. En el Capítulo 3 se propone un observador de estados para estimar velocidad angular del robot diferencial, el cual será utilizado para la implementación del controlador. En el Capítulo 4 se describe el controlador que tiene como objetivo la coordinación de robots móviles para seguir una trayectoria descrita por una ecuación de curvas paramétricas, donde el robot líder tiene incorporada la curva paramétrica de referencia la cual van a seguir los demás robots móviles. Finalmente se presentan las conclusiones del trabajo de grado, donde se explica los resultados obtenidos, los objetivos cumplidos y futuros trabajos.

ABSTRACT

In this document we present the design of a nonlinear controller based on position and orientation error in order to coordinate the movement of two mobile robots using the leader-follower strategy. Initially, the mathematical model of the differential robot is described, expressed in terms of ordinary linear differential equations, which were simulated and validated using the Matlab software. In Chapter 3 a state observer is proposed to estimate the angular velocity of the differential robot, which will be used for the implementation of the controller. Then it is describe the controller that aims to coordinate mobile robots to follow a path described by a parametric curve, where the leading robot has the reference parametric curve incorporated which will follow the other mobile robots. Finally, remarks and conclusions are presented, taking into account obtained results and future work.

Capítulo 1

INTRODUCCIÓN

1.1. Estado del arte

Hoy en día, los avances tecnológicos y los requerimientos en la eficiencia y calidad de los procesos de producción han generado la necesidad de diseñar e implementar sistemas robóticos más robustos y flexibles con el fin de mejorar el rendimiento de los procesos, entre ellos: vigilancia [1] [2], búsqueda y rescate [3] [4], agricultura [5], exploración [6] [7], entre otros. Todas las tareas mencionadas requieren gran capacidad de maniobrabilidad y destreza por parte de los robots, de modo que estas tareas no solo pueden llevarse a cabo por un único robot. En estos casos el uso de múltiples unidades robóticas trabajando de forma cooperativa ha sido considerado como una solución.

La coordinación entre diferentes individuos o agentes para realizar tareas comunes, no es un problema reciente en la ingeniería. Este comportamiento es muy común en todo ser vivo, como por ejemplo en las manadas de aves, peces y en general de toda especie del reino animal, en

donde cada individuos se agrupa para realizar diversas formaciones con el fin de protegerse en contra de depredadores [8].

Existen varias técnicas para realizar coordinación de unidades robóticas [8], [9], [10]. En [8] se propone un esquema líder – seguidor para coordinar robots móviles utilizando un control difuso moderno, mejorando el tiempo de respuesta y coordinación del movimiento con respecto a los esquemas tradicionales de control difuso. Un esquema similar se propone en [9], donde los autores diseñan un control híbrido conformado por una estrategia de linealización entrada-salida junto con algunas reglas de compensación, esto con el fin de no solo coordinar las unidades robóticas sino también de controlar la velocidad de los robots. En [11] se diseñan controladores deslizantes de primer y segundo orden para estabilizar asintóticamente los vehículos hacia una formación deseada variante en el tiempo. La ventaja de esta estrategia de control es que no necesita medir o estimar la velocidad del líder. Otras técnicas de control no lineal como backstepping [12] y realimentación dinámica [13] han sido implementadas para realizar coordinación de movimiento.

Por otra parte, en la literatura existen estrategias de control de seguimiento remoto [14], [15], [16] que hacen uso de internet como medio de comunicación y donde dan solución a problemas como incertezas, retrasos, pérdidas de paquetes e intervalos de muestreos variantes en el tiempo presentes en los medios de transmisión.

1.2. Planteamiento del problema

El objetivo en la coordinación y cooperación de robots móviles es que el grupo de robots realicen una misma tarea de forma sincronizada, lo cual es claramente un problema de control que implica el diseño de controladores adecuados para obtener la coordinación de movimiento deseada. Los esquemas más utilizados son el líder-seguidor y control distribuido, donde la mayoría de ellos son basados en un conocimiento exacto de las variables de estado (posición, velocidad y aceleración) de las unidades robóticas. No obstante, en la práctica estas variables no están siempre disponibles y además generalmente están contaminadas con ruido.

Para darle solución a estas problemáticas, en este proyecto se propone el diseño de un controlador no lineal, haciendo uso de un observador de estado para estimar la velocidad angular de cada rueda del robot diferencial.

1.3. Justificación

La rápida convergencia de la robótica y la inteligencia artificial han acelerado el desarrollo de una nueva generación de robots inteligentes capaces de realizar tareas cooperativas de gran complejidad en ámbitos industriales, comerciales y de aplicaciones de consumo [10]. Lo cual ha sido impulsado por diversas investigaciones en el área de control, telecomunicaciones y robótica. Es por esto que este proyecto busca alinearse con las nuevas tendencias en investigación y contribuir a las líneas de investigación del grupo GIBIO.

1.4. Objetivos

1.4.1. General

Diseñar un controlador no lineal basado en el error de posición que coordine el movimiento de dos robots móviles utilizando la estrategia líder-seguidor a través de la herramienta matemática de Matlab.

1.4.2. Específicos

- Describir el modelo matemático del robot móvil diferencial.
- Diseñar un observador de estado para estimar la velocidad angular de cada motor del robot diferencial.
- Diseñar un control no lineal basado en el error de posición del robot móvil.
- Realizar pruebas para validar la coordinación del sistema robótico siguiendo una trayectoria deseada, a través de Matlab.

1.5. Alcance

Este proyecto tiene como finalidad la implementación de dos robots móviles que realicen una coordinación en diferentes trayectorias. Mediante el algoritmo de Matlab se realiza simulaciones donde se coordinen dos robots siguiendo la formación líder – seguidor.

1.6. Metodología

Para el desarrollo de los objetivos planteados en este proyecto se contemplan las siguientes fases:

1. Descripción de las unidades robóticas. El primer paso es describir el modelo cinemático de cada robot basado en la posición cartesiana y su orientación. Se implementará un algoritmo en Simulink para validar las dinámicas de los robots móviles.

2. Diseño de un observador de estado. Puesto que el controlador no lineal depende de las coordenadas cartesianas, y en la práctica solo se tiene información de una sola variable de estado (voltaje), se diseñará un observador de estados para estimar la velocidad angular de cada rueda proporcionada por los motores del robot diferencial. Se implementará un diagrama de bloques en Simulink para validar el funcionamiento del observador en lazo abierto para verificar únicamente los movimientos del robot móvil sin aplicar control.

3. Diseño de un control no lineal para la coordinación de movimiento de robots móviles. El objetivo de los robots móviles es seguir una trayectoria preestablecida mientras los robots mantienen su formación. El algoritmo de control está basado en las señales del error de posición de cada robot. Cada robot tiene información de la posición de sus vecinos, si este error disminuye a cero, significa que el movimiento coordinado se está realizando. Se realizarán simulaciones numéricas en Matlab para verificar el funcionamiento de la técnica de control.

4. Por último se realizará una etapa de validación donde se verificarán si las dos unidades robóticas logran la coordinación.

Capítulo 2

MODELAMIENTO DE UNIDADES ROBÓTICAS.

En este capítulo se describe el modelo cinemático del robot móvil diferencial. Inicialmente se define el sistema de coordenadas que sirven como marco de referencia para describir el movimiento de las unidades robóticas. Luego se describen las restricciones cinemáticas del robot móvil respecto a su desplazamiento en el plano $X_I Y_I$ y la estructura mecánica del robot. El modelo cinemático opera bajo dos criterios, el primero es cuando el robot móvil tiene su modelo cinemático sin un punto de control desplazado, el segundo es cuando el robot móvil tiene un punto de control desplazado. Finalmente se encarga de mostrar el respectivo algoritmo con sus simulaciones para generar los movimientos del robot móvil. En esta última etapa se realiza la validación del modelo para una determinada velocidad lineal en sus ruedas derecha v_d e izquierda v_i .

2.1. Sistema de coordenadas

El desplazamiento de las unidades robóticas en el espacio se rige a partir de dos sistemas de coordenadas.

Inicialmente se tiene un sistema de coordenadas inerciales, cuenta con un sistema de marco global que se fijan en un plano en el que el robot móvil se desplaza, también este plano se considera como el marco de referencia y se denota por X_I, Y_I

Por otra parte, se tiene un segundo sistema de coordenadas las cuales pertenecen a las del robot, este es un sistema de coordenadas locales unidas al robot móvil y se moveran con él. Estará denotado como X_r, Y_r para las posiciones y α para la orientación.

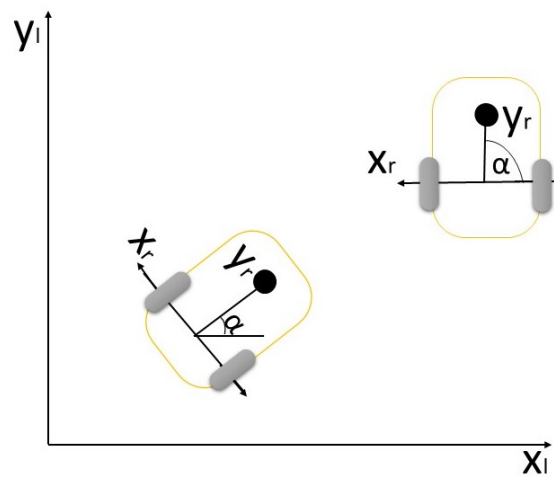


Figura 2.1: Representación de coordenadas inerciales y del robot diferencial / Autor.

2.2. Restricciones cinemáticas

El sistema de locomoción de robots móviles cuenta con ruedas tipo triciclo, el cual consiste en dos ruedas fijas laterales en la parte trasera para la tracción y dirección, una rueda centrada en la parte delantera del robot móvil llamada “rueda libre”, esta sirve para la mejora de dirección y estabilidad del robot.

Este diseño requiere ser guiado no holónimo, el cual consiste en poderse mover instantáneamente hacia adelante y hacia atrás pero no lateralmente ya que depende del movimiento de cada rueda trasera para darle dirección al robot móvil, esto significa que el robot solo podrá moverse en un movimiento curvo hacia adelante o hacia atrás para desplazarse lateralmente. Sin embargo, cuenta con menor deslizamiento en las ruedas a diferencia de otros tipos de robots.

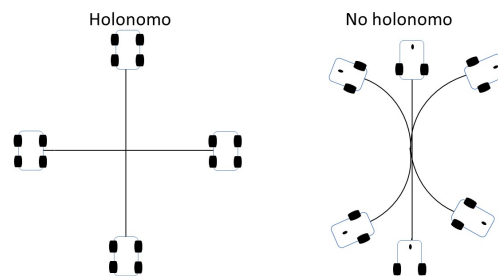


Figura 2.2: Esquema de robot holonómico y no holonómico / Autor.

A pesar de ser un robot triciclo, se interpreta como un robot tipo diferencial, es decir, dos ruedas de dirección y tracción laterales, ya que el modelo de un robot triciclo convencional se realiza con una tracción y dirección únicamente de la rueda delantera, para nuestro diseño la tracción y dirección estará dada por las ruedas traseras, la rueda libre delantera será la que de mejor estabilidad y fácil direccionamiento al robot móvil.

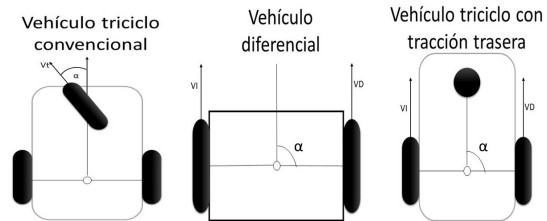


Figura 2.3: Esquema de robot triciclo, diferencial y triciclo con tracción trasera / Autor.

Como se observa en la Figura 2.3 el robot móvil triciclo con tracción trasera tiene las mismas características del robot diferencial, para este caso se trabaja con el modelo cinemático de un robot diferencial.

2.3. Modelo cinemático del robot móvil

El modelo cinemático es una descripción matemática del desplazamiento del robot en el plano $X_I Y_I$ sin considerar las fuerzas que afectan su movimiento [17]. Su desplazamiento se describe en función de las velocidades de las ruedas.

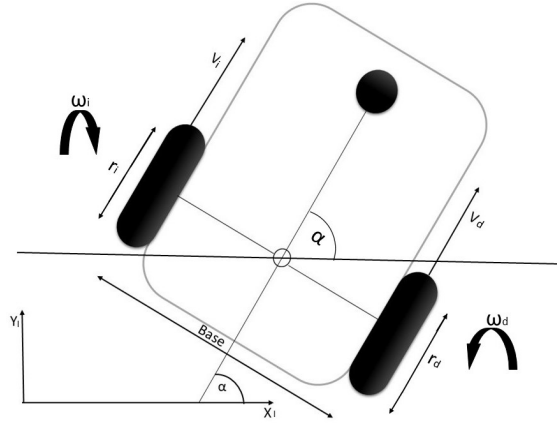


Figura 2.4: Esquema del modelamiento en el robot triciclo con tracción trasera / Autor.

Las velocidades lineales correspondientes a las ruedas derecha e izquierda v_d, v_i se definen en función del radio de cada rueda (r_d, r_i) y su respectiva velocidad angular ω_d, ω_i , es decir:

$$v_i = r_i \omega_i \quad (2.1)$$

$$v_d = r_d \omega_d \quad (2.2)$$

Para el desplazamiento del robot en línea recta, ya sea hacia adelante o hacia atrás, deberá tener sus velocidades lineales iguales, para calcular la velocidad lineal del robot se debe realizar el promedio de estas dos velocidades, también se debe de tener en cuenta que el radio de ambas ruedas debe ser iguales entre ellas para no generar cambios de trayectoria $r_i = r_d = r$.

$$v = \frac{v_d + v_i}{2} \quad (2.3)$$

Para que el robot móvil pueda girar hacia la derecha o izquierda, este deberá rotar sobre su centro de masa, el cual se encuentra en el punto medio entre las dos ruedas (observe Figura 2.4), si una rueda gira hacia adelante y la otra hacia atrás, generara una rotación sobre su centro

de masa, por esta razón la velocidad angular ω la cual esta ligada a la orientación del robot será la diferencia entre las velocidades lineales de cada rueda sobre la longitud que hay entre ellas.

$$\omega = \frac{v_d - v_i}{base} \quad (2.4)$$

Con las ecuaciones de velocidad lineal (2.3) y angular (2.4) del robot móvil, mediante diferentes ecuaciones y estrategias como ecuaciones de Lagrange [18] se obtiene las ecuaciones del modelo cinemático del robot movil.

$$\begin{aligned} \dot{X} &= v \cdot \cos(\alpha) \\ \dot{Y} &= v \cdot \sin(\alpha) \\ \dot{\alpha} &= \omega \end{aligned} \quad (2.5)$$

Donde $[X, Y, \alpha]$ denotan los vectores de posición y orientación del robot sin especificaciones adicionales con respecto a un marco de coordenadas inerciales. El modelo cinemático se puede escribir de forma matricial representado por los términos de las velocidades lineales y angulares del robot diferencial.

$$\begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{\alpha} \end{bmatrix} = \begin{bmatrix} \cos(\alpha) & 0 \\ \sin(\alpha) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} \quad (2.6)$$

2.4. Modelo cinemático del robot móvil con punto de control desplazado

El control de posición del robot móvil, se describe con el modelo cinemático del robot con un punto de control desplazado, en este caso el punto de control se encuentra en una distancia H del centro de masa del robot móvil. En la ecuación (2.6) existe una relación no invertible entre las velocidades angulares de cada rueda (ω_d, ω_i) y las salidas del sistema (X, Y), por lo tanto el punto medio en el centro de masa del robot móvil, debe desplazarse a una distancia H dentro de la misma estructura del robot móvil para evitar este problema.

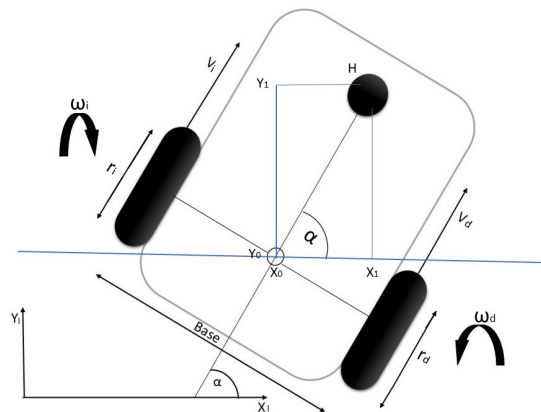


Figura 2.5: Modelo gráfico de robot móvil con punto de control desplazado/ Autor

Para describir el modelo presentado en la Figura 2.5 se debe tener en cuenta las respectivas distancias donde $(X_0 - X_1) = Xr$, $(Y_0 - Y_1) = Yr$.

Estas son las distancias donde se encuentra el punto de desplazamiento del robot móvil. Observe la Figura 2.5 la distancia H esta donde se encuentra la rueda libre, y observe como

este punto con respecto al plano cartesiano genera un triángulo rectángulo. Para calcular las distancias se sabe que el $\cos(\alpha) = \frac{(X_r)}{H}$ y $\sin(\alpha) = \frac{(Y_r)}{H}$.

