



Universidad Autónoma de San Luis Potosí



Facultad de Ingeniería

Centro de Investigación y Estudios de Posgrado

Síntesis de Controladores para la Formación de un Sistema Multiagente

TESIS

Que para obtener el título de:
Maestro en Ingeniería Eléctrica

PRESENTA:

Ing. Julián Alejandro Hernández Gallardo

Directores de Tesis:

Dr. Emilio Jorge González Galván
Dr. César Fernando Francisco Méndez Barrios



San Luis Potosí, San Luis Potosí

Agosto de 2021



UASLP
Universidad Autónoma
de San Luis Potosí



FACULTAD DE
INGENIERÍA

17 de junio de 2021

**ING. JULIÁN ALEJANDRO HERNÁNDEZ GALLARDO
P R E S E N T E.**

En atención a su solicitud de Temario, presentada por los **Dres. Emilio Jorge González Galván y César Fernando Francisco Méndez Barrios**, Asesor y Coasesor de la Tesis que desarrollará Usted con el objeto de obtener el Grado de **Maestro en Ingeniería Eléctrica**, me es grato comunicarle que en la sesión del H. Consejo Técnico Consultivo celebrada el día 17 de junio del presente, fue aprobado el Temario propuesto:

TEMARIO:

“Síntesis de Controladores para la Formación de un Sistema Multiagente”

1. Introducción.
2. Fundamentos Matemáticos: Sistemas Dinámicos y Sistemas Multiagente.
3. Modelado y Control de un Robot Móvil.
4. Modelado y Control para la Formación de un Sistema Multiagente.
5. Validación Experimental.
6. Conclusiones y Trabajo Futuro.
Bibliografía.

“MODOS ET CUNCTARUM RERUM MENSURAS AUDEBO”

A T E N T A M E N T E



DR. RICARDO ROMERO MENDOZA
SECRETARIO DE CONSEJO
UNIVERSIDAD AUTONOMA
DE SAN LUIS POTOSI
FACULTAD DE INGENIERIA
SECRETARIA

www.uaslp.mx

Copia. Archivo.
*etn.

Av. Manuel Nava 8
Zona Universitaria • CP 78290
San Luis Potosí, S.L.P.
tel. (444) 826 2330 al39
fax (444) 826 2336



Dedicatoria

A mis padres Martha y Julián, sin su apoyo no sería la persona que soy ahora.



Agradecimientos

A **Dios**, por la vida de mis padres y hermanos, por bendecirme cada día de mi vida, por permitirme cumplir esta nueva meta en mi vida.

A mi familia, mis padres **Julián** y **Martha**, por el apoyo incondicional, por ser los principales motores de mis sueños, gracias por confiar en mí cada día, por la educación y valores inculcados para hacerme un hombre de bien. A mis hermanos, **Cinthya**, **Metzli** y **Bryan**, por siempre tener una palabra de apoyo en mis momentos de desesperación y cada risa que me sacaban cuando me sentía triste.

A mis asesores, **Dr Emilio** y **Dr César**, por la confianza para el desarrollo de este trabajo de tesis, por el conocimiento compartido durante las asignaturas del posgrado y por su paciencia.

Al **Consejo Nacional de Ciencia y Tecnología**, por el apoyo económico brindado para mi manutención durante la duración de todo mi posgrado.

A la **Universidad Autónoma de San Luis Potosí**, por brindarme una educación de calidad, por el préstamo de equipo e instalaciones.

A mi novia **Daniela**, por el cariño, los consejos y apoyo durante todo este tiempo, por siempre tener palabras de aliento para mí. La segunda tesis donde soportas mi estrés y actitud, gracias por estar siempre.

Julián Alejandro Hernández Gallardo



Resumen

En el presente documento se propone el diseño e implementación de un control centralizado para el control de formación y seguimiento de trayectorias para un sistema multiagente formado por robots móviles terrestres de tracción diferencial. Se presenta el análisis de los modelos cinemáticos, de tipo no holonómico y holonómico respectivamente. Se presenta una metodología funcional para el diseño y construcción de los robots móviles, en segunda instancia, se diseñaron diferentes controladores de bajo orden de tipo PI para el control de velocidad del robot individual para posteriormente diseñar e implementar controladores para posición y seguimiento de trayectorias de manera individual. Por último se presenta el análisis de estabilidad de la formación utilizando la teoría de Lyapunov. Todo lo anteriormente mencionado se validó de manera computacional y de manera experimental.

Palabras Clave:

Robots móviles, Sistemas Multiagente, Controladores PI, Formación de Robots.



Abstract

This report proposes the design and implementation of a centralized control for training control and trajectory tracking for a multi-agent system consisting of differential traction ground mobile robots. The analysis of kinematic models, of non-holonomic and holonomic type respectively, is presented. A functional methodology for the design and construction of the mobile robots is presented, in second instance, different low order controllers of PI type were designed for the velocity control of the individual robot, to later design and implement controllers for position and trajectory tracking individually. Finally, the stability analysis of the formation is presented using the Lyapunov theory. All the aforementioned was validated computationally and experimentally.

Keywords:

Mobile robotics, Multi-agent systems, PI Controllers, Robotics Formation.



Acrónimos

AE	Asintóticamente Estable.
CD	Corriente Directa.
CLA	Control en Lazo Abierto.
CLC	Control en Lazo Cerrado.
CPC	Cinemática del Punto Central del Robot Móvil.
CPD	Cinemática del Punto Desplazado del Robot Móvil.
CPO	Control de Posición sin Orientación.
CPCO	Control de Posición con Orientación.
CPF2R	Control de Posición de la Formación de Dos Robots.
CPOGA	Control de Posición sin Orientación con Ganancia Adaptativa.
CSC	Control de Seguimiento de Caminos.
CST	Control de Seguimiento de Trayectoria.
CSTF2R	Control de Seguimiento de Trayectoria de la Formación de Dos Robots.
CSTF2RR	Control de Seguimiento de Trayectoria de la Formación de Dos Robots con Retardo.
CSTF3R	Control de Seguimiento de Trayectoria de la Formación de Tres Robots.
CSTR	Control de Seguimiento de Trayectoria con Retardo.
FPS	Frames por Segundo.
E	Estable.
MAS	Multi Agent Systems (Sistemas Multiagente).
PCB	Printed Circuit Board (Placa de Circuito Impreso).
PI	Proporcional Integral.
RMR	Robot Móvil con Ruedas.
SAD	Sistema de Adquisición de Datos.
SVA	Sistema de Visión Artificial.
TEL	Teorema de Estabilidad de Lyapunov.
UAV	Unmanned Aerial Vehicle (Vehículo aéreo no tripulado).