Despejando las distancias X_r, Y_r se obtiene la distancia del punto H con respecto al centro de masa del robot.

$$\begin{aligned} X_r &= H \cdot \cos(\alpha) \\ Y_r &= H \cdot \sin(\alpha) \end{aligned} \tag{2.7}$$

Esta es la distancia del punto de control H desplazado, ahora se agrega la posición sin este punto. Se suma las posiciones del robot móvil sin punto de control desplazado X, Y .

$$\begin{aligned} X_r &= X + H \cdot \cos(\alpha) \\ Y_r &= Y + H \cdot \sin(\alpha) \end{aligned} \tag{2.8}$$

La ecuación (2.8) relaciona las posiciones del robot con punto de control desplazado (X_r, Y_r) con las posiciones del robot sin punto de control desplazado (X, Y) . Estas representan las ecuaciones geométricas del sistema, ahora para obtener el modelo cinemático se deriva con respecto al tiempo tomando a α como variable de estado.

$$\begin{aligned} \dot{X}_r &= \dot{X} - H \cdot \dot{\alpha} \cdot \sin(\alpha) \\ \dot{Y}_r &= \dot{Y} + H \cdot \dot{\alpha} \cdot \cos(\alpha) \\ \dot{\alpha} &= \omega \end{aligned} \tag{2.9}$$

Ahora se reemplaza los valores de la ecuación (2.5) en la ecuación (2.9) para obtener el modelo cinemático que será utilizado más adelante en el control de posición del robot móvil.

$$\begin{aligned}
\dot{X}r &= v \cdot \cos(\alpha) - H \cdot \omega \sin(\alpha) \\
\dot{Y}r &= v \cdot \sin(\alpha) + H \cdot \omega \cos(\alpha) \\
\dot{\alpha} &= \omega
\end{aligned} \tag{2.10}$$

Al igual que el modelo anterior, este se puede escribir de forma matricial representado por las velocidades lineales y angulares.

$$\begin{bmatrix} \dot{X}r \\ \dot{Y}r \end{bmatrix} = \begin{bmatrix} \cos(\alpha) & -H \cdot \sin(\alpha) \\ \sin(\alpha) & H \cdot \cos(\alpha) \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} \tag{2.11}$$

Donde $X\dot{Y}r$ representa las velocidades en el punto de interés del robot, U es la matriz de velocidades y J es una matriz de relación entre las velocidades $X\dot{Y}r$ y U .

$$J = \begin{bmatrix} \cos(\alpha) & -H \cdot \sin(\alpha) \\ \sin(\alpha) & H \cdot \cos(\alpha) \end{bmatrix} \tag{2.12}$$

$$U = \begin{bmatrix} v \\ \omega \end{bmatrix} \tag{2.13}$$

$$X\dot{Y}r = \begin{bmatrix} \dot{X}r \\ \dot{Y}r \end{bmatrix} \tag{2.14}$$

Reescribiendo la ecuación de forma compacta se tiene la ecuación que representa el movimiento del robot móvil.

$$X\dot{Y}r = J \cdot U \tag{2.15}$$

En la representación de dos o más robots móviles se considera un grupo de N robots donde $p \in [1, N]$, donde cada uno de ellos tiene la siguiente representación.

$$X\dot{Y}r_p = J_p \cdot U_p \quad (2.16)$$

De las cuales denotan los vectores de posición y orientación del p -ésimo robot del grupo con respecto a las coordenadas del robot.

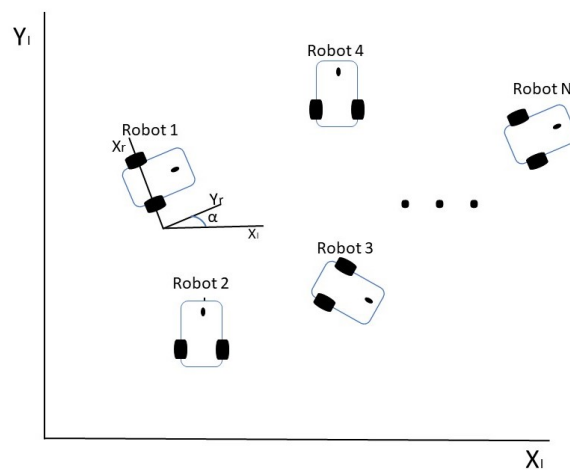


Figura 2.6: distribución de N número de robots/ Autor

2.5. Validación del sistema

2.5.1. Algoritmo implementado en Matlab para el movimiento del robot móvil

Inicialmente se demuestra los movimientos posibles en el robot móvil, movimientos en línea recta hacia adelante y hacia atrás o movimientos curvos para realizar el cambio de dirección en el robot diferencial.

Este algoritmo se realiza en Matlab implementando la ecuación (2.10) correspondiente a la del robot móvil con punto de control desplazado y mediante funciones externas como “MobileRobot y MobilePlot” se gráfica el trayecto realizado del robot móvil.

A continuación, se muestra el algoritmo 1 correspondiente a las ecuaciones implementadas para sus movimientos, el cual está basado en ingresar diferentes velocidades lineales a cada rueda para comprobar el comportamiento del robot móvil.

Algoritmo 1: Solución de las ecuaciones cinemáticas del robot diferencial

◇ Se establece un tiempo de simulación t con un tiempo de muestreo ts
 $ts = 0.1;$
 $t = 0:ts:4;$
◇ Se ingresa valores de entrada, siendo las velocidades en cada rueda
 $V_i;$
 $V_d;$
◇ Se establece las posiciones iniciales en XY y la orientación en α
 $X(1) = 0;$
 $Y(1) = 0;$
 $alpha(1) = 0;$
◇ Se define las constantes. El radio de las ruedas r y la longitud de la base del robot en metros.
 $base = 0.2;$
 $r = 0.05;$
◇ Se calcula las velocidades lineales y angulares del robot de acuerdo a las ecuaciones (2.3)(2.4).
 $v = (r*((VI + VD)/2));$
 $w = r*((VD - VI)/base);$
for $k = 1:length(t)$ **do**
 ◇ Se calcula las velocidades de interés a partir de las velocidades lineales y angulares remplazadas en la ecuación(2.10)
 $xpunto(k) = v(k)*cos(alpha(k)) - H*w(k)*sin(alpha(k));$
 $ypunto(k) = v(k)*sin(alpha(k)) + H*w(k)*cos(alpha(k));$
 $alphapunto = w(k);$
 ◇ Integración numérica para obtener las posiciones a partir de las velocidades de interés.
 $x(k+1) = x(k) + ts*xpunto(k);$
 $y(k+1) = y(k) + ts*ypunto(k);$
 $alpha(k+1) = alpha(k) + ts*alphapunto;$
end

En el algoritmo 2 se muestra la forma en la que debe graficar el trayecto del robot, especifica cuanto debe moverse, tamaño de la gráfica y funciones adicionales para representar gráficamente el robot móvil en el plano $X_I Y_I$.

En esta parte serán necesarias las funciones de MobileRobot la cual tiene especificaciones geométricas del robot móvil (largo, ancho, alto o color) para lograrse ver gráficamente. Por otra parte se realiza el esquema de un robot movil diferencial en una proyección 3D mediante la función "MobilePlot" de matlab, esta describe el trayecto con el cual se dirige el robot. Se debe tener en cuenta que en las siguientes simulaciones se es necesario este algoritmo para lograr observar el trayecto del robot, lo que quiere decir que en las siguientes simulaciones también se tendrá este algoritmo.

Algoritmo 2: Graficar el recorrido del robot movil

- ◇ Se define una variable *pasos* que se utiliza para definir a cuantos pasos realiza el movimiento el robot

```
pasos=4;
```

- ◇ Se configura la gráfica donde se observa el movimiento del robot, posición donde se observa el robot y el tamaño de la cuadrícula donde se observa la simulación

```
fig=figure;
```

```
set(fig,'position',[10 60 980 600]);
```

```
axis square; cameratoolbar;
```

```
axis([-3 3 -3 3 -0.5 1]);
```

- ◇ Se llama la función *MobileRobot* para representar el robot en la gráfica

```
MobileRobot;
```

```
M1=MobilePlot(x(1),y(1),alpha(1));hold on ;
```

```
M2=plot(xr(1),yr(1),'b','LineWidth',2);
```

```
for i=1:pasos:length(t) do
```

- ◇ Realiza unicamente el trayecto del robot, sin representar graficamente el robot movil

```
delete (M1);
```

```
delete (M2);
```

```
M1=MobilePlot(x(i),y(i),alpha(i)); hold on;
```

```
M2=plot(xr(i),yr(i),'b','LineWidth',2);;
```

```
pause(ts);
```

2.5.2. Simulación para la orientación del robot móvil.

En la validación del sistema se hizo uso del software Matlab para verificar el comportamiento del robot móvil. Al obtener su modelo cinemático e implementarlo en un algoritmo, era posible observar el movimiento del robot móvil al aplicar diferentes velocidades lineales en cada rueda del robot. En las simulaciones se tuvo en cuenta que el radio de las ruedas $r = 0,05m$ y el diámetro de la base es $Base = 0,2m$ con un tiempo de muestreo de $ts = 0,1$.

Primero se realizan las simulaciones con velocidades lineales iguales en ambas ruedas del robot diferencial, como antes se mencionó para velocidades lineales iguales, el movimiento del robot móvil deberá recorrer un trayecto en línea recta. Se realiza la simulación con velocidades lineales iguales positivas para ambas ruedas obteniendo un recorrido presentado en la Figura 2.7 .

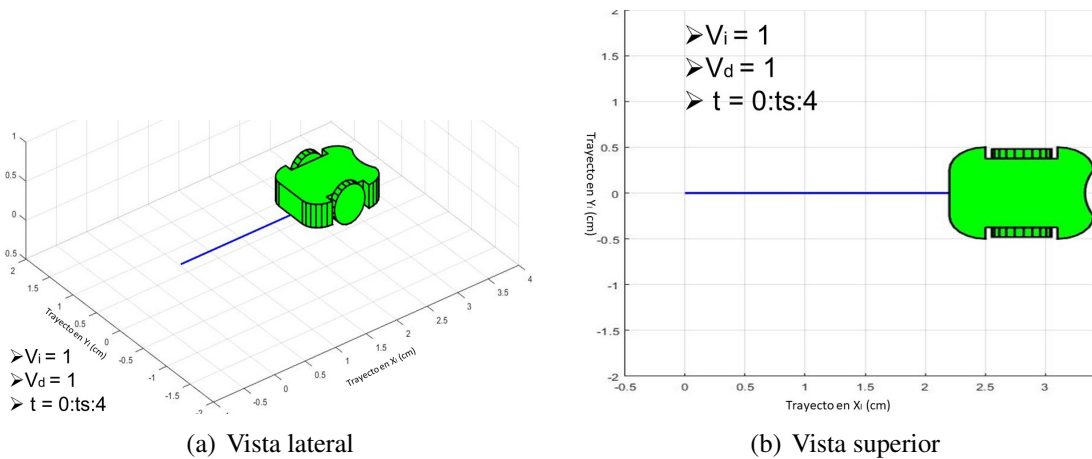


Figura 2.7: Movimiento del robot hacia adelante/ Autor.

Ahora se suponen velocidades lineales iguales negativas dando como resultado un trayecto en línea recta hacia atrás presentado en la Figura 2.8.

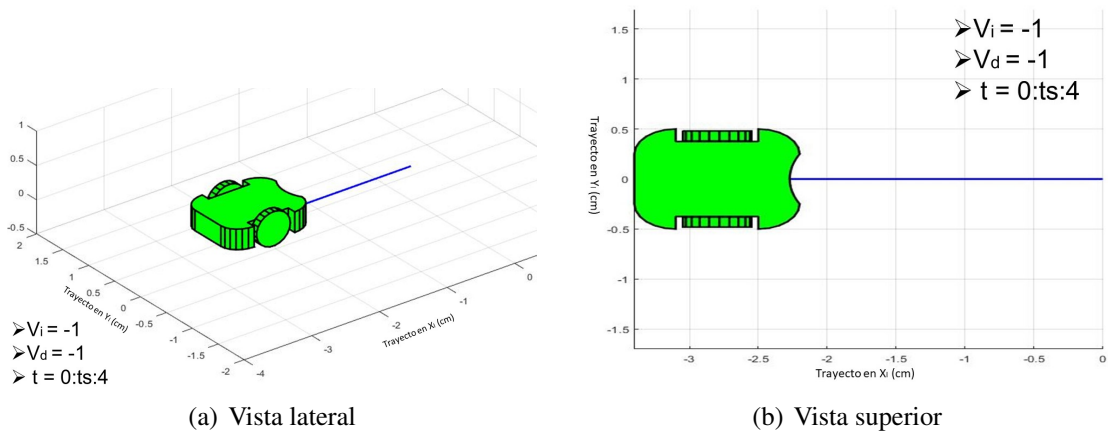


Figura 2.8: Movimiento del robot hacia atras/ Autor.

Ahora se hace la simulación para velocidades lineales diferentes en cada rueda. Primero se realiza la prueba para que el robot móvil gire en dirección izquierda, en este caso se deja la rueda derecha con una velocidad lineal mayor que en la rueda izquierda y se obtiene un resultado como se observa en la Figura 2.9.

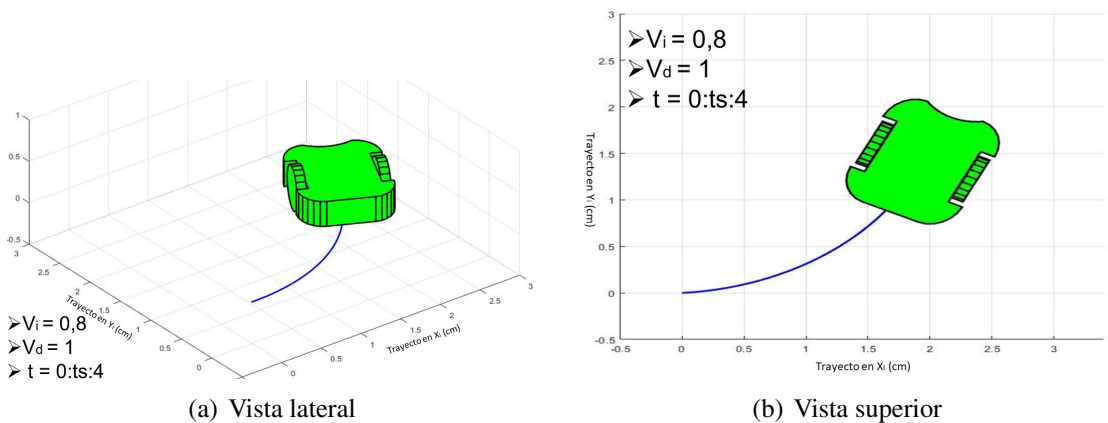


Figura 2.9: Movimiento del robot girando hacia la izquierda/ Autor.

Se hace la prueba para que el robot móvil gire en dirección derecha dejando la velocidad

lineal de la rueda izquierda mayor que en la rueda derecha obteniendo el resultado que se observa en la Figura 2.10.

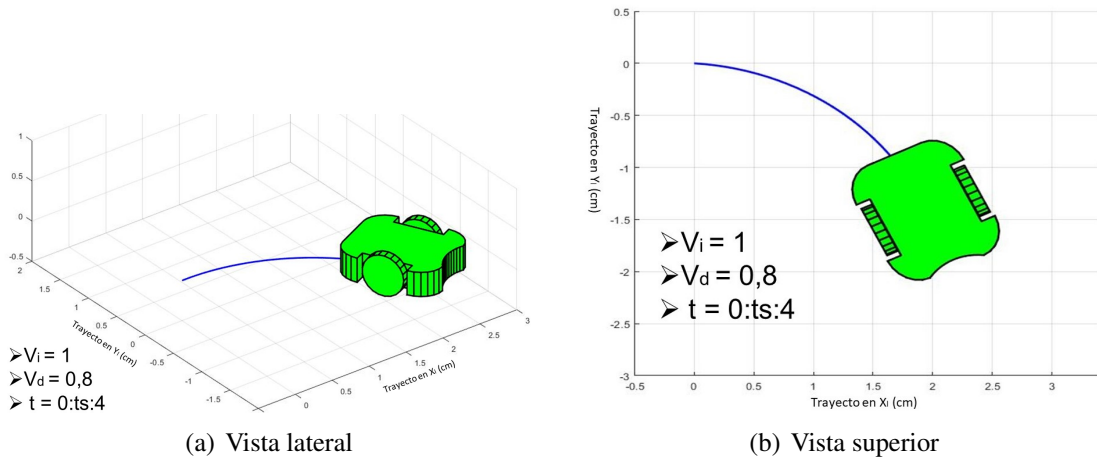
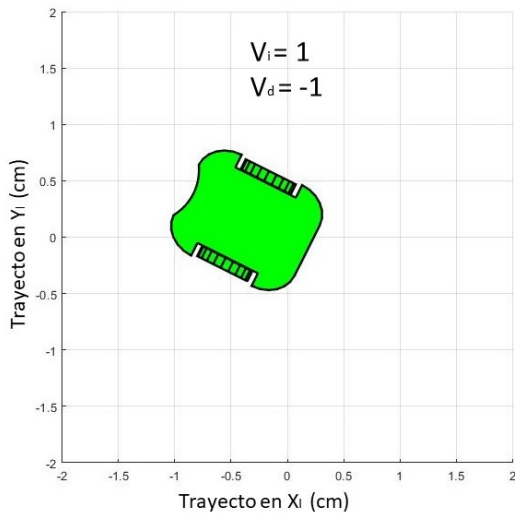
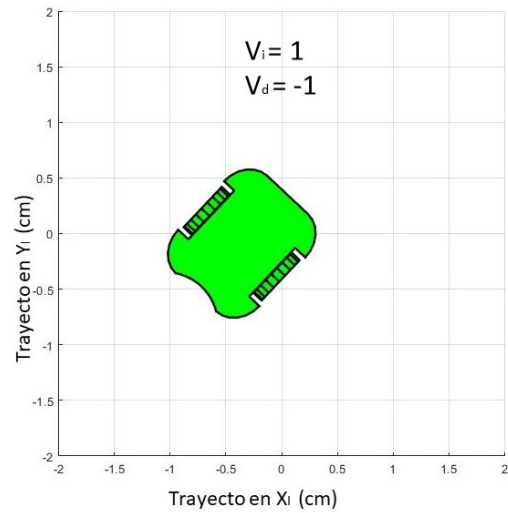


Figura 2.10: Movimiento del robot girando hacia la derecha./ Autor.

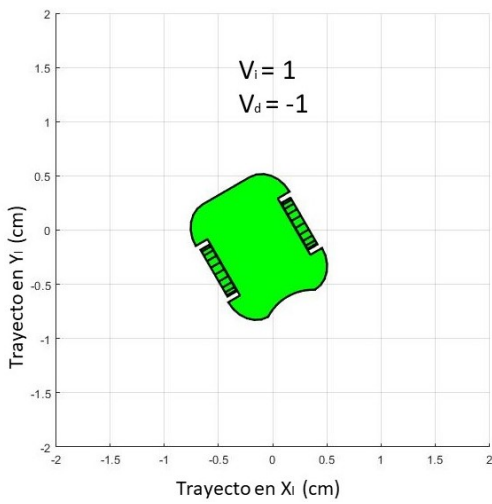
Por último, se realiza la prueba para velocidades lineales contrarias, es decir una rueda con velocidad lineal positiva y la otra negativa. El comportamiento del robot móvil, como se mencionó anteriormente sería una rotación sobre su centro de masa observado en la Figura 2.11.



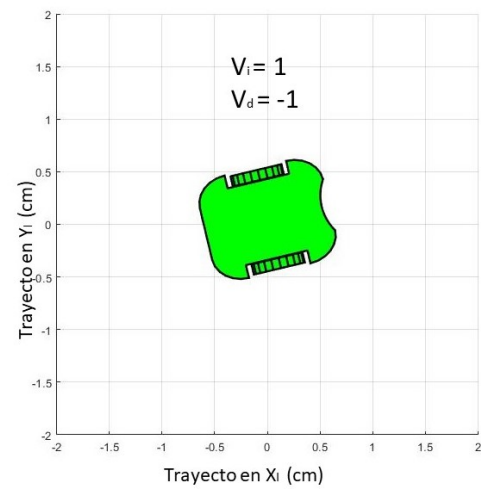
(a)



(b)



(c)



(d)

Figura 2.11: Movimiento del robot girando sobre su centro de masa/ Autor.

Capítulo 3

DISEÑO DE OBSERVADOR DE ESTADOS

En este capítulo se introduce un estimador de estados a cada motor del robot diferencial, cuyo objetivo es proporcionar información de la velocidad angular al introducirle voltajes de entrada. Inicialmente se describe el modelo del motor de corriente directa, donde se interesa estimar la velocidad angular del motor a partir de sus estados. Después se interesa verificar la observabilidad del sistema para determinar si es posible implementar un estimador de estados. Finalmente se diseña un estimador de estados teniendo en cuenta la estructura del observador, la sintonización de las ganancias l y las respectivas simulaciones en Simulink para observar el comportamiento del estimador de estados y implementarlo con el modelo cinemático del robot móvil y hacer pruebas en lazo abierto para verificar únicamente los movimientos del robot diferencial.

3.1. Motor de corriente directa.

Se define el modelo matemático con el cual se identifica el motor del robot diferencial. En la descripción del motor se toma como un modelo electromecánico, tomando la parte eléctrica y mecánica para generar el movimiento del motor, a continuación se muestra el esquema general para describir el motor.

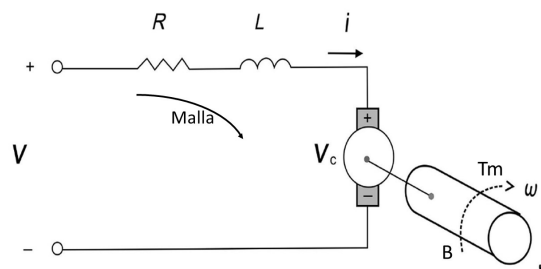


Figura 3.1: Esquema de motor./ Diseño disponible en: https://www.meca.cinvestav.mx/temarios/clase_203204.pdf

Inicialmente se realiza el sistema eléctrico del motor, su funcionamiento eléctrico está definido por una resistencia de armadura R , una inductancia o bobina de armadura L , una fuente de voltaje de entrada v y una corriente i que circula en la Figura (3.1) generando un voltaje en el motor denominado v_c . La ecuación eléctrica del motor se describe de la siguiente manera.

$$v(t) = R \cdot i(t) + L \cdot \frac{di(t)}{dt} + v_c(t) \quad (3.1)$$

Ahora se define el sistema mecánico del motor, donde se tiene en cuenta un torque generado por el motor Tm , que será igual a un momento de inercia In multiplicado por la aceleración generada en el motor $\frac{d\omega(t)}{dt}$, mas un coeficiente de fricción f_r multiplicado por la velocidad del

motor ω . La ecuación mecánica del motor se describe de la siguiente manera.

$$Tm(t) = In \cdot \frac{d\omega(t)}{dt} + f_r \cdot \omega(t) \quad (3.2)$$

Obtenidas las ecuaciones mecánicas y eléctricas del motor, se aplica el método de Laplace para obtener la función de transferencia del motor de corriente directa. Este modelo relaciona la respuesta del sistema ω con una señal de entrada v .

$$\frac{\omega(s)}{v(s)} = \frac{K_m}{L \cdot In \cdot S^2 + (R \cdot In + L \cdot f_r)S + R \cdot f_r + K_c \cdot K_m} \quad (3.3)$$

Donde K_m es una constante mecánica para relacionar el torque ($Tm = K_m \cdot i$) y K_c es una constante eléctrica para relacionar el voltaje del motor ($v_c = K_c \cdot \omega$). También se puede representar el modelo del motor en espacio de estados. Para comprender un poco mejor el modelamiento del motor se puede observar en las referencias [19] [20] para determinar la función de transferencia y sus variables en espacio de estados a detalle.

$$\begin{bmatrix} \dot{\omega} \\ \dot{i} \end{bmatrix} = \begin{bmatrix} -\frac{f_r}{In} & \frac{K_m}{In} \\ -\frac{K_c}{L} & -\frac{R}{L} \end{bmatrix} \begin{bmatrix} \omega \\ i \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{L} \end{bmatrix} v \quad (3.4)$$

$$y = \begin{bmatrix} 0 & 1 \end{bmatrix} \begin{bmatrix} \omega \\ i \end{bmatrix} \quad (3.5)$$

Donde también se puede describir de forma compacta como se muestra a continuación:

$$\begin{aligned}\dot{x} &= Ax + Bu \\ y &= Cx\end{aligned}\tag{3.6}$$

Donde x es el vector de estados, y son las salidas del motor, u son los vectores de entradas, una matriz de estados A , matriz de entradas B y matriz de salidas del sistema C . Finalmente se escoge un motor para reemplazar los valores en la ecuación (3.4). El motor seleccionado es el GM25-13CPR el cual cumple con las siguientes características.

| | |
|-------------------------------|------------------------------|
| Voltaje nominal | 12V DC |
| Resistencia de armadura R | 6.14 Ω |
| constante eléctrica K_c | 0.04913 Vs/rad |
| constante mecánica K_m | 0.04913 N·m/A |
| Inductancia de armadura L | 8.9 mH |
| coeficiente de fricción f_r | 40.923 μ (N·m - s)/Rad |
| Momento de inercia I_n | 7.95 μ Kg·m ² |

Tabla 3.1: Tabla de características de motor GM25 - 13CPR / disponible en <https://www.play-zone.ch/de/yfrobot-gm25-13cpr-jga25-gear-motor-mit-encoder.html>

3.2. Estimador de velocidad angular.

Para realizar un estimador de estados es necesario saber si los estados son observables, una forma para garantizar que el sistema sea observable es con la matriz de observabilidad denominada O .

Definición[observabilidad]: Un sistema lineal invariante en el tiempo es observable si a partir de las mediciones de la entrada y la salida en un intervalo de tiempo finito se pueden obtener los estados del sistema.

Criterio[observabilidad]: Un sistema lineal invariante en el tiempo es observable si la matriz de observabilidad O tiene un $Rango = n$ y su determinante es diferente de cero.

$$O = \begin{bmatrix} C \\ C \cdot A \\ C \cdot A^2 \\ C \cdot A^{n-1} \end{bmatrix} \quad Det(O) \neq 0 \quad (3.7)$$

Donde A es la matriz de estados del motor y C es la matriz de salida del motor. Un observador de estados lineal para el sistema (3.4) (3.5) se describe como:

$$\begin{aligned} \dot{\hat{x}} &= A\hat{x} + Bu + l(y - \hat{y}) \\ \hat{y} &= C\hat{x} \end{aligned} \quad (3.8)$$

Donde \hat{x} es el vector de estados estimado, \hat{y} son las salidas estimadas, y son las salidas, B es la matriz de las entradas, u los vectores de entradas y l es la matriz de ganancia, la cual determina la convergencia del observador.

Ahora se define la dinámica del error de estimación \dot{e} en el sistema dado por el estado actual estimado \dot{x} menos el estado final estimado $\dot{\hat{x}}$.

$$\dot{e} = \dot{x} - \dot{\hat{x}} \quad (3.9)$$

La ecuación (3.6) que corresponde al motor y (3.8) que respresenta la ecuación del observador son reemplazadas en la ecuación (3.9).

$$\dot{e} = Ax + Bu - A\hat{x} - Bu - l(Cx - C\hat{x}) \quad (3.10)$$

Se restan los terminos y se factoriza la ecuación

$$\dot{e} = A(x - \hat{x}) - l \cdot C(x - \hat{x}) \quad (3.11)$$

Se define el error como $e = x - \hat{x}$ para ser reemplazado en la ecuación (3.11) de esta forma se obtiene la ecuación conocida como la dinámica del error del sistema.

$$\dot{e} = (A - l \cdot C)e \quad (3.12)$$

Si el sistema es observable, es posible determinar una matriz de ganancia l tal que $A - l \cdot C$ tenga valores deseados. Para determinar la matriz l se deben escoger sus autovalores, estos deben tener polos negativos garantizando la estabilidad del observador y deben estar mucho más alejados que los polos de la planta para asegurar que la dinámica del error sea más rápida que el sistema.

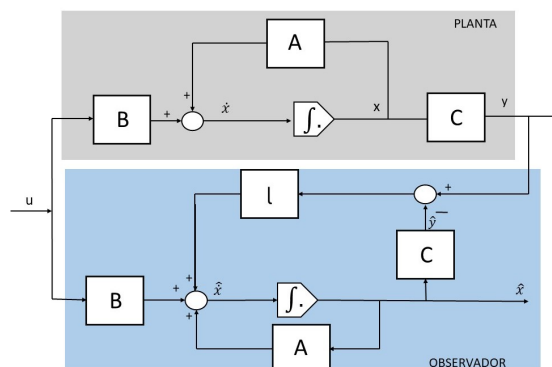


Figura 3.2: Diagrama de bloques de planta y observador de estados / Autor

3.3. Implementación del estimador de velocidad angular en el motor GM25-13CPR.

Inicialmente se verifica que el sistema sea observable, se hace uso de la ecuación (3.7) y se reemplazan los valores del motor GM25-13CPR.

$$O = \begin{bmatrix} 0 & 1 \\ -5,5 & -689,8 \end{bmatrix} \quad \begin{array}{l} Det(O) = 5,5 \\ Rango = 2 \end{array} \quad (3.13)$$

Al tener la matriz de observabilidad con su determinante diferente de 0 y su rango igual a un numero entero, se dice que este sistema es observable.

Ahora se debe escoger los valores de l para implementar el estimador de estados, estos deben tener polos negativos garantizando la estabilidad del observador y deben estar mucho más alejados que los polos de la planta para asegurar que la dinámica del error sea más rápida que el sistema. Inicialmente se identifica la ubicación de los polos del motor, mediante el comando $eig(A)$ en el software Matlab se halla la ubicación de estos polos los cuales estan ubicados en $(S + 59,2)(S + 635,7)$ y se verifica la respuesta del sistema ante una entrada paso.

Después se escoge la ubicación de los polos del observador teniendo en cuenta que el estimador debe ser 5 veces más rápido que la planta. Se propone $(S + 200)(S + 3000)$, obteniendo la ecuación deseada del sistema.

$$S^2 + 3200S + 600000 \quad (3.14)$$

Ahora se determina los autovalores de la matriz de ganancia l . Inicialmente se sustituye los valores del motor en la ecuación (3.12) obteniendo lo siguiente.

$$\dot{e} = \begin{bmatrix} -5,15 & 61798 \\ -5,52 & -689 \end{bmatrix} - \begin{bmatrix} l_1 \\ l_2 \end{bmatrix} \cdot \begin{bmatrix} 0 & 1 \end{bmatrix} \quad (3.15)$$

Se resuelve la ecuación (3.15) sin conocer los valores de l_1, l_2

$$\dot{e} = \begin{bmatrix} -5,15 & 61798 - l_1 \\ -5,52 & -689 - l_2 \end{bmatrix} \quad (3.16)$$

Se determina la ecuación del polinomio característico de (3.16) y se halla su determinante, siendo la ecuación del polinomio característico como $(P(\dot{e}) = \lambda \cdot I - \dot{e})$.

$$\lambda^2 + (695,15 + l_2) \cdot \lambda + 321807 + 5,15 \cdot l_2 - 5,52 \cdot l_1 \quad (3.17)$$

Ahora se iguala la ecuación del polinomio deseado (3.14) con la ecuación del polinomio característico (3.17) para despejar l_1, l_2 .

$$\begin{aligned}
\lambda^2 &= S^2 \\
(695,15 + l_2) &= 3200 \\
321807 + 5,15 \cdot l_2 - 5,52 \cdot l_1 &= 600000
\end{aligned}
\tag{3.18}$$

Se hallan los valores de l_1, l_2 obteniendo como resultado.

$$\begin{aligned}
l_1 &= -48,06 \\
l_2 &= 2504,8
\end{aligned}
\tag{3.19}$$

Estos valores de ganancia l son reemplazados en la ecuación (3.16) para determinar la dinámica del error \dot{e} del sistema.

$$\dot{e} = \begin{bmatrix} -5,15 & 61846,06 \\ -5,52 & -3193,8 \end{bmatrix}
\tag{3.20}$$

3.4. Validación del observador de estados.

3.4.1. Etapas de implementación en Simulink

En la simulación del observador de estados se implementa por medio del software Matlab utilizando la herramienta de Simulink, por medio de diagrama de bloques se diseña la planta y el observador de estados.