Símbolos

\mathcal{C}^2	Continuamente diferenciable.
\mathcal{G}	Sistema de coordenadas global.
\mathbf{x}	Estados del sistema.
\mathcal{V}	Función candidata de Lyapunov.
\mathbf{p}	Vector de localización.
$\dot{\mathbf{p}}$	Vector de velocidades.
$\dot{\mathbf{q}}$	Vector de entradas.
\mathbb{R}	Número reales.
θ	Ángulo de rotación de la rueda del robot.
ω	Velocidad angular.
v	Velocidad lineal.
l	Longitud entre las ruedas del robot.
r	Radio de las ruedas del robot.
φ	Orientación del robot respecto al eje x .
t	Tiempo.
a	Distancia del punto descentralizado del robot.
\mathbf{J}	Matriz Jacobiana.
\mathbf{J}^{-1}	Inversa matriz Jacobiana.
\mathbf{J}^\dagger	Pseudoinversa matriz Jacobiana.
\mathbf{K}	Matriz de ganancias.
e	Error de posición.
\mathbf{e}	Vector de errores.
δ	Error de velocidad.
d	Distancia euclideana entre el punto inicial y punto final.
D	Distancia euclideana entre dos robots.
Φ_C	Orientación de la formación.
K_p	Constante proporcional de controlador PID.
K_i	Constante integral de controlador PID.
K_d	Constante derivativa de controlador PID.
λ	Factor de velocidad en la respuesta del controlador PID λ .
ssi	Si y solo si.
diag	Matriz diagonal.



Índice General

Resumen	v
Abstract	vii
Índice General	xiii
Índice de Figuras	xvii
Índice de Tablas	xxi
Capítulo 1: Introducción	1
1.1 Motivación	1
1.2 Antecedentes	3
1.3 Objetivos	5
1.4 Sinopsis	6
Capítulo 2: Fundamentos Matemáticos: Sistemas Dinámicos y	
Sistemas Multiagente	9
2.1 Conceptos Matemáticos	9
2.1.1 Sistema Cartesiano de Referencia	9
2.1.2 Coordenadas Cartesianas	10
2.2 Robótica	10
2.2.1 Robótica Móvil	12
2.3 Cinemática	15
2.3.1 Cinemática de Robots	16
2.3.2 Cinemática Directa e Inversa de un Robot Móvil	16
2.4 Estabilidad	19
2.5 Sistemas Multiagente	20
Capítulo 3: Modelado y Control de un Robot Móvil	23
3.1 Plataforma Experimental: Diseño y Construcción del Robot	23
3.1.1 Actuadores	23

3.1.2	Sensor de Velocidad	24
3.1.3	Driver de Potencia	25
3.1.4	Comunicación Inalámbrica	26
3.1.5	Sistema de Control	27
3.2	Identificación de la Dinámica del Motor	30
3.3	Control de Velocidad del Motor	34
3.3.1	Controlador PI	34
3.4	Modelo Cinemático del Robot Móvil	42
3.4.1	Restricciones Cinemáticas	42
3.4.2	Cinemática del Punto Central del Robot	45
3.4.3	Cinemática de un Punto Desplazado del Robot	48
3.5	Control en Lazo Abierto	50
3.6	Control en Lazo Cerrado	51
3.6.1	Control de Posición sin Orientación	52
3.6.2	Control de Posición sin Orientación con Ganancia Adap- tativa	56
3.6.3	Control de Seguimiento de Trayectoria	60
3.6.4	Control de Seguimiento de Caminos	64
3.6.5	Control de Posición con Orientación Final	70
3.6.6	Control de Seguimiento de Trayectoria con Retardo	76

Capítulo 4: Modelado y Control para la Formación de un Sistema Multiagente 83

4.1	Formación de Dos Robots	83
4.1.1	Modelo Cinemático	83
4.1.2	Control de Posición	86
4.1.3	Control de Seguimiento de Trayectoria	90
4.1.4	Control de Seguimiento de Trayectoria con Retardo	93
4.2	Formación de Tres Robots	95
4.2.1	Modelo Cinemático	95
4.2.2	Control de Posición	99
4.2.3	Control de Seguimiento de Trayectoria	100

Capítulo 5: Validación Experimental 103

5.1	Sistema de Adquisición de Datos	103
5.2	Pruebas Experimentales de un Robot	105
5.2.1	Cinemática de un Punto Central del Robot	106

5.2.2	Cinemática de un Punto Descentralizado del Robot . . .	107
5.2.3	Control de Posición sin Orientación	108
5.2.4	Control de Posición sin Orientación con Ganancia Adap- tativa	110
5.2.5	Control de Seguimiento de Trayectoria	112
5.3	Pruebas Experimentales de la Formación de Dos Robots . . .	114
5.3.1	Trayectoria de un Círculo	114
5.3.2	Trayectoria de una Función Seno	115
	Conclusiones y Trabajo Futuro	117
	Bibliografía	119



Índice de Figuras

1.1	Fenómenos Naturales estudiados como Sistemas Multiagente.	2
1.2	Interacción de las Neuronas del Cerebro Humano.	2
2.1	Representación de un Vector en Coordenadas Cartesianas.	9
2.2	Robot y su Interacción con el Entorno.	11
2.3	Clasificación de los Robots Móviles.	12
2.4	Configuración de Robots Móviles Terrestres sobre Ruedas.	14
2.5	Esquema de una Trayectoria.	15
2.6	Diagrama de Bloques del Modelo Cinemático.	17
2.7	Diagrama de Bloques del Modelo Cinemático Diferencial.	19
2.8	Esquema de un Sistema Multiagente.	20
3.1	Motor CHIHAI CHR-GM25-370.	24
3.2	Datasheet del Motor CHR-GM25-370.	24
3.3	Encoder de Cuadratura del Motor CHR-GM25-370.	25
3.4	Módulo TB6612FNG.	26
3.5	BT HC05.	26
3.6	Arduino MEGA.	27
3.7	PCB Compatible con Arduino MEGA.	28
3.8	Robot en Dibujo Asistido por Computadora.	29
3.9	Robot Construido.	29
3.10	Zona Muerta del Motor.	31
3.11	Diagrama de Comunicaciones para Experimentos de Identificación.	32
3.12	Respuesta en Lazo Abierto de los Motores del Robot No. 1.	32
3.13	Respuesta en Lazo Abierto de los Motores del Robot No. 2.	33
3.14	Diagrama de Bloques Controlador PI Estándar.	35
3.15	Región de Estabilidad en el Plano Paramétrico K_p vs K_i del Robot No. 1.	35
3.16	Región de Estabilidad en el Plano Paramétrico K_p vs K_i del Robot No. 2.	36
3.17	Respuestas Numéricas del Robot No. 1 a Controladores PI.	40