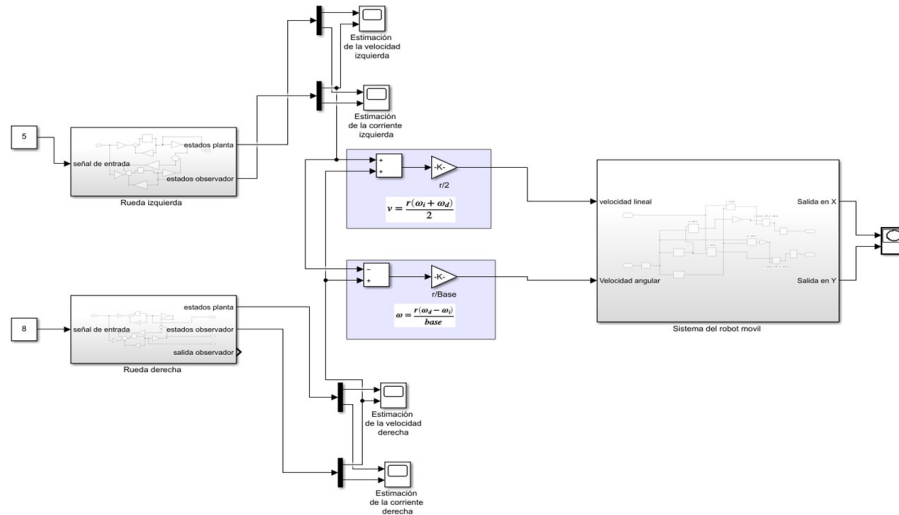
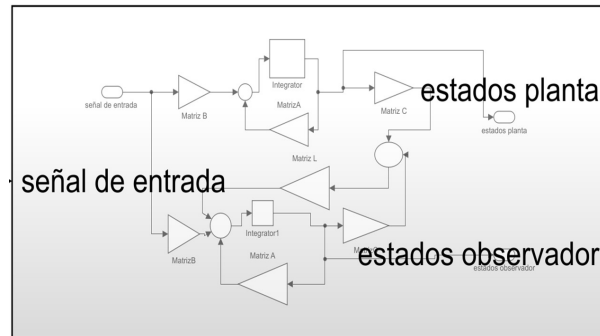


Figura 3.3: Esquema general del diagrama de bloques implementado en Simulink para el observador de estados. / Autor

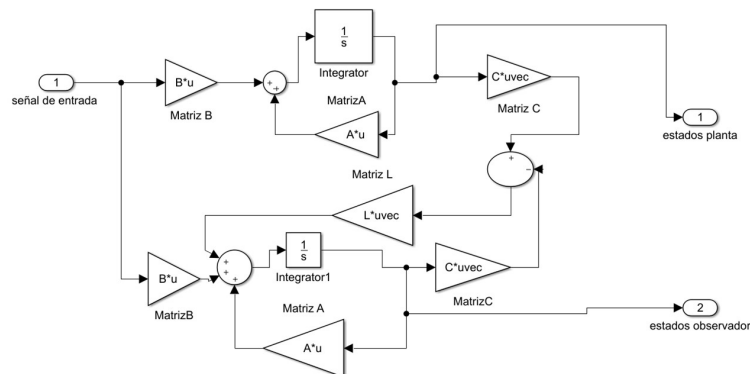
La Figura 3.3 representa el esquema general para la implementación del observador de estados. Inicialmente se ingresan constantes como entradas a cada motor del robot móvil, siendo el motor la planta. Este sistema contendrá el observador de estados, los cuales darán como salidas los estados de velocidad angular y corriente estimadas. Mediante las velocidades angulares de cada motor (ω_d, ω_i) se puede calcular las velocidades lineales y angulares del robot diferencial (v, ω), estas servirán como entradas para generar los movimientos del robot móvil y lograr verlas en un gráfico con respecto al plano global $X_I Y_I$.

Para describir el esquema del diagrama de bloques en Simulink se divide en 3 partes. Inicialmente se tiene la planta con su respectivo observador de estados para cada rueda del robot diferencial. Otra parte está enfocada en la obtención de las velocidades lineales y angulares del robot diferencial (2.3)(2.4). Finalmente se tiene el diagrama de bloques del movimiento del robot móvil con punto de control desplazado (2.10) en el marco global.

En la primera etapa de la simulación se tiene el motor de corriente directa junto con el observador de estados.



(a) Representación general



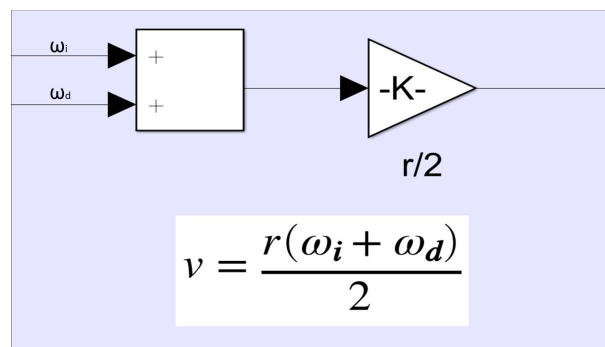
(b) Representación específica

Figura 3.4: Representación de la planta y el observador de estados del sistema.

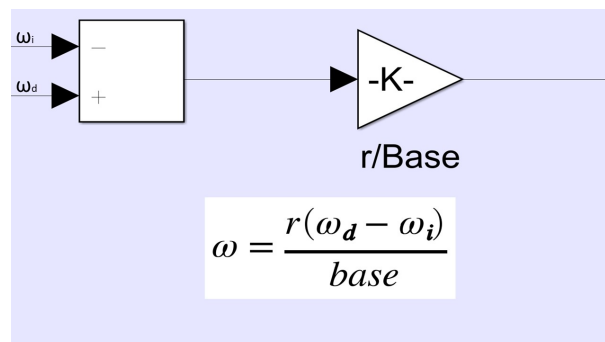
Se tiene una representación general, la misma que se observa en la Figura 3.3 y representación específica Figura 3.4 (b). En la representación específica se observa a detalle el diagrama de bloques del motor y el observador de estados implementado, donde se tiene 2 salidas. La salida estados planta corresponde a los estados de salida entregados por la planta en respuesta de algún tipo de entrada, en este caso los estados de salida son la velocidad angular y corriente del motor. La salida estados observador corresponde a los estados estimados por el observador

para ser comparados con los del motor.

La siguiente etapa es para calcular las velocidades v, ω del robot diferencial a partir de las velocidades angulares estimadas de cada motor.



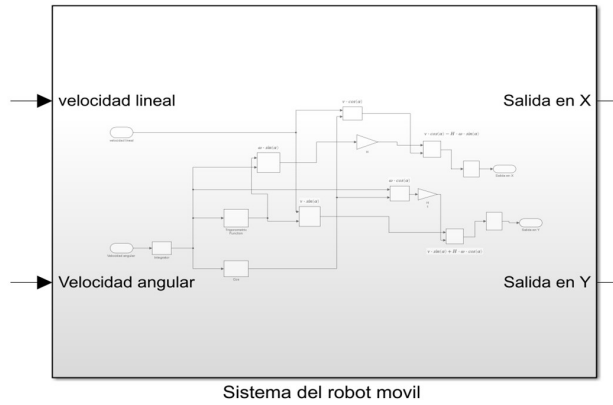
(a) Velocidad lineal



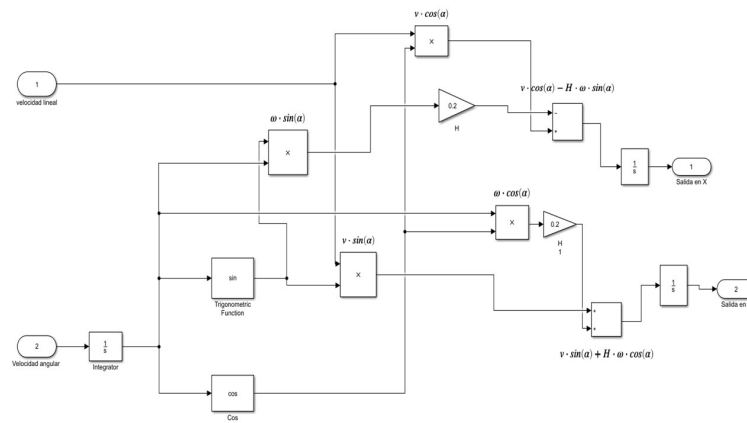
(b) Velocidad angular

Figura 3.5: Esquema de velocidad lineal y angular del robot diferencial implementado en Simulink/ Autor.

La etapa final representa los movimientos del robot móvil, se utiliza la ecuación (2.10), la cual describe el movimiento del robot diferencial con un punto de control desplazado en el marco de coordenadas globales. Se introducen diferentes velocidades lineales y angulares calculadas como se represento en la Figura 3.5.



(a) Representación general

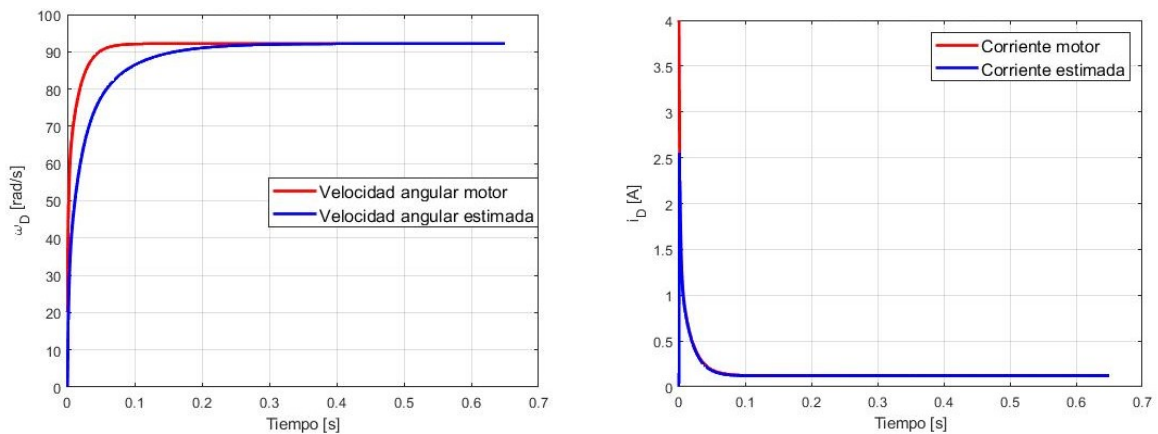


(b) Representación especifica

Figura 3.6: Esquema para representar los movimientos del robot movil/ Autor.

3.4.2. Simulaciones realizadas.

Inicialmente se muestra el funcionamiento del observador de estados, mediante la herramienta Simulink se implementa el diagrama de bloques representado en la Figura 3.4 correspondiente a la planta y el observador de estados. Se ingresa al sistema una constante como entrada y se compara los estados del motor con los estados del observador, estos serán la velocidad angular y la corriente.



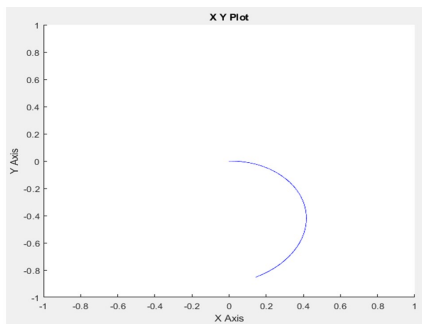
(a) Velocidad angular del motor y velocidad angular estimada

(b) Corriente del motor y corriente estimada

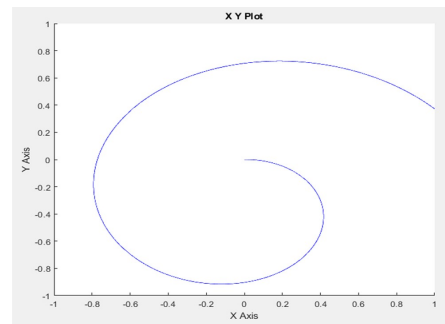
Figura 3.7: Simulación de los estados del motor y los estados estimados/ Autor.

La Figura 3.7 muestra como resultado la simulación de los estados de velocidad angular y corriente del motor con color rojo, por otro lado se muestra los estados estimados de la velocidad angular y corriente representados con en color azul.

Finalmente se realiza la simulación para observar los movimientos del robot diferencial, se implementa el diagrama general representado en la Figura 3.3 proponiendo diferentes cambios de orientación en el robot móvil. Inicialmente se representa el giro hacia la derecha estableciendo un voltaje de entrada en el motor izquierdo mayor que el motor derecho (motor izquierdo = 6 voltios , motor derecho = 4 voltios), se obtiene un trayecto representado en la Figura 3.8.



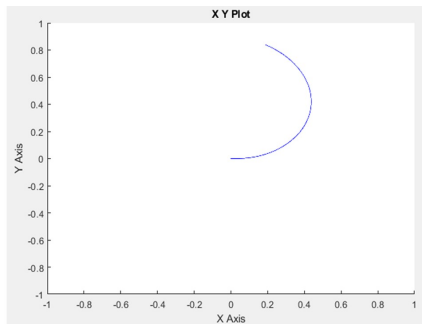
(a) Movimiento con tiempo se simulación $t_s = 2$



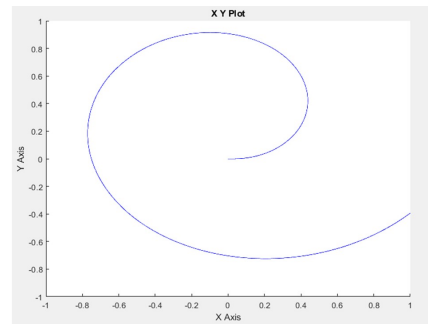
(b) Movimiento con tiempo se simulación $t_s = 8$

Figura 3.8: Movimiento del robot diferencial hacia la izquierda/ Autor.

Por otro lado se demuestra el cambio de orientación hacia la izquierda dejando el voltaje de entrada del motor derecho mayor que del motor izquierdo (motor izquierdo = 6 voltios, motor derecho = 4 voltios), se obtiene un trayecto representado en la Figura 3.9.



(a) Movimiento con tiempo de simulación $t_s = 2$



(b) Movimiento con tiempo de simulación $t_s = 8$

Figura 3.9: Movimiento del robot diferencial hacia la izquierda/ Autor.

Capítulo 4

COORDINACIÓN DE ROBOTS MOVILES DIFERENCIALES.

Este capítulo contiene información sobre el control no lineal en robots móviles diferenciales, tanto para realizar una trayectoria deseada como para la coordinación entre ellos. A continuación se define el modelo matemático del robot para llegar a una posición deseada junto con su ruta deseada, también especificar los valores de ganancia y el error de posición requerido para el seguimiento de rutas en el robot móvil. Una segunda parte se encarga del seguimiento y coordinación entre los robots móviles. El seguimiento es para que el robot líder siga una curva con una ecuación parametrizada. En la coordinación, el robot seguidor tiene la tarea de recibir los datos de la ruta del robot líder y se encarga de seguir la misma. Finalmente se realizan las simulaciones para posicionar el robot móvil a un punto deseado, seguir una ruta con curva parametrizada y coordinación de dos o más robots móviles.

4.1. Llevar al robot de un punto inicial a un punto final en el marco global.

4.1.1. Objetivo de control

El sistema representado en la ecuación (2.15) relaciona las velocidades lineales y angulares contenidas en la matriz U y las velocidades del trayecto del robot $X\dot{Y}_r$. Se realiza la inversa del sistema despejando la matriz de velocidades del robot, con el fin de ingresar posiciones y velocidades de referencia ($X\dot{Y}_{ref}, X\dot{Y}_{ref}$) en el algoritmo de control para calcular las velocidades lineales y angulares para llegar al punto deseado denotado en la ecuación (4.1).

$$U = J^{-1} \cdot X\dot{Y}_{ref} \quad (4.1)$$

Donde $X\dot{Y}_{ref}$ corresponde a la matriz de velocidades de referencia de la ruta que se desea seguir. Al escribir la ecuación (4.1) de forma matricial se observa que al calcular la matriz U en la ecuación (4.2), se tiene como resultado una matriz de 2x1 cuyo valor en la fila 1, columna 1 corresponde a la velocidad lineal v y el valor de la fila 2, columna 1 corresponde a la velocidad angular ω .

$$\begin{bmatrix} v \\ \omega \end{bmatrix} = \begin{bmatrix} \cos(\alpha) & -H \cdot \sin(\alpha) \\ \sin(\alpha) & H \cdot \cos(\alpha) \end{bmatrix}^{-1} \cdot \begin{bmatrix} \dot{X}_{ref} \\ \dot{Y}_{ref} \end{bmatrix} \quad (4.2)$$

De esta forma se reemplazan las velocidades lineales y angulares en la ecuación (2.10) correspondiente al modelo cinemático del robot móvil con un punto de control desplazado para generar el movimiento del robot diferencial.

4.1.2. Estructura de controlador

En el seguimiento de un punto en el marco global, sin ninguna especificación adicional se realiza con la ecuación cinemática del robot (4.2). Sin embargo, el sistema no converge al valor deseado, inicialmente se define el error de posición ε entre las posiciones deseadas $(X_{ref}, Y_{ref} = XY_{ref})$ y la posición inicial del robot con punto de control desplazado (Xr, Yr) representado en la ecuación (4.3).

$$\varepsilon = \begin{bmatrix} X_{ref} - Xr \\ Y_{ref} - Yr \end{bmatrix} \quad (4.3)$$

Para que el sistema converja con mayor precisión al valor deseado, se necesita de una constante de ganancia k . Esta ganancia junto con el error de posición representa la ley de control presentada en la ecuación (4.4).

$$U = J^{-1} \cdot (X\dot{Y}_{ref} + k \cdot \varepsilon) \quad (4.4)$$

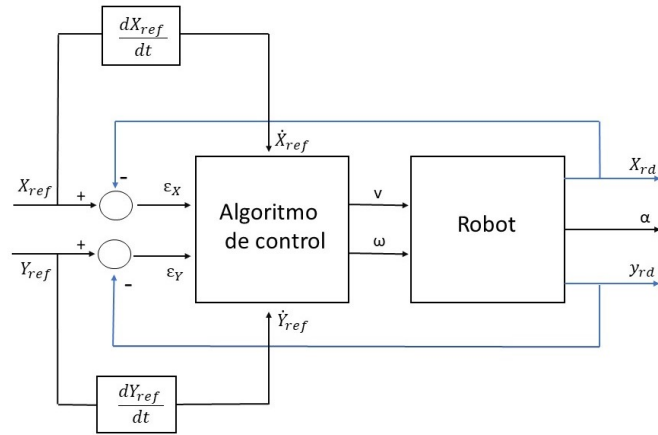


Figura 4.1: Diagrama del algoritmo de control. / Autor

En la Figura 4.1 se observa el diagrama general del control introducido al sistema. Inicialmente se ingresan señales de referencia, en este caso son las posiciones de referencia (X_{ref}, Y_{ref}) . La posición de referencia junto con la velocidad de referencia $(\dot{X}_{ref}, \dot{Y}_{ref})$ es ingresada en el algoritmo de control para calcular las velocidades lineales y angulares necesarias para posicionar al robot en las posiciones de referencia. Finalmente se ingresan estas velocidades lineales y angulares al modelo cinemático del robot móvil para generar el movimiento del trayecto realizado, arrojando las posiciones realizadas en el trayecto por el robot.

En la ecuación (4.4) se incluye la posición de interés y la posición inicial del robot con punto de control desplazado en el error ϵ , en este caso solo se desea llegar a un punto en el marco global sin una trayectoria específica, es decir, se utiliza coordenadas rectangulares para determinar la trayectoria del robot. Las velocidades de referencia del robot $\dot{X}_{ref}, \dot{Y}_{ref}$ son las derivadas de las posiciones de referencia, en las coordenadas rectangulares se tiene estas posiciones de referencia como valores constantes, al derivar estos valores para obtener las velocidades de referencia, estas pueden ser reemplazadas por cero.

$$U = J^{-1} \cdot (k \cdot \varepsilon) \quad (4.5)$$

De esta forma se obtiene la ecuación (4.5) con la cual se calcula las velocidades lineales y angulares contenidas en la matriz U para realizar la ruta deseada.

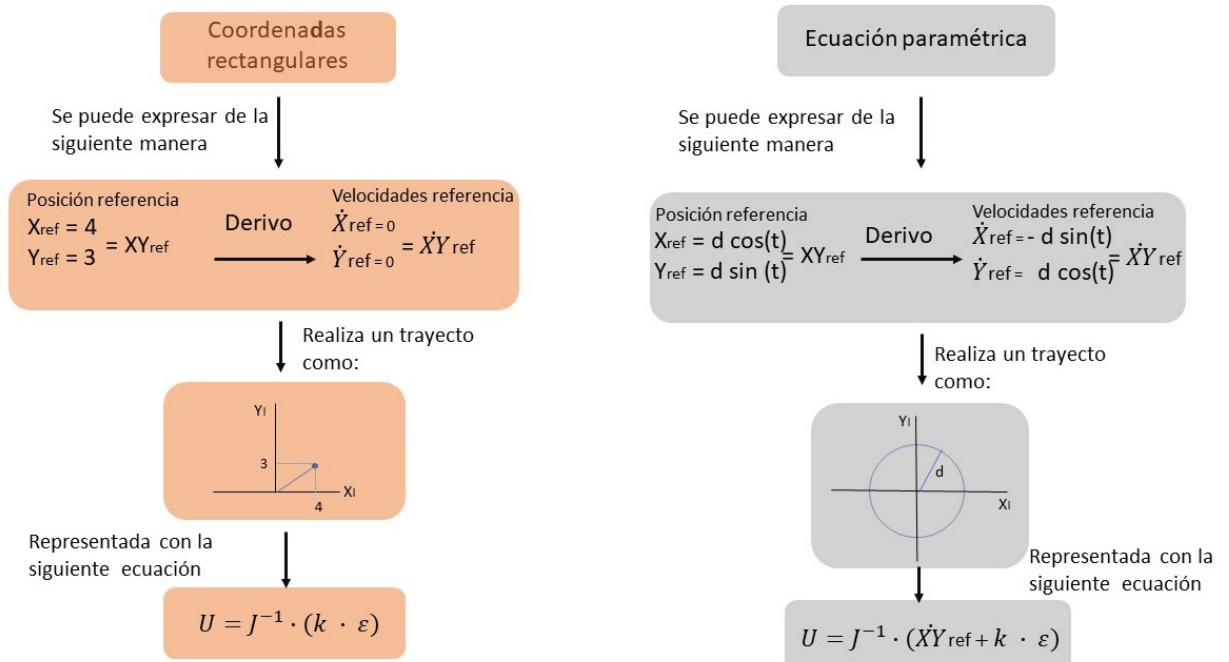


Figura 4.2: Diagrama explicando coordenadas rectangulares y coordenadas mediante ecuación paramétrica / Autor

Observe el diagrama en la Figura 4.2, donde se presenta el seguimiento de una ruta mediante coordenadas rectangulares y ecuaciones paramétricas. Se sabe que para el seguimiento de una ruta deseada se necesita la posición de referencia X_{ref} y la velocidad de referencia \dot{X}_{ref} , en el diagrama se propone la posición de referencia en coordenadas rectangulares como $X_{ref} = 4, Y_{ref} = 3$, también se necesita las velocidades de referencia para realizar la ruta, sin embargo, como se observa en el ejemplo planteado en el diagrama, para coordenadas rectangulares, por tener posiciones de referencia como valores constantes, al derivarlos para obtener sus velocidades de referencia, estos pueden ser reemplazados por cero. En el seguimiento de ruta mediante ecuaciones paramétricas se tiene posiciones de referencia como $X_{ref} = \sin(\alpha), Y_{ref} = \cos(\alpha)$, estos valores no son constantes y su derivada es diferente de cero, por lo anterior se establece que para el seguimiento de un punto de posición se desprecia la velocidad de referencia obteniendo la ecuación (4.5).

También se describe el diagrama de control en el motor de corriente directa para una implementación real. Este se describe en la Figura 4.3

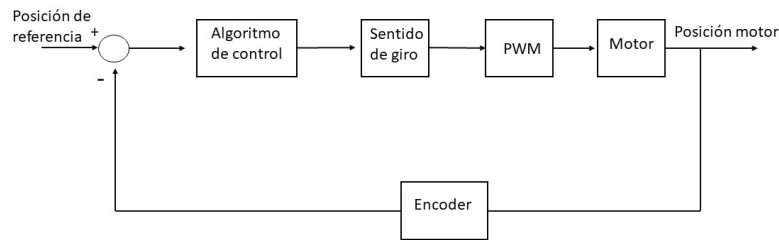
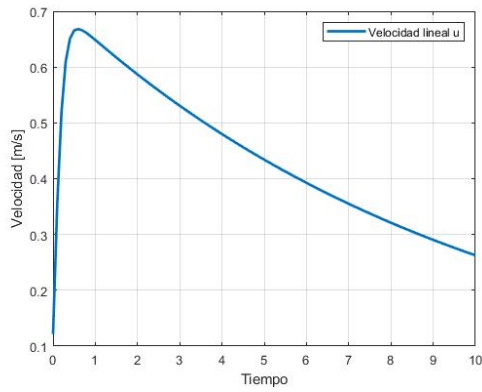


Figura 4.3: Diagrama de algoritmo de control en el motor. / Autor

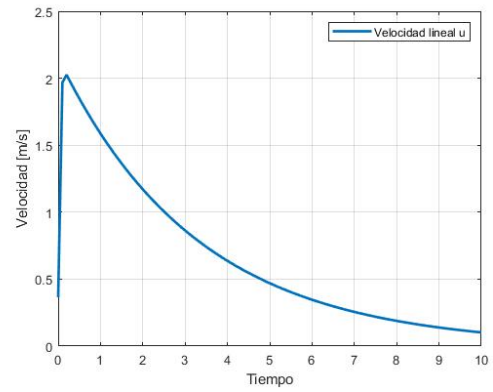
Inicialmente en la Figura 4.3 se introduce una posición de referencia a la cual se desea llegar. Se necesita un algoritmo de control para la posición del motor, un selector de sentido de giro, este puede ser un puente H, un modulo PWM el cual module la velocidad del motor y un sensor como el encoder, este permitirá medir la velocidad y posición del motor para después realimentarla con la posición de referencia.

4.2. Ajuste de ganancia k

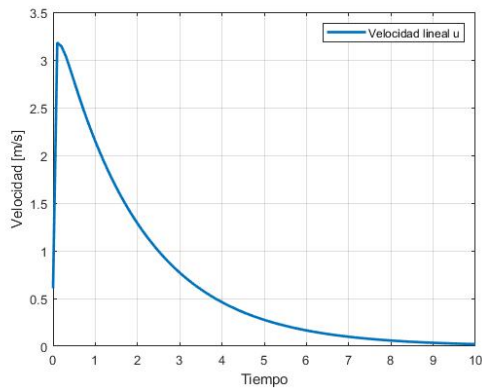
Se realiza simulaciones para establecer una ganancia k adecuada para el control, se efectúa varias simulaciones en los puntos deseados $X_{ref} = 6$, $Y_{ref} = -4$ con diferentes ganancias k para observar la acción de control en la velocidad lineal presentada en la Figura 4.4 y la velocidad angular en la Figura 4.5.



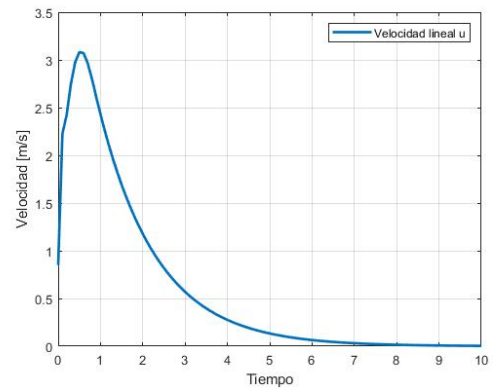
(a) $k = 0.1$



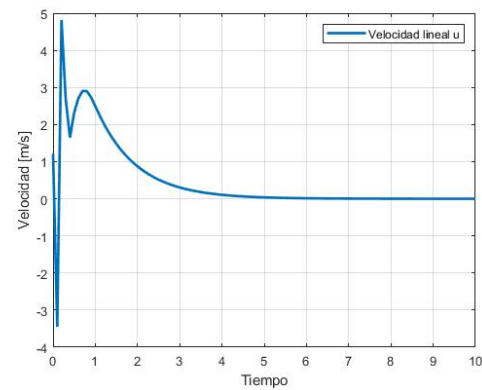
(b) $k = 0.3$



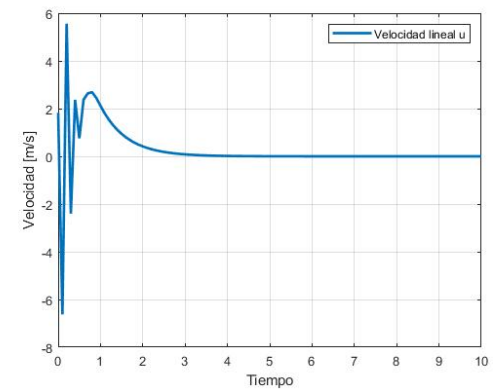
(c) $k = 0.5$



(d) $k = 0.7$

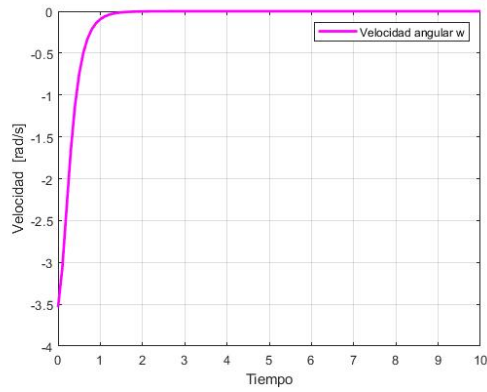


(e) $k = 1$

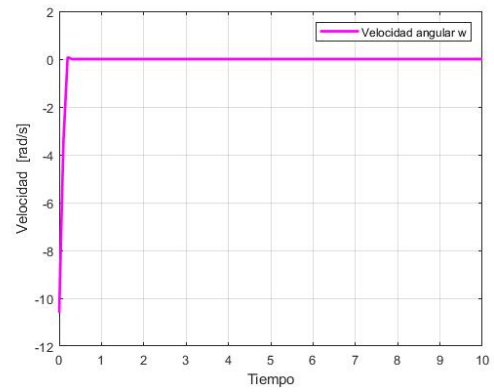


(f) $k = 1.5$

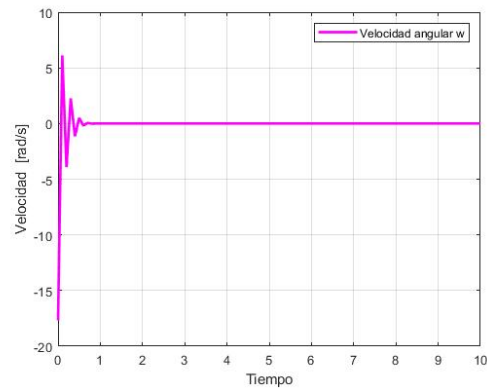
Figura 4.4: Acción de control en la velocidad lineal del robot diferencial al posicionarse en los puntos $X_{ref} = 6$, $Y_{ref} = -4$ / Autor.



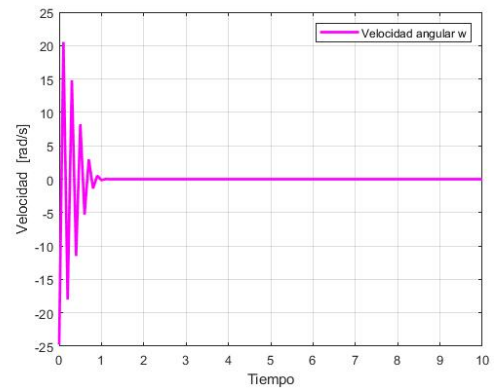
(a) $k = 0.1$



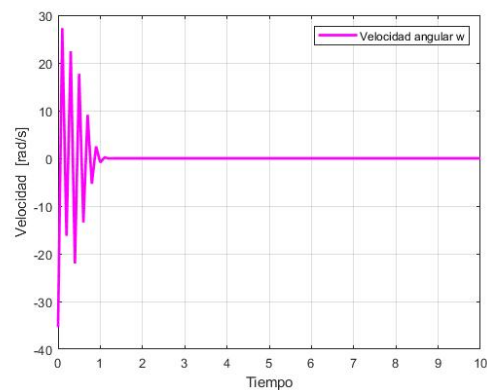
(b) $k = 0.3$



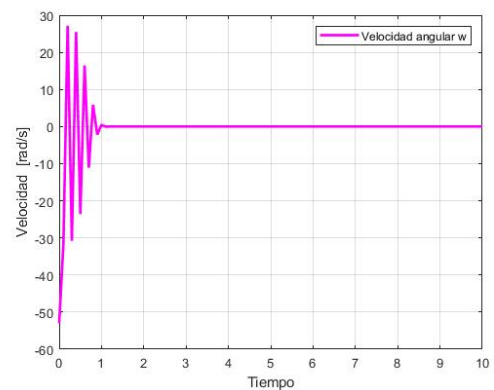
(c) $k = 0.5$



(d) $k = 0.7$



(e) $k = 1$



(f) $k = 1.5$

Figura 4.5: Acción de control en la velocidad angular del robot diferencial al posicionarse en los puntos $X_{ref} = 6$, $Y_{ref} = -4$ / Autor.

Observe en la Figura 4.4 el comportamiento en la acción de control de la velocidad lineal del robot diferencial, cambia a medida que aumenta la ganancia k . La velocidad lineal aumenta cada vez que se aumenta esta ganancia, observe el índice (a) con $k = 0,1$, muestra una velocidad lineal máxima de aproximadamente 0.67 m/s y esta disminuye hasta cero de forma uniforme. En el índice (f) con $k = 1.5$, se presenta una velocidad máxima de 5.8 m/s , además, se observa cambios repentinos en la velocidad lineal en los primeros instantes de tiempo en la simulación.

En la Figura 4.5 se observa la acción de control en la velocidad angular del robot móvil, se presenta un comportamiento similar al de la velocidad lineal, a medida que se aumenta la ganancia k , la velocidad angular presenta cambios repentinos. Observe en el índice (a) con $k = 0,1$ como la velocidad angular empieza desde -3.5 m/s y como esta converge a cero de una forma suave sin presentar cambios bruscos. El índice (b) con $k = 1,5$ presenta una velocidad angular que empieza desde -51 m/s y como esta converge a cero pero con cambios repentinos en los primeros instantes de tiempo de simulación, estos cambios repentinos se ven reflejados en el robot móvil donde presenta pequeñas oscilaciones en la orientación del robot y genera una ruta presentada en la Figura 4.6.