3.18	Respuestas Numéricas del Robot No. 2 a Controladores PI.	40
3.19	Respuestas Experimentales del Robot No. 1 a Controladores PI. . .	41
3.20	Respuestas Experimentales del Robot No. 2 a Controladores PI. . .	41
3.21	Esquema del Robot Móvil en Vista del Plano $\{X, Z\}_G$	44
3.22	Restricciones Cinemáticas del Robot Diferencial.	45
3.23	Esquema del Robot Móvil Diferencial para el Modelo No Holonómico. .	46
3.24	Respuesta Numérica de la CPC.	48
3.25	Esquema del Robot Móvil Diferencial para el Modelo Holonómico. .	49
3.26	Respuesta Numérica de la CPD.	50
3.27	Esquema para la Posición del Robot Diferencial.	53
3.28	Respuesta Numérica No. 1 del CPO.	54
3.29	Respuesta Numérica No. 2 del CPO.	55
3.30	Respuesta Numérica No. 3 del CPO.	56
3.31	Respuesta Numérica No. 1 del CPOGA.	58
3.32	Respuesta Numérica No. 2 del CPOGA.	59
3.33	Respuesta Numérica No. 3 del CPOGA.	60
3.34	Respuesta Numérica No. 1 del CST.	62
3.35	Respuesta Numérica No. 2 del CST.	63
3.36	Respuesta Numérica No. 3 del CST.	64
3.37	Esquema para el CSC.	66
3.38	Respuesta Numérica No. 1 del CSC.	68
3.39	Respuesta Numérica No. 2 del CSC.	69
3.40	Respuesta Numérica No. 3 del CSC.	70
3.41	Esquema de la Postura del Robot en Coordenadas Polares.	71
3.42	Respuesta Numérica No. 1 del CPCO.	74
3.43	Respuesta Numérica No. 2 del CPCO.	75
3.44	Respuesta Numérica No. 3 del CPCO.	76
3.45	Esquema del CSTR.	77
3.46	Respuesta Numérica No. 1 del CSTR.	79
3.47	Respuesta Numérica No. 2 del CSTR.	80
3.48	Respuesta Numérica No. 3 del CSTR.	81
4.1	Esquema de la Formación de Dos Robots.	83
4.2	Diagrama de Bloques para el CPF2R.	86
4.3	Respuestas Numéricas No. 1 del CPF2R.	88
4.4	Respuestas Numéricas No. 2 del CPF2R.	89
4.5	Diagrama de Bloques para el CSTF2R.	90

4.6	Respuestas Numéricas del CSTF2R al Seguimiento de un Círculo. . .	91
4.7	Respuestas Numéricas del CSTF2R al Seguimiento de una Función Seno.	92
4.8	Esquema del CSTRR.	93
4.9	Respuestas Numéricas del CSTF2RR al Seguimiento de un Círculo. . .	94
4.10	Esquema de la Formación de Tres Robots.	95
4.11	Respuestas Numéricas No. 1 del CPF3R.	99
4.12	Respuestas Numéricas No. 2 del CPF3R.	100
4.13	Respuestas Numéricas del CSTF3R al Seguimiento de un Círculo. . .	101
4.14	Respuestas Numéricas del CSTF3R al Seguimiento de una Función Seno.	102
5.1	Diagrama de Comunicaciones.	103
5.2	Área de Trabajo en los Experimentos.	104
5.3	Programas Utilizados en los Experimentos.	105
5.4	Respuesta Experimental del Robot No. 1 a la CPC.	106
5.5	Respuesta Experimental del Robot No. 2 a la CPC.	106
5.6	Respuesta Experimental del Robot No. 1 a la CPD.	107
5.7	Respuesta Experimental del Robot No. 2 a la CPD.	107
5.8	Respuesta Experimental del Robot No. 1 al CPO.	108
5.9	Respuesta Experimental del Robot No. 2 al CPO.	109
5.10	Respuesta Experimental del Robot No. 1 al CPOGA.	110
5.11	Respuesta Experimental del Robot No. 2 al CPOGA.	111
5.12	Respuesta Experimental del Robot No. 1 al CST.	112
5.13	Respuesta Experimental del Robot No. 2 al CST	113
5.14	Respuestas Experimentales al CSTF2R al Seguimiento de un Círculo. . .	114
5.15	Respuestas Experimentales al CSTF2R al Seguimiento de una Fun- ción Seno.	115



Índice de Tablas

2.1	Componentes de RMR.	14
3.1	Constantes del Modelo del Robot No. 1.	33
3.2	Constantes del Modelo del Robot No. 2.	34
3.3	Constantes del Control $PI\sigma$ Robot No. 1.	36
3.4	Constantes del Control $PI\sigma$ Robot No. 2.	36
3.5	Constantes del Control $PI\sigma$ Arbitrario Robot No. 1.	37
3.6	Constantes del Control $PI\sigma$ Arbitrario Robot No. 2.	37
3.7	Constantes del Control $PI\lambda$ Robot No. 1.	38
3.8	Constantes del Control $PI\lambda$ Robot No. 2.	38
3.9	Constantes del Control PI Halman Robot No. 1.	38
3.10	Constantes del Control PI Halman Robot No. 2.	39
3.11	Constantes de los Controladores PI	39
3.12	Parámetros de Rendimiento de los Controladores PI para G_{R1}	42

Introducción

En este capítulo se presenta la motivación para el desarrollo de este trabajo, así como los trabajos reportados en la literatura mostrando sus enfoques y plataformas experimentales utilizadas. El capítulo termina presentando la organización de la parte restante de la tesis.

1.1. Motivación

Durante las últimas tres décadas, los avances significativos en las tecnologías de comunicación, electrónica, robótica y control han hecho posible el desarrollo de sistemas multiagente que realizan tareas que agentes individuales por sí solos no pueden realizar. Las tareas que antes se realizaban con dispositivos grandes, costosos y complejos ahora se pueden realizar con un conjunto de sistemas más pequeños, más baratos y más confiables.

La idea de que un conjunto de agentes individuales trabajen y cooperen entre sí está inspirado en varios ejemplos de la biología y la medicina. La Figura 1.1 muestra un ejemplo biológico de un sistema de agentes múltiples en la naturaleza: colonias de hormigas, parvadas de aves, cardumen de peces, colonias de luciérnagas, panales de abejas. Los sistemas anteriores son sólo algunos ejemplos de fenómenos en la naturaleza que pueden ser estudiados como sistemas multiagente.

Un cardumen de peces se comporta como un sola masa para poder intimidar depredadores, además que el cardumen tiene la capacidad de evitar obstáculos y de reconfigurarse conforme se mueven [1]. Muchas aves vuelan juntas en formación de V, lo que reduce el gasto de energía y mejora la locomoción de los individuos de la parvada [2]. Las luciérnagas se juntan en miles de árboles y los machos destellan la parte trasera de su cuerpo de manera rítmica y sincrónica con el único propósito de la adaptación sexual, mientras que si la hembra le responde con la misma frecuencia, es sinónimo de aceptación [3]. Por último,

las hormigas presentan fenómenos como son la agrupación para formar puentes colgantes consigo mismas, la descarga de feromonas para marcar caminos que lleven a comida, entre algunas otras [4].