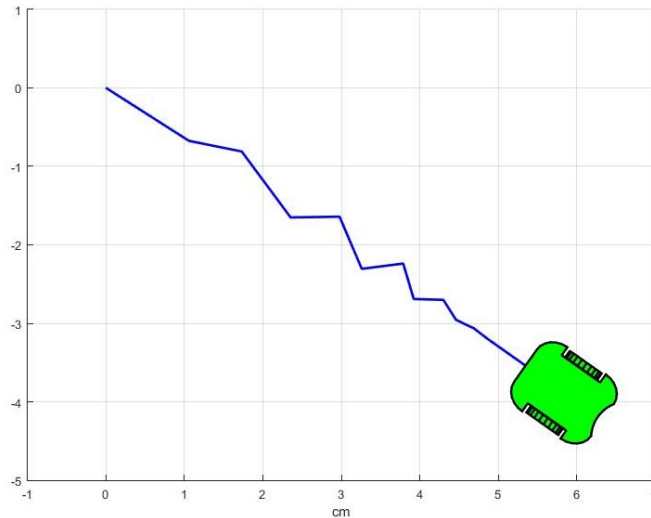


Figura 4.6: Simulación del robot diferencial para posicionarse en los puntos $X_{ref} = 6$, $Y_{ref} = -4$ / Autor

De acuerdo a los resultados presentados en las Figuras 4.4, 4.5 y 4.6 se propone un intervalo de valores de ganancia k tales que presente más estabilidad en el trayecto del robot móvil y no genere cambios repentinos en la acción de control de la velocidad lineal y angular, este intervalo esta dado entre valores de $k = [0.1, 0.4]$. Este intervalo de valores se establece para la ganancia en el control de posición en un punto deseado, seguimiento de ruta con ecuación parametrizada y la coordinación entre dos o más robots diferenciales.

4.3. Seguimiento de ruta con curva parametrizada en el robot móvil.

Anteriormente se tenía un seguimiento de puntos en el plano cartesiano, donde se tiene un punto inicial y se deseaba llegar a un punto final sin ninguna especificación adicional. Para el seguimiento de una trayectoria en el robot móvil se tendrá una trayectoria de referencia, en este caso se utiliza trayectorias de curvas con representación paramétrica teniendo el tiempo como una restricción, es decir, el robot móvil debe seguir una trayectoria para llegar al punto deseado en un tiempo dado.

Una representación paramétrica del robot móvil permite representar curvas en el plano cartesiano, en el anterior ejemplo se tenía un punto específico en el marco global conocido como coordenadas rectangulares que estarán en función de una con respecto a la otra. En las ecuaciones paramétricas se consideran ambas coordenadas como funciones de una nueva variable también denominada parámetro β , este parámetro permite representar una curva con valores que recorren intervalos de números reales.

Un ejemplo sería describir la trayectoria de un círculo, donde escogemos un punto P del círculo que estará a una distancia d con un ángulo t el cual también definiría la orientación del trayecto.

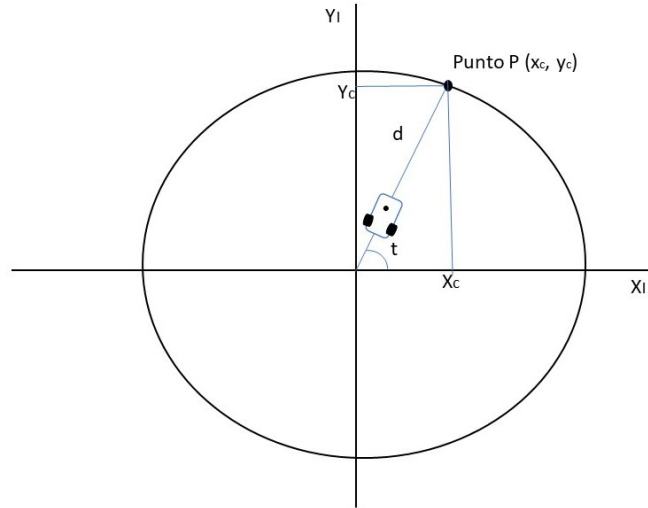


Figura 4.7: Trayecto de círculo en el marco global/ Autor

Para obtener la trayectoria del círculo se calcula la distancia del punto P en el eje X_I, Y_I del marco global, obteniendo así la ecuación paramétrica. Sabemos que el coseno del ángulo será igual al cateto adyacente sobre la hipotenusa y el seno del ángulo será el cateto opuesto sobre la hipotenusa.

$$\begin{aligned} \cos(t) &= \frac{X_c}{d} \\ \sin(t) &= \frac{Y_c}{d} \end{aligned} \quad (4.6)$$

Ahora se despeja X_c, Y_c que serán también las posiciones de referencia en la ecuación paramétrica.

$$\begin{bmatrix} X_{refc} \\ Y_{refc} \end{bmatrix} = \begin{bmatrix} d \cdot \cos(t) \\ d \cdot \sin(t) \end{bmatrix} = \begin{bmatrix} X_c \\ Y_c \end{bmatrix} = XY_c \quad (4.7)$$

Ahora se deriva con respecto al tiempo, obteniendo las velocidades de interés para generar la trayectoria deseada del robot móvil. Donde \dot{XY}_c es la matriz de velocidades de interés para el trayecto del círculo.

$$\dot{XY}_c = \begin{bmatrix} \dot{Xref}_c \\ \dot{Yref}_c \end{bmatrix} = \begin{bmatrix} -d \cdot \sin(t) \\ d \cdot \cos(t) \end{bmatrix} \quad (4.8)$$

A diferencia de la ecuación (4.5), esta cuenta con posiciones de interés no lineales y que su derivada es diferente de 0 pertenecientes a la ecuación paramétrica del trayecto XY_c , al derivar estas posiciones se obtiene las velocidades de interés del trayecto \dot{XY}_c . Se usa la ecuación (4.4) tomando en cuenta las velocidades de interés del círculo para realizar la ley de control, mediante el error de posición ε y velocidades de interés (\dot{XY}_c), para luego calcular las velocidades lineales y angulares con las que realiza el trayecto y están contenidas en la matriz U .

$$U = J^{-1} \cdot (\dot{XY}_c + k \cdot \varepsilon) \quad (4.9)$$

4.4. Coordinación de robots móviles en estructura líder - seguidor

En la coordinación de dos o más robots se aplica el mismo algoritmo de control mencionado anteriormente para las rutas, donde el robot líder tendrá posiciones y velocidades de interés del círculo (XY_c, \dot{XY}_c) y los robots seguidores tienen posiciones y velocidades de interés basado en las posiciones finales del robot líder.

Se realiza un ejemplo para que un robot líder X_{r1}, Y_{r1} siga la ruta de un círculo y un robot seguidor X_{r2}, Y_{r2} imite su trayectoria. Inicialmente se realiza la ruta para que el robot líder realice un círculo, para ello se debe tener en cuenta las posiciones de interés (XY_c), ecuación (4.7) y las velocidades de interés para generar la trayectoria del círculo (\dot{XY}_c) ecuación (4.8).

Se calcula el error de posición, el cual está dado por el punto de interés menos el punto inicial del robot móvil con punto de control desplazado ecuación (4.3). Los puntos iniciales y la orientación inicial en los que empieza a moverse el robot líder en el marco global son X_1, Y_1, α dejándolos con valores iguales a 0, indicando que el robot móvil empezará a moverse desde las coordenadas 0,0 del marco global.

$$\begin{aligned} X_1 &= 0 \\ Y_1 &= 0 \\ \alpha_1 &= 0 \end{aligned} \tag{4.10}$$

Se calcula las posiciones iniciales de los robots con un punto de control desplazado X_{r1}, Y_{r1} basado en las ecuaciones (2.8).

$$\begin{aligned} X_{r1} &= X_1 + H \cdot \cos(\alpha_1) \\ Y_{r1} &= Y_1 + H \cdot \sin(\alpha_1) \end{aligned} \tag{4.11}$$

Con las posiciones iniciales del robot con un punto de control desplazado X_{r1}, Y_{r1} y las posiciones de referencia del círculo X_{refc}, Y_{refc} se puede calcular el error de posición ε_1 para el robot líder.

$$\varepsilon_1 = \begin{bmatrix} X_{refc} - X_{r1} \\ Y_{refc} - Y_{r1} \end{bmatrix} \tag{4.12}$$

Al calcular el error de posición, mediante la ecuación (4.9) se obtiene las velocidades lineales y angulares que debe tener el robot líder para seguir esta ruta, se implementa una ganancia k , la matriz de error del robot líder ε_1 , la matriz de velocidades de interés para seguir la trayectoria del círculo $X\dot{Y}r_c$ y la matriz de relación J^{-1} para obtener la matriz de velocidades U_1 utilizadas para realizar la ruta del círculo mostrada en la ecuación (4.13).

$$U_1 = J^{-1}(k \cdot \varepsilon_1 + X\dot{Y}r_c) \quad (4.13)$$

Donde $X\dot{Y}r_c$ corresponde a la matriz de velocidades de interés para generar la ruta del círculo y la matriz U_1 que representa las velocidades lineales y angulares con las que realiza la ruta el robot líder. (4.2)

Sabiendo que las velocidades lineales y angulares del robot diferencial están contenidas en la matriz U_1 , estas son reemplazadas en la ecuación para generar el movimiento del robot líder (2.15).

$$\begin{aligned} \dot{X}r_1 &= v_1 \cdot \cos(\alpha_1) - H \cdot \omega_1 \sin(\alpha_1) \\ \dot{Y}r_1 &= v_1 \cdot \sin(\alpha_1) + H \cdot \omega_1 \cos(\alpha_1) \\ \dot{\alpha}_1 &= \omega_1 \end{aligned} \quad (4.14)$$

También se sabe que $\dot{X} = v \cdot \cos(\alpha)$, $\dot{Y} = v \cdot \sin(\alpha)$, entonces se simplifica la ecuación quedando de la siguiente forma.

$$\begin{aligned}
\dot{X}r_1 &= \dot{X}_1 - H \cdot \omega_1 \sin(\alpha_1) \\
\dot{Y}r_1 &= \dot{Y}_1 + H \cdot \omega_1 \cos(\alpha_1) \\
\dot{\alpha}_1 &= \omega_1
\end{aligned} \tag{4.15}$$

Se integra numéricamente las velocidades del robot para obtener las posiciones del robot generadas en el trayecto. Para una mejor notación se establece que estas posiciones finales para el caso del robot líder serán Xrd_1, Yrd_1

$$\begin{aligned}
Xrd_1 &= X_1 - H \cdot \cos(\alpha_1) \\
Yrd_1 &= Y_1 - H \cdot \sin(\alpha_1)
\end{aligned} \tag{4.16}$$

Ya realizada la trayectoria que debe seguir el robot líder, se procede a realizar la trayectoria que debe imitar el seguidor (Xr_2, Yr_2). En este caso también se calcula el error de posición para el seguidor ε_2 teniendo en cuenta que sus puntos de interés (X_{ref2}, Y_{ref2}), serán las posiciones finales generadas del trayecto por el robot líder (Xrd_1, Yrd_1).

$$\begin{aligned}
X_{ref2} &= Xrd_1 \\
Y_{ref2} &= Yrd_1
\end{aligned} \tag{4.17}$$

Teniendo los puntos de interés y dejando los mismos puntos iniciales donde se empieza a mover el robot líder, se calcula el error de posición del robot seguidor.

$$\varepsilon_2 = \begin{bmatrix} Xrd_1 - Xr_2 \\ Yrd_1 - Yr_2 \end{bmatrix} \tag{4.18}$$

Ahora se calcula las velocidades lineales y angulares contenidas en la matriz U_2 con las que realiza el trayecto (4.9). En este caso se necesita las velocidades de interés (XY_{ref2}) del robot seguidor, de este modo se deriva las posiciones finales del robot líder que a su vez son las posiciones de interés del robot seguidor (Xrd_1, Yrd_1).

$$XY_{ref2} = \begin{bmatrix} \dot{X}r_2 = \dot{X}rd_1 \\ \dot{Y}r_2 = \dot{Y}rd_1 \end{bmatrix} \quad (4.19)$$

$$U_2 = j_2^{-1}(k \cdot \varepsilon_2 + XY_{ref2}) \quad (4.20)$$

Las velocidades obtenidas son remplazadas en la ecuación (2.15) para generar el movimiento del robot seguidor.

$$\begin{aligned} \dot{X}r_2 &= v_2 \cdot \cos(\alpha_2) - H \cdot \omega_2 \sin(\alpha_2) \\ \dot{Y}r_2 &= v_2 \cdot \sin(\alpha_2) + H \cdot \omega_2 \cos(\alpha_2) \\ \dot{\alpha}_2 &= \omega_2 \end{aligned} \quad (4.21)$$

Sabiendo que $\dot{X} = v \cdot \cos(\alpha)$, $\dot{Y} = v \cdot \sin(\alpha)$, se simplifica la ecuación quedando de la siguiente forma.

$$\begin{aligned} \dot{X}r_2 &= \dot{X}_2 - H \cdot \omega_2 \cdot \sin(\alpha_2) \\ \dot{Y}r_2 &= \dot{Y}_2 + H \cdot \omega_2 \cdot \cos(\alpha_2) \\ \dot{\alpha}_2 &= \omega_2 \end{aligned} \quad (4.22)$$

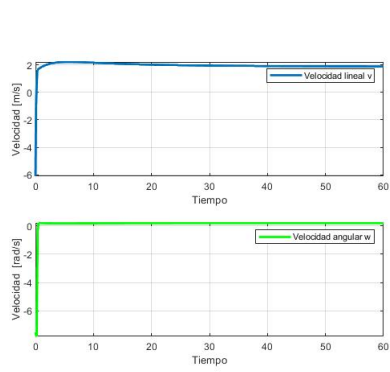
Se integra numéricamente las velocidades del robot para obtener las posiciones del robot generadas en el trayecto. Para una mejor notación se establece que estas posiciones finales para el

caso del robot seguidor serán Xrd_2, Yrd_2

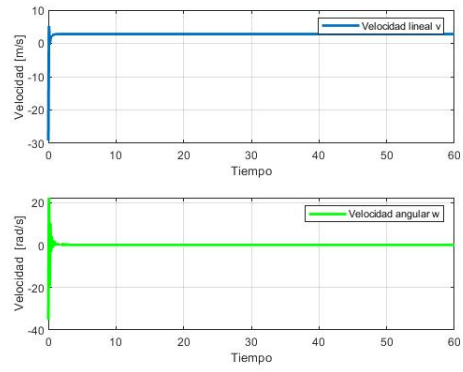
$$\begin{aligned} Xrd_2 &= X_2 - H \cdot \cos(\alpha_2) \\ Yrd_2 &= Y_2 - H \cdot \sin(\alpha_2) \end{aligned} \tag{4.23}$$

4.4.1. Ajuste de ganancia k para el seguimiento de una trayectoria

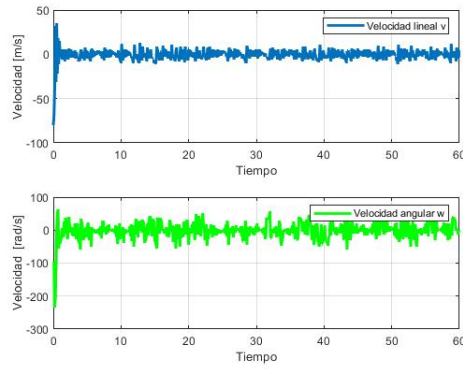
Se realiza simulaciones para establecer una ganancia k adecuada para el seguimiento de trayectorias, se efectúa varias simulaciones con diferentes ganancias k , se observa la acción de control en la velocidad lineal y angular en la Figura 4.8.



(a) $k = 0.5$



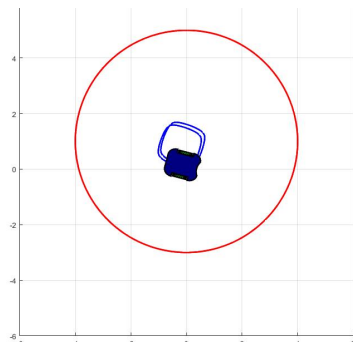
(b) $k = 6$



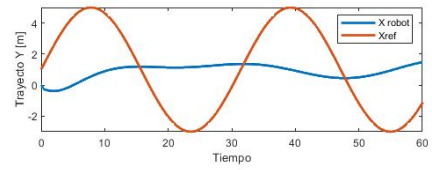
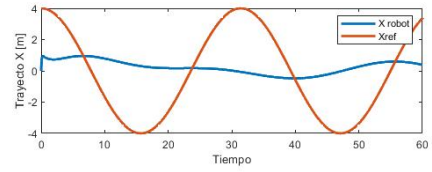
(c) $k = 18$

Figura 4.8: Acciones de control en el seguimiento de trayectoria / Autor.