(a) Colonias de luciérnagas.



(b) Colonias de hormigas.



(c) Parvada de aves.



(d) Cardumen de peces.

Figura 1.1: Fenómenos Naturales estudiados como Sistemas Multiagente.

En el área de la medicina el ejemplo más claro de un sistema multiagente se ve en el funcionamiento de nuestro cerebro, donde cada una de las neuronas (véase la Figura 1.2), interacciona entre sí y con el medio, esto con el fin de tomar decisiones.



Figura 1.2: Interacción de las Neuronas del Cerebro Humano.

Tales hechos (biológicos y médicos) ilustran el concepto de *sincronización*, el cual por definición de la Real Academia Española es «hacer coincidir en el tiempo dos o más movimientos o fenómenos» [5].

A principios del año 2000, el control *consensus* en enjambres de agentes móviles aumentó el interés de la comunidad científica debido a sus varias aplicaciones ya sea en hardware o software. El problema del *consensus* consiste en establecer las condiciones bajo las cuales las diferencias entre dos movimientos cualesquiera de un grupo de sistemas dinámicos, convergen a cero asintóticamente [6, 7, 8]. Existen aplicaciones como lo son osciladores acoplados [9, 10], control de seguimiento de trayectorias en formación de robots móviles [11], la sincronización de convertidores CD-CD [12], microrredes eléctricas [13] entre muchas más aplicaciones.

Las técnicas de control para el problema de control de seguimiento de trayectorias de formación de robots móviles han sido estudiados durante los últimos años. Uno de los inconvenientes de estas técnicas es que el controlador se basa en la suposición de que la velocidad angular del robot líder debe ser diferente de cero, no se toman en cuenta las formaciones en línea recta. Esto motiva la investigación de nuevos enfoques para diseñar nuevas técnicas de control no lineal que puedan ayudar a los robots a seguir trayectorias manteniendo una formación en línea recta y un modelo que permita la interacción entre los agentes individuales.

1.2. Antecedentes

En la teoría de sincronización existen varios problemas de control que se pueden abordar como lo son: la formación de vehículos aéreos no tripulados [14], enjambres robóticos [15], la sincronización de osciladores caóticos [9], entre otros. Desde el punto de vista de control, se han analizado las condiciones que permiten llevar determinadas variables de un grupo de sistemas (bajo una interacción adecuada) a cierto valor común mediante leyes de control locales, definido formalmente como *consenso* [8].

Los comportamientos cooperativos más comunes en el estado del arte son *consensus* [8], *rendezvous* [16] y *flocking* [17]. Se denomina *flocking* a la formación

de vehículos que imitan a las parvadas de aves. En [18] se menciona que cada agente de un flocking debe cumplir con tres principios básicos: evitar colisiones, intentar igualar la velocidad de sus compañeros cercanos y mantener la distancia entre ellos.

Por otra parte, se llama control de formación al problema de controlar la posición relativa y la orientación de los robots móviles que conforman un MAS para realizar una tarea dada [19]. En la literatura, se han propuesto muchos enfoques para resolver el problema de control de formación de un MAS, entre los más destacados están: líder-seguidor [20, 21], líder virtual [22], estructuras virtuales y potenciales artificiales [23], cluster space [24].

En [25] se propone un control de sincronización interna que consiste en la retroalimentación de estado entre cada agente individual de un sistema multiagente. Se propone que existe una trayectoria deseada y que se conocen las orientaciones de cada uno de los agentes individuales. Esto significa que el robot líder es manipulado por un operador, mientras que algoritmo hace que la orientación de los robots seguidores siempre converjan a la orientación del robot líder de manera sincronizada.

En cuanto a estos enfoques, en [20, 21] estudian la aproximación líder seguidor, el cual utiliza como referencia una trayectoria definida por el líder. Además, en este trabajo la estabilidad interna de la formación esta implícita en la estabilidad de las leyes de control de los agentes individuales. El problema de esta aproximación es que no presenta una buena robustez frente a perturbaciones. En [22] utilizan el planteamiento de líder virtual, donde los agentes individuales en el entrenamiento sintetizan conjuntamente un único agente líder, posiblemente ficticio, cuya trayectoria actúa como líder para el grupo. Este planteamiento mejora la robustez frente a las perturbaciones que la aproximación líder-seguidor no toma en cuenta. La ventaja de este método es que bajo ciertas condiciones se pueden reducir los hilos de comunicación entre los agentes y aún así evitar colisiones entre ellos.

Por otro lado, en [26] se expuso un algoritmo de control y sincronización, dicho control se basa en métodos de comportamiento y la técnica de acoplamiento cruzado, para con esto reducir el error más significativo, que sería el

error de sincronización. En este algoritmo se considera que se puede medir la orientación y las posición de cada uno de los agentes del sistema multiagente conformado por robots móviles terrestres de tracción diferencial. La desventaja que posee este algoritmo son las posibles singularidades que pueden aparecer para ciertos valores del ángulo de orientación, por lo que para evitar dichas singularidades es necesario definir adecuadamente las trayectorias deseadas (tipo \mathcal{C}^2). Además, debido a que está basado en la técnica del acoplamiento cruzado, el algoritmo de control es formulado dependiendo de la tarea a realizar, por ejemplo mantener una formación elíptica, circular, cuadrada, triangular, entre otras.

El problema de la teoría de sincronización de sistemas dinámicos se ha estudiado extensamente para sistemas representados por dinámicas simples tales como integradores simples o dobles [6, 7, 8, 11] o masas puntuales [27].

En la robótica móvil existen áreas de oportunidad en el estudio de trayectorias permisibles y no permisibles para la tarea de seguimiento, en este trabajo no se aborda dicho estudio, pero se recomiendan los trabajos [28, 29, 30]. Por lo tanto, en este trabajo se utilizan trayectorias de tipo \mathcal{C}^2 ya reportadas en la literatura, las cuales ya han sido validadas de manera numérica y experimental para su uso en el seguimiento para robots móviles.

Por otro lado, para el control de robots móviles es indispensable el cálculo de la matriz Jacobiana y de su inversa para cada robot, existen condiciones para los cuales se presentan problemas numéricos para el cálculo de la misma (no existe), temas que en esta investigación no se analizaron, pero se pueden estudiar en [31, 32, 33]. En este trabajo se consideran condiciones ideales para que el cálculo de la inversa de esa matriz siempre pueda resolverse.

1.3. Objetivos

Objetivo General.

- Realizar un control centralizado para el control de formación y seguimiento de trayectorias de un MAS conformado por N robots móviles terrestres sobre ruedas, cuya posición será determinada por un sistema de visión artificial monocular.

Objetivos Específicos.

- Modelar cinemáticamente con matrices de transformación homogénea entre sistemas de coordenadas.
- Diseñar un control de velocidad para cada uno de los agentes.
- Diseñar un sistema de visión artificial monocular para el cálculo de la posición de cada uno de los agentes.
- Realizar pruebas de seguimiento de trayectorias y control de formación.

1.4. Sinopsis

Esta tesis presenta el modelo cinemático y control de la formación de un sistema multiagente formado por robots móviles terrestres sobre ruedas en configuración diferencial en donde se realimenta su posición por medio de un sistema de visión monocular para implementar un control en lazo cerrado.

Capítulo 2: se estudian los conceptos básicos para la obtención de los modelos matemáticos del robot y de la formación, como lo son coordenadas cartesianas, partes e interacciones de un robot móvil, configuración diferencial de robots móviles terrestres sobre ruedas, obtención de cinemática directa e inversa, además de las definiciones de estabilidad y sistemas multiagente.

Capítulo 3: se muestra el diseño y construcción de la plataforma experimental utilizada en el desarrollo de este proyecto de tesis, posteriormente se detalla el proceso que se siguió para la identificación de las plantas de proceso, una vez realizado lo anterior se sintonizaron varios controladores de bajo orden tipo PI para el control de velocidad del robot seleccionando el mejor de acuerdo al rendimiento de la respuesta, tomando en cuenta validación numérica y experimental. Posteriormente para el modelo del robot se comienza el análisis con el estudio de las restricciones cinemáticas de un robot en configuración diferencial, las consideraciones físicas en las que el modelo cinemático del robot es válido, para después presentar un modelo cinemático no holonómico y un modelo holonómico. Acto seguido, se realiza el diseño, el análisis de estabilidad y la validación numérica de leyes de control para diferentes tareas, como el

control de posición, el seguimiento de trayectorias y el seguimiento de caminos.

Capítulo 4: en este capítulo se presenta el modelo geométrico y la cinemática de formación de un sistema multiagente conformado por dos y tres robots móviles diferenciales, también se presenta el diseño del control, el análisis de estabilidad y la validación numérica para el control de posición y el control de seguimiento de trayectorias de la formación.

Capítulo 5: en este capítulo se validó experimentalmente las leyes de control diseñadas en el Capítulo 3 (para un solo robot móvil) y Capítulo 4 (para la formación del sistema multiagente).

Fundamentos Matemáticos: Sistemas Dinámicos y Sistemas Multi-agente

En este capítulo se presentan los conceptos teóricos para el desarrollo del trabajo. Específicamente se introducen los fundamentos teóricos necesarios para el análisis de la cinemática del robot, el análisis de estabilidad, el control de seguimiento de trayectoria y el estudio de sistemas multiagente.

2.1. Conceptos Matemáticos

2.1.1. Sistema Cartesiano de Referencia

El sistema de referencia que se utiliza para la localización espacial de un robot móvil se define mediante ejes perpendiculares entre sí con un origen definido. Estos se conocen como sistemas cartesianos [34].

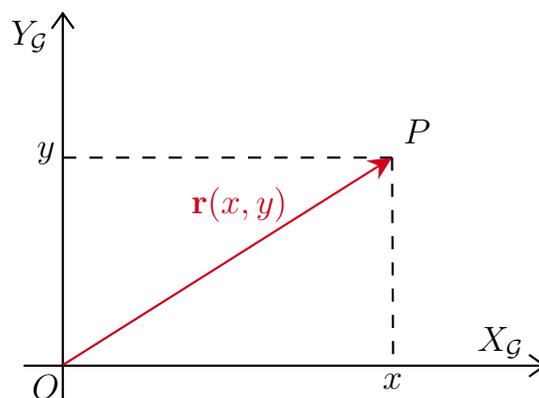


Figura 2.1: Representación de un Vector en Coordenadas Cartesianas.

El sistema de coordenadas global $\mathcal{G} := [X_G \ Y_G]^T$ queda definido por dos vectores coordenados X_G y Y_G que son perpendiculares entre sí con un punto de intersección común denominado O (origen).

2.1.2. Coordenadas Cartesianas

Se analiza la posición de un cuerpo rígido en un sistema de coordenadas $\mathcal{G} := \begin{bmatrix} X_{\mathcal{G}} & Y_{\mathcal{G}} \end{bmatrix}^T$. En este sistema un punto P estará definido por sus coordenadas (x, y) . Este punto en el plano tiene un vector $\mathbf{r}(x, y)$ asociado, que va desde el origen O hasta el punto P (véase la Figura 2.1).

Por lo tanto, la posición del extremo del vector \mathbf{r} (conocido como cabeza) está definido por las dos componentes (x, y) , denominadas coordenadas cartesianas del vector y que son las proyecciones del vector \mathbf{r} sobre los ejes $X_{\mathcal{G}}$ y $Y_{\mathcal{G}}$ (componentes vectoriales).

2.2. Robótica

Un robot es un sistema mecánico-electrónico conformado por actuadores, sensores y un sistema de control como elementos básicos necesarios para cerrar la cadena actuación - medidas - actuación (véase la Figura 2.2). La parte mecánica involucra desde la estructura rígida de un robot móvil hasta las ruedas para el movimiento. La parte electrónica involucra los sensores para la medición de variables, fuente de voltaje y etapa de potencia para los actuadores [34].

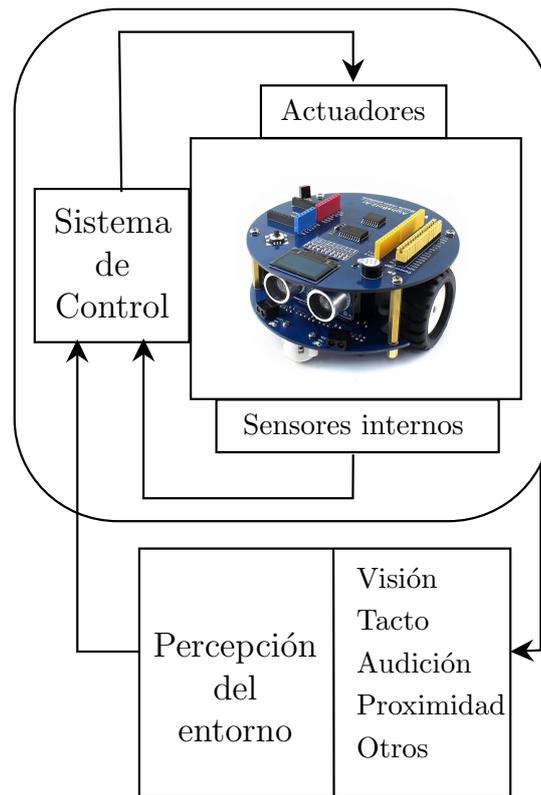


Figura 2.2: Robot y su Interacción con el Entorno.