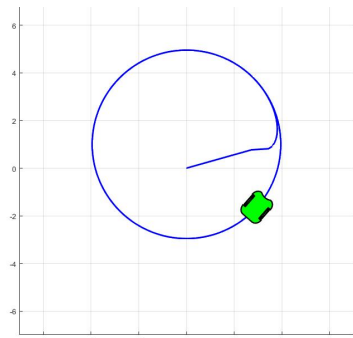
Así mismo se observa el trayecto realizado en la Figura 4.9 , de este modo se establece el rango de la ganancia k con respecto a lo obtenido en las acciones de control de la velocidad lineal, angular y el trayecto realizado por el robot móvil.



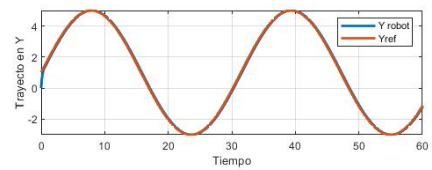
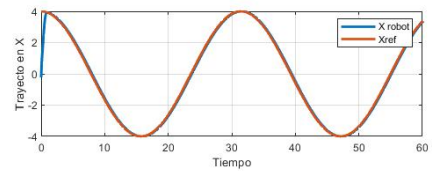
(a) $k = 0.5$



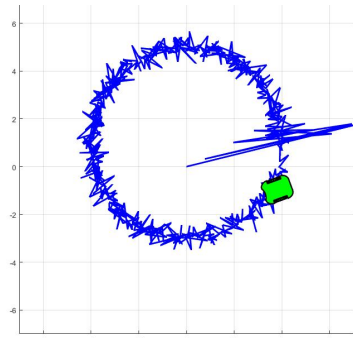
(b) $k = 0.5$



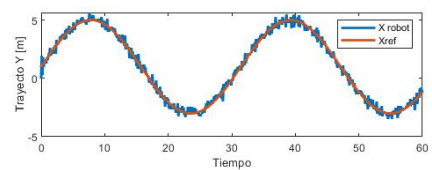
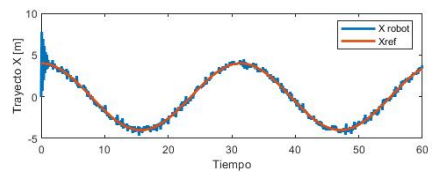
(c) $k = 6$



(d) $k = 6$



(e) $k = 18$



(f) $k = 18$

Figura 4.9: Trayectoria realizada con diferentes valores de ganancia k / Autor.

Basado en las acciones de control y el trayecto descrito en las Figuras 4.8 4.9, se propone una ganancia $k = [2 - 6.5]$ para el seguimiento de trayectorias.

4.5. Validación del sistema

4.5.1. Algoritmos implementados.

Inicialmente se realiza la simulación para el seguimiento de un punto de referencia en el marco global, el robot móvil deberá seguir este punto sin ninguna especificación adicional, es decir, el robot deberá ir hasta el punto deseado en línea recta sin la necesidad de cambiar su orientación.

Esta simulación se realizará de acuerdo a los puntos de referencia empleados X_{ref}, Y_{ref} , junto con sus posiciones iniciales que normalmente en la simulación se colocaron como $X = 0, Y = 0$ para realizar el error en la posición y así emplear la ley de control descrita en la ecuación (4.5) para determinar el trayecto generado sin ningún cambio de orientación en un único robot diferencial.

Algoritmo 3: Algoritmo de movimiento de robot móvil a un punto deseado primera parte

◇ Se establece el tiempo de simulación t con un tiempo de muestreo ts .

$ts = 0.1;$

$t = 0:ts:20;$

◇ Establecer los puntos deseados en el marco global

$X_{ref};$

$Y_{ref};$

◇ Se introduce las posiciones y orientación iniciales, en las cuales empieza a moverse el robot diferencial.

$x(1) = 0;$

$y(1) = 0;$

$\alpha(1) = 0;$

$H =$ distancia en metros del punto de control desplazado del centro de masa;

◇ Se establece la posición inicial del robot móvil con un punto de control desplazado.

(2.8)

$X_r(1) = x(1) + H \cdot \cos(\alpha(1));$

$Y_r(1) = y(1) + H \cdot \sin(\alpha(1));$

Algoritmo 4: Algoritmo de movimiento de robot móvil a un punto deseado segunda parte

for $k = 1:length(t)$ **do**

◇ Se calcula el error de posición basado en la ecuación (4.3)

$xre(k) = xref - xr(k);$

$yre(k) = yref - yr(k);$

$error = [xre(k);yre(k)];$

◇ Se introduce las matrices J y ganancia k para ser remplazadas en la ecuación (4.5) y calcular la matriz de velocidades U

$J = [\cos(\phi(k)) - a*\sin(\alpha(k));\sin(\alpha(k)) a*\cos(\alpha(k))];$

$K = [0.5 \ 0;0 \ 0.5]; U = \text{inv}(J)*K*error;$

◇ Al resolver la ecuación (4.5), arroja la matriz U en la cual está contenida la velocidad lineal y angular ver ecuación (4.2).

$v(k) = U(1);$

$w(k) = U(2);$

◇ Se calcula las velocidades de interés a partir de las velocidades lineales y angulares remplazadas en la ecuación (2.10)

$xpunto(k) = v(k)*\cos(\alpha(k)) - H*w(k)*\sin(\alpha(k));$

$ypunto(k) = v(k)*\sin(\alpha(k)) + H*w(k)*\cos(\alpha(k));$

$alphapunto = w(k);$

◇ Integración numérica para obtener las posiciones a partir de las velocidades de interés.

$x(k+1) = x(k) + ts*xpunto(k);$

$y(k+1) = y(k) + ts*ypunto(k);$

$\alpha(k+1) = \alpha(k) + ts*alphapunto;$

$x(k+1) = xr(k+1) - a*\cos(\alpha(k+1));$

$y(k+1) = yr(k+1) - a*\sin(\alpha(k+1));$

En la segunda simulación se realiza para que un robot líder siga diferentes trayectorias (circulo, seno, coseno) y un robot seguidor imite la trayectoria dada. En este caso se necesita de las ecuaciones con curvas paramétricas especificadas en la ecuación (4.7) junto con sus derivadas, las cuales darán información sobre las velocidades y posiciones de interés para el robot líder, mediante la ley de control planteada en la ecuación (4.9) se calculan las velocidades lineales y angulares para determinar las posiciones finales generadas en el trayecto del circulo.

Algoritmo 5: Simulación para el seguimiento y coordinación de trayectorias. Primera parte.

◇ Se establece el tiempo de simulación t con un tiempo de muestreo ts .
 $ts = 0.1;$
 $t = 0:ts:60;$
 ◇ se establece la ecuación paramétrica la cual va a seguir el robot líder (4.7)
 X_{ref} = ecuación paramétrica en X;
 Y_{ref} = ecuación paramétrica en Y ;
 ◇ Se deriva la ecuación paramétrica obteniendo las velocidades de interés del robot líder
 x_{refd} = derivada de X_{ref} ;
 y_{refd} = derivada de Y_{ref} ;
 ◇ Se introduce las posiciones y orientación iniciales, en las cuales empieza a moverse el robot líder.
 $x(1) = 0;$
 $y(1) = 0;$
 $\alpha(1) = 0;$
 H = distancia en metros del punto de control desplazado del centro de masa;
 ◇ Se establece la posición inicial del robot móvil con un punto de control desplazado.
 (2.8)
 $X_r(1) = x(1) + H * \cos(\alpha(1));$
 $Y_r(1) = y(1) + H * \sin(\alpha(1));$

Algoritmo 6: Simulación para el seguimiento y coordinación de trayectorias. Segunda parte.

◇ Ecuaciones para realizar el error, la ley de control y el cálculo para las velocidades del trayecto del robot líder.

for k=1:length(t)

◇ Se calcula el error en XY basado en la ecuación (4.3)

xre(k) = xref - xr(k);

yre(k) = yref - yr(k);

error = [xre(k);yre(k)];

◇ Se introduce las matrices J , la matriz $X\dot{Y}r$ y la ganancia k para ser reemplazadas en la ecuación (4.9) y calcular la matriz de velocidades U

$J = [\cos(\alpha(k)) - a*\sin(\alpha(k)); \sin(\alpha(k)) \ a*\cos(\alpha(k))];$

$K = [0.5 \ 0; 0 \ 0.5];$

xyp = [xrefd(k);yrefd(k)];

$v = \text{inv}(J)*(xyp + K*error);$

◇ Al resolver la ecuación (4.5), arroja la matriz U en la cual está contenida la velocidad lineal y angular ver ecuación (4.2).

$v(k) = U(1);$

$w(k) = U(2);$

◇ Se calcula las velocidades de interés a partir de las velocidades lineales y angulares reemplazadas en la ecuación (2.10)

xpunto(k) = $v(k)*\cos(\alpha(k)) - H*w(k)*\sin(\alpha(k));$

ypunto(k) = $v(k)*\sin(\alpha(k)) + H*w(k)*\cos(\alpha(k));$

alphapunto = $w(k);$

◇ Integración numérica para obtener las posiciones a partir de las velocidades de interés.

$x(k+1) = x(k) + ts*xpunto(k);$

$y(k+1) = y(k) + ts*ypunto(k);$

$\alpha(k+1) = \alpha(k) + ts*alphapunto;$

$x(k+1) = xr(k+1) - a*\cos(\alpha(k+1));$

$y(k+1) = yr(k+1) - a*\sin(\alpha(k+1));$

Algoritmo 7: Simulación para el seguimiento y coordinación de trayectorias. Tercera parte.

- ◇ Se declaran las variables que se utilizan y las entradas que van a ser las posiciones de interés del robot seguidor.
- ◇ Condición indicando que el robot líder termino el trayecto y el siguiente robot puede imitarlo.

if (*error(length(t)) < 0.01*) **then**

- ◇ Se establece las posiciones de interés de acuerdo con el trayecto final del robot líder

Xref2 = x ;

Yref2 = y ;

- ◇ Se deriva las posiciones de interés obteniendo las velocidades de interés del robot seguidor.

xrefd2= derivada de Xref;

yrefd2= derivada de Yref;

- ◇ Establesco las posiciones y orientación inicial del robot seguidor desplazadas con respecto al líder en el marco global.

x2(1) = -1;

y2(1) = 6;

alpha2(1) = 0;

for *k=1:length(t)* **do**

- ◇ Mismo procedimiento, aplicar el error, ley de control y determinar las velocidades en base de las nuevas posiciones de referencia y posiciones iniciales del robot seguidor.

end

end

4.5.2. Simulaciones para posicionar el robot diferencial a un punto de interés con coordenadas rectangulares.

En la implementación del algoritmo de simulación en Matlab se hace uso de la ecuación (4.5). Se indica una posición inicial desde la que empieza a moverse el robot diferencial, estas serán $X(1) = 0, Y(1) = 0$. También se indica el valor de referencia en el eje X, Y del marco global donde se espera que el robot móvil se posicione, serán $X_{ref} = 6, Y_{ref} = -4$.

Inicialmente se realiza la simulación con una constante de ganancia $k = 0,1$ y tiempo de simulación $t = 0 : 0,1 : 10$.

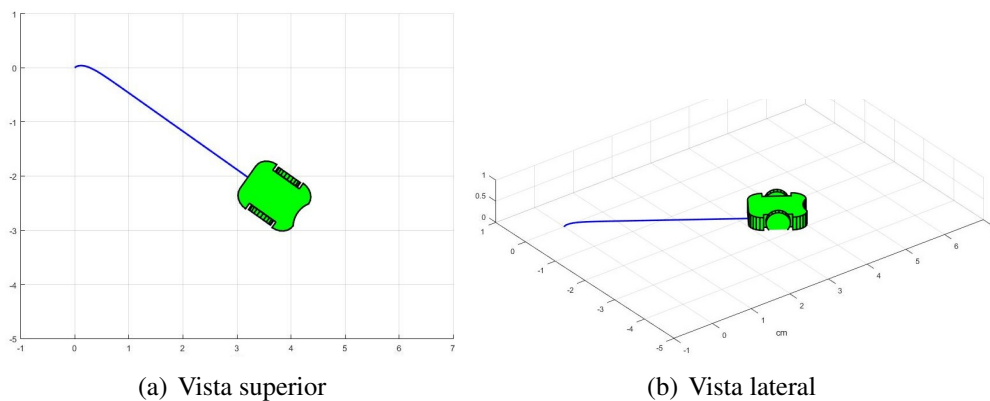


Figura 4.10: Simulación del robot diferencial para posicionarse en los puntos $X_{ref} = 6, Y_{ref} = -4$ / Autor.

Como se observa en la Figura 4.10 el robot móvil no logra llegar a los puntos deseados, esto es debido por los parámetros de ganancia y el tiempo de simulación dado en el sistema.

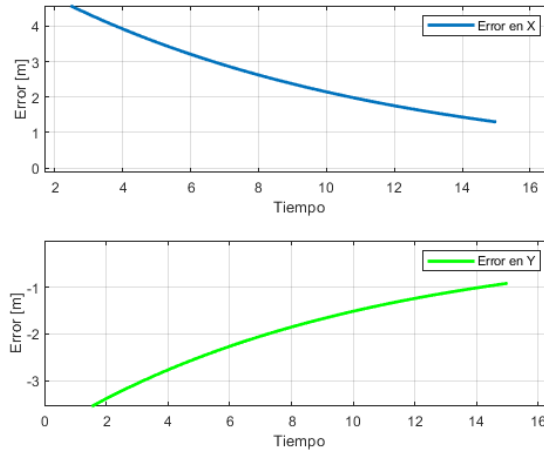


Figura 4.11: Error de posición con posiciones de interés $X_{ref} = 6$, $Y_{ref} = -4$ / Autor

En la Figura 4.11 se observa que el error en la posición X,Y no converge a 0 y es por esto que el robot móvil no se posiciona al valor deseado, para la siguiente simulación se realiza con la misma posición de interés pero con un tiempo de simulación $t = 0 : 0,1 : 30$ y una ganancia $k = 0,1$.

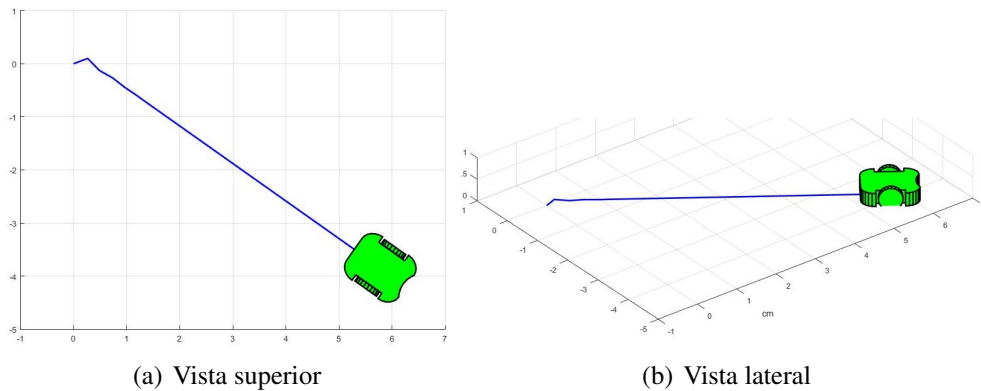


Figura 4.12: Simulación del robot diferencial para posicionarse en los puntos $X_{ref} = 6$, $Y_{ref} = -4$ y tiempo de simulación $t = 0 : 0,1 : 30$ / Autor.

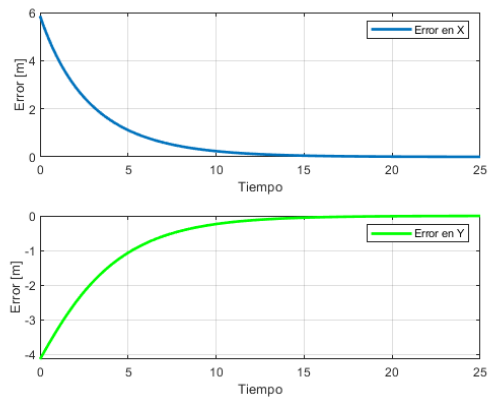


Figura 4.13: Error de posición con posiciones de interés $X_{ref} = 6$, $Y_{ref} = -4$ y tiempo de simulación $t = 0 : 0,1 : 30$ / Autor

Al proporcionarle mayor tiempo de simulación, el error logra convergir a cero como se observa en la Figura 4.13 posicionando el robot al punto deseado presentado en la Figura 4.12. Esto es por el pequeño valor en la constante de ganancia k que hace mover al robot móvil más lento, por tal motivo se necesita más tiempo en la simulación para que el robot diferencial logre su ruta.

4.5.3. Simulación de coordinación de robots líder - seguidor.

Se realiza las simulaciones probando diferentes trayectorias en los robots diferenciales. Inicialmente se hace la prueba con la trayectoria de un círculo donde en el algoritmo de Matlab al robot líder se implementa como referencia la ecuación paramétrica del círculo (4.7) y en el robot seguidor se implementa los valores finales de la trayectoria realizada por el líder, ambos con una constante de ganancia de $k = 0,4$.