En la Figura 2.2 se puede observar un diagrama que representa a un robot y su interacción con el ambiente de trabajo, partiendo de lo general a lo particular:

- **Sistema de control:** Éste es el que realiza cálculos utilizando lazos cerrados para la corrección de las variables de interés.
- **Sensores internos:** Permiten obtener el valor numérico de variables medibles para determinar el estado de la estructura mecánica, como orientaciones, desplazamientos, velocidades, fuerzas, pares, etc. Las variables obtenidas por estos sensores son procesadas por el sistema de control.
- **Sensores externos:** Permiten proporcionar sentidos al robot como lo son el tacto, audición, visión, por mencionar algunos.

Dentro de la robótica del procesamiento de la información se involucran tres funciones principales las cuales se enlistan a continuación [34]:

- **Control de movimientos:** Esta función se realiza dentro del sistema de control y se realiza mediante el cálculo de errores y la disminución de los mismos por medio de lazos cerrados y leyes de control.

- **Percepción del entorno:** Esta función hace que un robot tenga la posibilidad de adaptarse a su entorno o ambiente de trabajo mediante las variaciones de sus sentidos, para poder tomar decisiones en situaciones imprevistas. Estas acciones se realizan por medio del sistema de control.
- **Planificación:** Esta función se realiza dentro del sistema de control y tiene como finalidad determinar una trayectoria desde una posición inicial a una posición final, sin colisiones y tratando de optimizar dicha trayectoria, esto para el movimiento de un efector final para manipuladores o del cuerpo de un robot móvil según sea el caso.

2.2.1. Robótica Móvil

La denominada robótica móvil ha cobrado una importancia creciente durante los años ochenta y noventa, ésta área se desarrolló para poder extender el campo de aplicación de la robótica, ya que el área de trabajo ya no es fijo [34, 35, 36].

Un robot móvil es definido como un vehículo de propulsión autónoma y movimiento programado por medio de control automático para realizar una tarea específica (seguimiento de trayectoria, control de velocidad) [34]. Existen diferentes tipos de robots móviles, los cuales se pueden clasificar por su tipo movimiento y su morfología (véase la Figura 2.3).

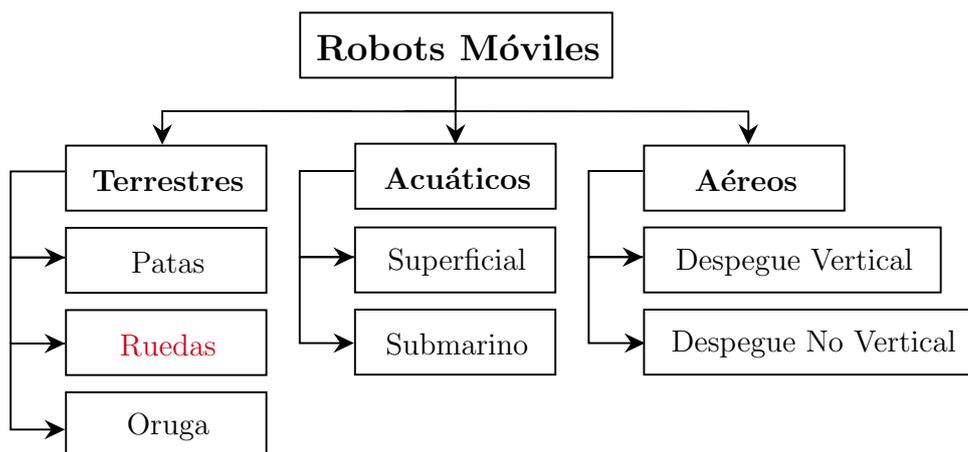


Figura 2.3: Clasificación de los Robots Móviles.

La robótica móvil es un tema de investigación en el que se requiere de la aplicación sinérgica de conocimientos en varias disciplinas. Además, los robots

móviles tienen un sin fin de aplicaciones entre las que resaltan: exploración marítima, reconocimiento de terreno, misiones de búsqueda y rescate de personas, limpieza de desechos peligrosos, asistencia médica y doméstica, actividades de ocio y entretenimiento, por mencionar algunas [34].

En los últimos años, con el objetivo de conformar herramientas más eficaces para la industria de la automatización, es utilizada la integración de varios tipos de robots. Tal es el caso de los robots móviles terrestres sobre ruedas que incorporan un brazo manipulador (robots híbridos) como elemento de seguridad en la inspección y manejo de paquetes sospechosos, para aplicaciones policíacas en neutralización de dispositivos explosivos o para la manipulación de productos químicos peligrosos y materiales radioactivos, entre otras [26, 37].

Robots Móviles Terrestres sobre Ruedas

Los robots móviles terrestres sobre ruedas son la solución más sencilla, rápida de diseñar y construir, además de su bajo costo para conseguir movilidad en terrenos duros y sin obstáculos (superficies planas), permitiendo conseguir velocidades relativamente altas (a comparación de orugas o patas). Uno de los fenómenos presentes y no deseados en los robots sobre ruedas es el deslizamiento entre el punto contacto rueda-suelo.

Los robots móviles terrestres sobre ruedas se pueden estudiar de manera independiente de acuerdo a su tipo de configuración en la tracción. Cada robot tiene restricciones de movilidad y velocidad diferentes, por lo que el análisis y diseño de control es diferente.

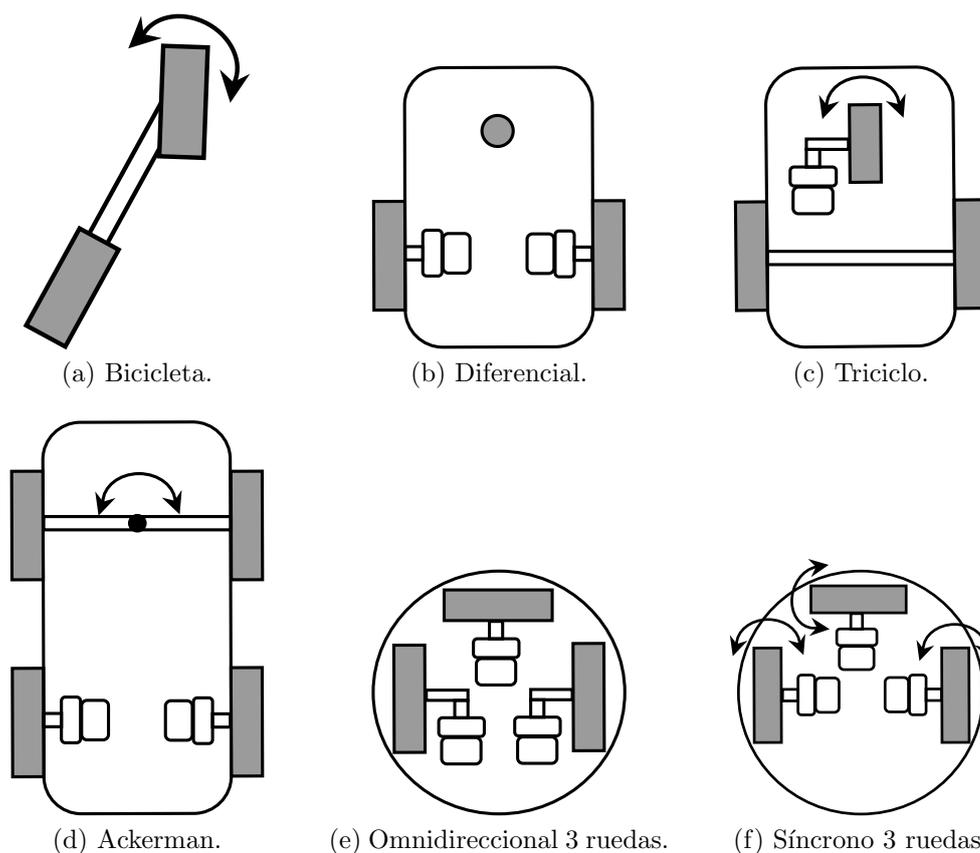


Figura 2.4: Configuración de Robots Móviles Terrestres sobre Ruedas.

En la Figura 2.4 se muestran algunas de las diferentes configuraciones para tracciones en RMR. En la Tabla 2.1 se muestra el tipo de rueda que utiliza y el número de actuadores utilizados.

Tabla 2.1: Componentes de RMR.

Tipo de Robot	Motores			Ruedas	
	Tracción	Dirección	Motriz	Loca	Omnidireccional
Bicicleta	1	1	2	0	0
Diferencial	2	0	2	1	0
Triciclo	1	1	3	0	0
Ackerman	2	1	4	0	0
Omnidireccional	3	0	0	0	3
Síncrono	3	3	3	0	0

En este trabajo se utilizan robots en configuración diferencial, en este tipo de configuración el movimiento se da por la diferencia de velocidades de dos ruedas laterales, las ruedas son montadas en un mismo eje de direccionamiento. Este

tipo de configuración es de mucha utilidad para cambios de direccionamiento, ya que se puede cambiar la orientación sin necesidad de realizar movimientos de traslación, lo que en la literatura se conoce como cambio de *spin* [34]. Las variables de control son las velocidades angulares de la rueda derecha y rueda izquierda. Adicionalmente, el robot posee por lo menos una rueda de soporte conocida como rueda loca, la que ayuda mantener el equilibrio del robot para velocidades relativamente altas.

2.3. Cinemática

La cinemática es la rama de la mecánica que estudia el movimiento de cuerpos rígidos sin considerar las causas que lo originan (masas, fuerzas, etc), y se restringe al análisis de la trayectoria en función del tiempo [38]. Para el estudio de la trayectoria se analizan velocidades y aceleraciones que describen el cambio del cuerpo en función del tiempo.

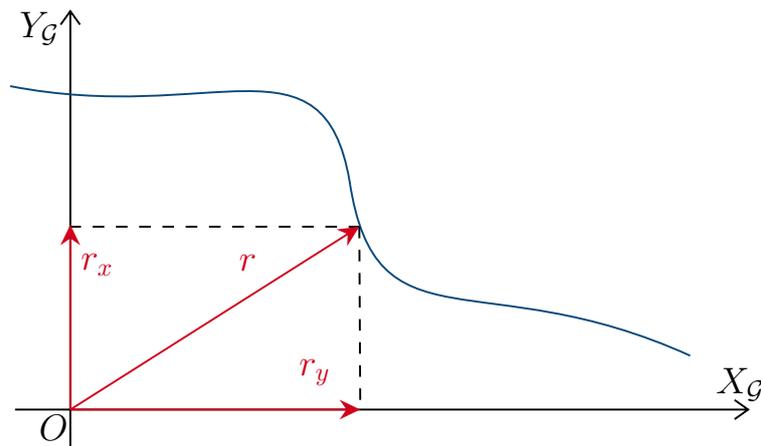


Figura 2.5: Esquema de una Trayectoria.

La estructura cinemática de un robot móvil se puede considerar como un conjunto de cadenas cinemáticas cerradas, tantas como ruedas en contacto con el suelo. Asimismo, la interacción rueda-suelo se define, desde el punto de vista cinemático, como una articulación planar con tres grados de libertad, donde uno de ellos, generalmente sin controlar, representa los deslizamientos laterales. Estos dos hechos dificultan la construcción del modelo, ya que se dan efectos no presentes en los manipuladores. En concreto, cobra gran importancia la perfecta sincronización de la velocidad de las ruedas para alcanzar una

localización determinada.

Por otro lado, para obtener modelos cinemáticos diferenciales para diferentes tipos de RMR, se desea adquirir las relaciones cinemáticas entre los distintos tipos de variables presentes en el robot móvil.

- Restringiendo el movimiento del robot a un plano, la posición de un punto del robot esta completamente definida por tres variables escalares, dos lineales y una angular (x, y, φ) , cuya forma vectorial se denomina *postura del robot* (\mathbf{p}).
- Su derivada de primer orden con respecto del tiempo ($\dot{\mathbf{p}}$) se denomina *vector de velocidad del robot* (o variables de estado).
- Las articulaciones de actuación, dirección y rotación del robot (ruedas) poseen variables que se denominan *velocidad de la rueda* (ω_R, ω_L) , las cuales se pueden promediar para obtener la velocidad lineal y velocidad angular (v, ω) el *vector de velocidad de control* $\dot{\mathbf{q}}$.

De manera más detallada existen trabajos que se centran en el modelado de las ruedas, de manera anticipada al modelo del robot completo [39].

2.3.1. Cinemática de Robots

La cinemática de robots se encarga de la configuración de éstos en su espacio de trabajo, de las relaciones entre sus parámetros geométricos y de las restricciones (en sus velocidades) impuestas en sus trayectorias. Las relaciones cinemáticas dependen de la estructura geométrica del robot y, tal y como se indica en [40], el análisis cinemático de un robot es un prerequisite fundamental para el estudio de su dinámica, las características de su estabilidad y el objetivo de control del robot.