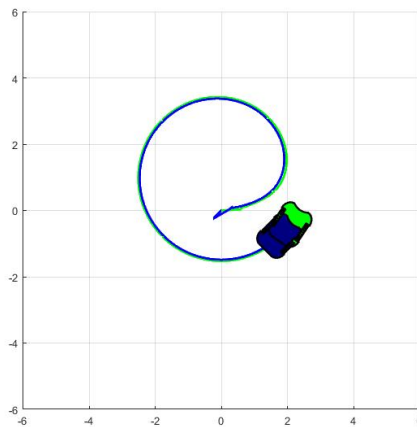
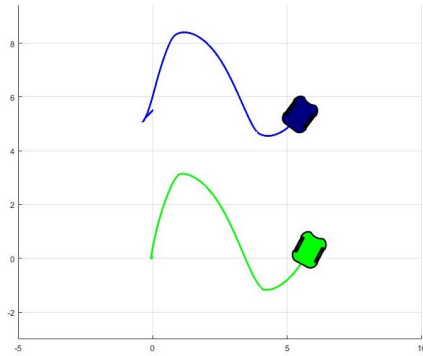
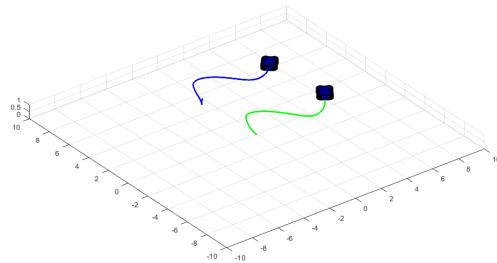


Figura 4.14: Trayecto de círculo para líder-seguidor / Autor

También se realiza la prueba para que el robot líder realice un trayecto de una señal senoidal por el eje X_I y el seguidor realice el mismo trayecto a una distancia de 2 cm del eje Y_I del líder. El robot líder esta representado con la línea verde y el seguidor con la línea azul.



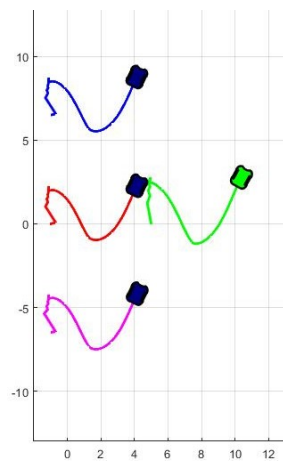
(a) Vista superior



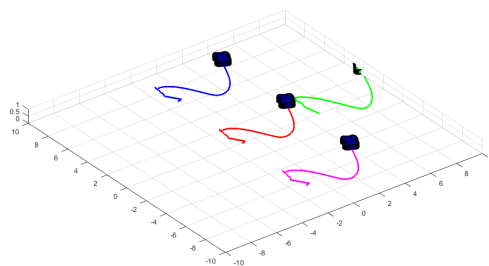
(b) Vista lateral

Figura 4.15: Trayecto senoidal para líder-seguidor / Autor.

La ultima simulación se realiza para observar el trayecto de una señal cosenoidal en el eje X_I por cuatro robots diferenciales, donde el líder está representado por el color verde y los seguidores por los colores, azul, rojo y magenta.



(a) Vista superior



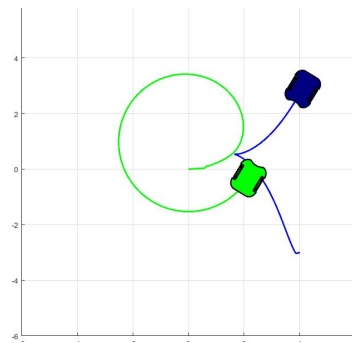
(b) Vista lateral

Figura 4.16: Trayecto cosenoidal para líder-seguidor / Autor.

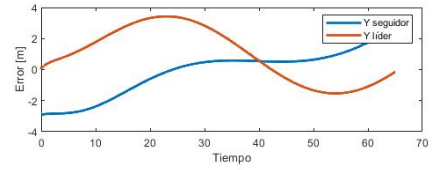
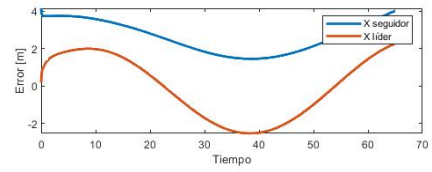
4.5.4. Ajuste de ganancia k en seguimiento de ruta realizada por robot líder

Se realiza simulaciones para establecer una ganancia k adecuada para el seguimiento de trayectorias, se efectúa varias simulaciones con diferentes ganancias k , el trayecto realizado en la Figura 4.17.

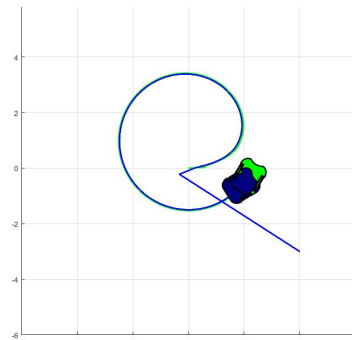
Basado en el trayecto descrito en la Figura 4.17 , se propone una ganancia $k = [7.5 - 11]$ para el seguimiento de trayectoria realizada por robot líder.



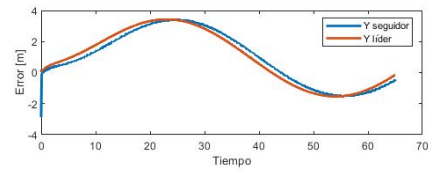
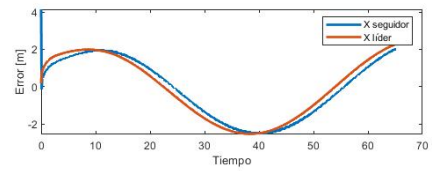
(a) $k = 0.5$



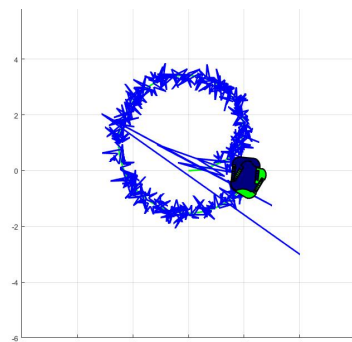
(b) $k = 0.5$



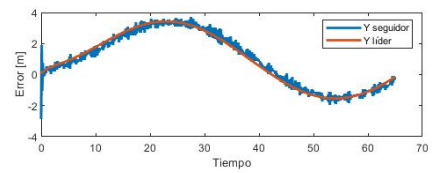
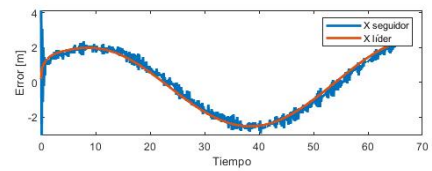
(c) $k = 10$



(d) $k = 10$



(e) $k = 17$



(f) $k = 17$

Figura 4.17: Seguimiento del robot líder con diferentes valores de ganancia k / Autor.

Capítulo 5

CONCLUSIONES

El objetivo general de este trabajo fue diseñar un controlador no lineal basado en el error de posición para coordinar el movimiento de dos o más robots diferenciales utilizando la formación líder – seguidor. La validación del rendimiento del controlador se hizo numericamente y permitió observar que las unidades robóticas seguían coordinadamente diferentes trayectorias de referencia, construidas a partir de ecuaciones paramétricas que determinan una ruta específica, en el marco de referencia de coordenadas globales. Las unidades robóticas son capaces de seguir al robot líder incluso en el escenario en el cual no se tiene disponibilidad de la medición de la velocidad angular, lo cual muestra que el observador de estados propuesto es efectivo. Para poder realizar el seguimiento y coordinación entre los robots mencionados anteriormente, fue necesario cumplir con las siguientes actividades.

- Se describió y comprendió el modelo matemático del robot móvil diferencial.
- Se diseñó un observador de estado para estimar la velocidad angular de cada motor del robot móvil.

- Se diseñó un control no lineal basado en el error de posición del robot móvil.
- Se realizaron pruebas para validar la coordinación del sistema robótico con una trayectoria simulada en Matlab.

5.1. Consideraciones practicas

- Con el fin de lograr mayor estabilidad de las unidades roboticas en tareas cooperativas, en la practica el conttrollador propuesto se sugiere implementarlo en una estructura robotica tipo triciclo con tracción trasera. Este tipo de robots brinda robusteza con respecto a las perturbaciones u obstaculos encontrados en el entorno.
- Con base en los resultados numéricos se pudo apreciar que para valores de ganancia del controlador $k > 0,5$, el seguimiento de la trayectoria presenta pequeñas oscilaciones en la orientación del robot como se muestra en la Figura 5.1, lo cual se propagara a los demás robots en la red.

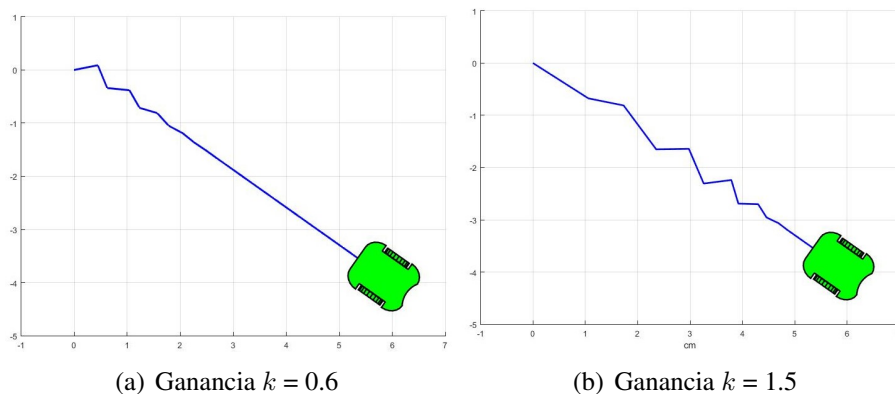
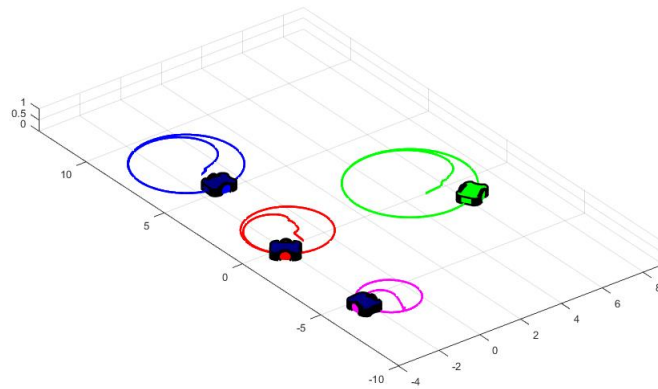
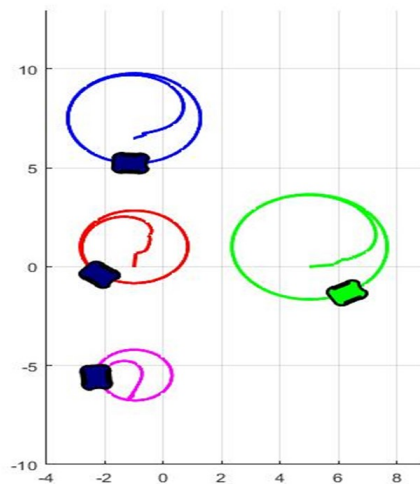


Figura 5.1: Trayecto del robot diferencial al punto (6,-4).

- El controlador propuesto realiza la coordinación de las unidades roboticas moviles simple y cuando todos los robots tengan acceso a la posición del lider. No obstante cuando esto no ocurre, es decir la interacción entre las unidades roboticas estan en configuración en "línea", cada robot solo conoce la posición de su vecino, existe error en la coordinación, como se muestra en la Figura 5.2.



(a) Vista lateral



(b) Vista superior

Figura 5.2: Trayecto círculo para cuatro robots moviles / Autor.

5.2. Trabajos futuros.

La robótica móvil tiene múltiples aplicaciones en el área industrial, es por eso que esta sigue siendo un área activa de investigación. Una forma de extender este trabajo es flexibilizar las topologías de interconexión y enlazar la información de los robots a un servidor para operar el sistema remotamente. Particularmente, se podrían desarrollar los siguientes aspectos.

- Extender el controlador para que considere diferentes topologías de interconexión entre cada unidad robótica.
- Realizar un análisis de estabilidad que permita determinar para qué valores de ganancia del controlador, el sistema realiza la coordinación de movimiento.
- Realizar la implementación real en la coordinación de robots móviles con estructura líder-seguidor, utilizando múltiples estrategias como Bluetooth o Wi-Fi para garantizar una comunicación de datos entre ellos.
- Implementar el controlador en dispositivos que tengan conectividad a un servidor a través de wifi, con el fin de almacenar y manipular los datos de las conexiones.

REFERENCIAS

- [1] G. L. M. Aranda, C. Sagiúes and Y. Mezouar, “Formation control of mobile robots using multiple aerial cameras,” *IEEE Transactions on Robotics*, vol. 31, no. 4, pp. 1064–1071, 2015.
- [2] A. Oviedo, “Robot de seguridad controlado por wifi cerberus 1.0.” Universidad tecnologica de pereira. Trabajo de grado, Colombia, 2016.
- [3] I. K. V. Šulák and P. Čičák, “Search using a swarm of unmanned aerial vehicles,” *2017 15th International Conference on Emerging eLearning Technologies and Applications (ICE-TA)*, pp. 1–6, 2017.
- [4] M. J. G. M. Bakhshipoura and F. Namdaria, “Swarm robotics search and rescue: A novel artificial intelligence inspired optimization approach,” *Applied Soft Computing* 57, pp. 708–726, 2017.
- [5] B. F. B. P. T. Blender, T. Buchner and C. Schlegel, “Managing a mobile agricultural robot swarm for a seeding task,” *IECON 2016 - 42nd Annual Conference of the IEEE Industrial Electronics Society*, pp. 6879–6886, 2016.
- [6] P. L. F. W. y. T. S. J. Varughese, R. Thenius, “A model for bio-inspired underwater swarm robotic exploration,” pp. 385–390, 2018.

- [7] J. M. F. Fossum and P. C. Haddow, “Repellent pheromones for effective swarm robot search in unknown environments,” *2014 IEEE Symposium on Swarm Intelligence*, pp. 1–8, 2014.
- [8] H. y. S. N. G. F. Fitriana, “Formation control of leader-follower robot using interval type-2 fuzzy logic controller,” *2017 International Conference on Electrical Engineering and Computer Science (ICECOS)*, pp. 44–49, 2017.
- [9] A. G. y M. R. Esfahanian, “Using swarm robots based on leader-followers method for spherical object manipulation,” *2013 First RSI/ISM International Conference on Robotics and Mechatronics (ICRoM), Tehran,.*, pp. 413–418, 2013.
- [10] P. P. Ray, “Internet of robotic things: Concept, technologies, and challenges,” *IEEE Access*, vol. 4, pp. 9489–9500, 2016.
- [11] F. T. K. A. Defoort, M. and Perruquetti, “Sliding-mode formation control for cooperative autonomous mobile robots,” *IEEE Transactions on Industrial Electronics*, vol. 55, pp. 3944–3953, 2008.
- [12] X. J. Li, X. and Z. Cai, “Backstepping based multiple mobile robots formation control.,” *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 887–892, 2005.
- [13] O. G. Mariottini, G.L. and D. Prattichizzo, “Image-based visual servoing for nonholonomic mobile robots using epipolar geometry,” *IEEE Transactions on Robotics*, pp. 425–437, 2005.
- [14] P. C. M. van de Wouw, Naghshtabrizi and J. Hespanha, “Tracking control for sampled-data

systems with uncertain sampling intervals and delays,” *International Journal of Robust and Nonlinear Control*, pp. 387–411, 2010.

- [15] W. Heemels and N. van de Wouw, “Stability and stabilization of networked control systems,” *Networked Control Systems*, pp. 203–253.
- [16] T. A. R. Carnevale, D. and D. Nesic, “A lyapunov proof of an improved maximum allowable transfer interval for networked control systems,” *IEEE Transactions on Automatic Control*, pp. 892–897, 2007.
- [17] R. Dhaouadi and A. A. Hatab, “Dynamic modelling of differential-drive mobile robots using lagrange and newton-euler methodologies: A unified framework,” *Advances in Robotics and Automation*, vol. 2, pp. 1–7, 2013.
- [18] M. A. M. V. Leonardo Enrique Solaque Guzmán and E. L. R. Vásquez, “Seguimiento de trayectorias con un robot móvil de configuración diferencial,” *Ing. USBmed*, vol. 5, no. 1, pp. 26–34, 2014.
- [19] M. RÍOS G., LUÍS HERNANDO; BUENO L., “Seguimiento de trayectorias con un robot móvil de configuración diferencial,” *Scientia Et Technica*, vol. XIV, no. 38, pp. 13–18, 2008.
- [20] F. Monasterio-Huelin and A. Gutiérrez, “Modelado de un motor dc,” 2020.