2.3.2. Cinemática Directa e Inversa de un Robot Móvil

Considere un robot móvil con coordenadas generalizadas q_1, q_2, \dots, q_n en el espacio de actuación (coordenadas articulares) y x_1, x_2, \dots, x_m en el espacio de localización/tarea (coordenadas de posición y orientación). Se definen los vectores:

$$\mathbf{q} := \begin{bmatrix} q_1 \\ q_2 \\ \vdots \\ q_n \end{bmatrix}, \quad \mathbf{p} := \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_m \end{bmatrix}. \quad (2.1)$$

El problema de calcular \mathbf{p} conociendo \mathbf{q} se denomina *cinemática directa*. En general, para $\mathbf{p} \in \mathbb{R}^n$ y $\mathbf{q} \in \mathbb{R}^m$, existe una relación descrita por una función (en general, no lineal), que representa al modelo del robot, dicha relación se puede representar mediante la ecuación:

$$\mathbf{p} = \mathbf{f}(\mathbf{q}), \quad \mathbf{f}(\mathbf{q}) = \begin{bmatrix} f_1(\mathbf{q}) \\ f_2(\mathbf{q}) \\ \vdots \\ f_m(\mathbf{q}) \end{bmatrix}. \quad (2.2)$$

El problema para resolver (2.2), es decir, de calcular \mathbf{q} a partir de \mathbf{p} , se conoce como *cinemática inversa* y se define como:

$$\mathbf{q} = \mathbf{f}^{-1}(\mathbf{q}) \mathbf{p}. \quad (2.3)$$

La Figura 2.6 muestra en diagrama de bloques los problemas de cinemática directa e inversa.

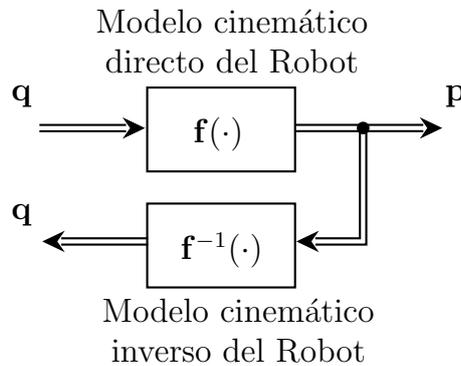


Figura 2.6: Diagrama de Bloques del Modelo Cinemático.

Para calcular los movimientos del robot, se deben definir las variables de control expresadas por las velocidades $\dot{\mathbf{q}} := \begin{bmatrix} \dot{q}_1 & \dot{q}_2 & \cdots & \dot{q}_n \end{bmatrix}^T$. Por lo tanto, se debe encontrar la relación diferencial entre \mathbf{q} y \mathbf{p} , este problema se le conoce como *cinemática diferencial directa* y se define como:

$$\dot{\mathbf{p}} = \mathbf{J}(\mathbf{q}) \dot{\mathbf{q}}, \quad (2.4)$$

donde:

$$\dot{\mathbf{q}} := \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \vdots \\ \dot{q}_n \end{bmatrix}, \quad \dot{\mathbf{p}} := \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \vdots \\ \dot{x}_m \end{bmatrix},$$

y la matriz $\mathbf{J} \in \mathbb{R}^{m \times n}$ se define como:

$$\mathbf{J} := \begin{bmatrix} \frac{\partial x_1}{\partial q_1} & \frac{\partial x_1}{\partial q_2} & \cdots & \frac{\partial x_1}{\partial q_n} \\ \frac{\partial x_2}{\partial q_1} & \frac{\partial x_2}{\partial q_2} & \cdots & \frac{\partial x_2}{\partial q_n} \\ \frac{\partial x_m}{\partial q_1} & \frac{\partial x_m}{\partial q_2} & \cdots & \frac{\partial x_m}{\partial q_n} \end{bmatrix} = [J_{ij}], \quad (2.5)$$

con (i, j) el elemento $J_{ij} := \frac{\partial x_i}{\partial q_j}$: Esta matriz se le conoce como *matriz jacobiana* del robot. La matriz jacobiana relaciona las velocidades en el punto de interés con las velocidades de actuación, esta matriz es indispensable para el análisis y diseño de algoritmos de control.

Para cada conjunto $\{q_1, q_2, \dots, q_n\}$ del robot, la matriz jacobiana representa la relación de los desplazamientos de las ruedas con el desplazamiento de la localización del robot en el espacio de tarea.

Análogamente a la cinemática inversa, al problema de calcular $\dot{\mathbf{p}}$ conociendo $\dot{\mathbf{q}}$ se le conoce como *cinemática diferencial inversa*. Si $m = n$, \mathbf{J} es cuadrada y la matriz jacobiana inversa \mathbf{J}^{-1} existe si y sólo si el $\det(\mathbf{J}) \neq 0$, entonces se puede obtener:

$$\dot{\mathbf{q}} = \mathbf{J}^{-1}(\mathbf{q}) \dot{\mathbf{p}}, \quad (2.6)$$

No obstante, en general la matriz jacobiana \mathbf{J} no es cuadrada, y en consecuencia su inversa no estará definida. Por tanto, para resolver el problema de cinemática diferencial inversa se recurre al cálculo de la *matriz pseudoinversa*. En este contexto, la cinemática diferencial inversa quedará definida mediante la siguiente expresión:

$$\dot{\mathbf{q}} = \mathbf{J}^\dagger(\mathbf{q}) \dot{\mathbf{p}}, \quad (2.7)$$

donde \mathbf{J}^\dagger denota a la pseudoinversa de \mathbf{J} .

La Figura 2.7 se ilustra en diagrama de bloques los problemas de cinemática diferencial directa e inversa.

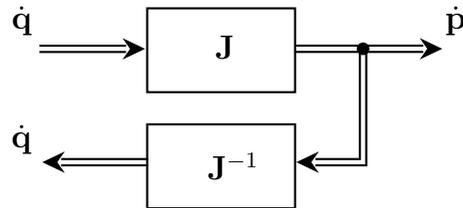


Figura 2.7: Diagrama de Bloques del Modelo Cinemático Diferencial.

2.4. Estabilidad

La teoría de Lyapunov es una de las herramientas fundamentales para el análisis de estabilidad [41, 42]. En efecto, esta nos permite estudiar el comportamiento cualitativo de las soluciones de cualquier sistema dinámico, es decir, de sistemas lineales o no lineales. Adicionalmente, esta teoría se puede utilizar en el diseño de controladores estabilizantes.

El objetivo de este método es encontrar una función candidata de Lyapunov $\mathcal{V}(\mathbf{x})$, la cual es función de los estados del sistema (\mathbf{x}), y se elige de tal modo que cumpla con ciertas condiciones asegurando así la estabilidad en sus puntos de equilibrio.

Teorema 2.4.1: Estabilidad de Lyapunov

Sea $\mathbf{x}^* = 0$ un punto de equilibrio del sistema

$$\dot{\mathbf{x}} = f(\mathbf{x}), \quad (2.8)$$

y sea \mathcal{D} un dominio tal que $\mathbf{x}^* \in \mathcal{D}$. Asúmase que $\mathcal{V} : \mathcal{D} \rightarrow \mathbb{R}_+$ es una función continuamente diferenciable que satisface lo siguiente:

(i) $\mathcal{V}(0) = 0 \wedge \mathcal{V}(\mathbf{x}) > 0, \quad \forall \mathbf{x} \in \mathcal{D} \setminus \{0\}$.

(ii) $\dot{\mathcal{V}} \leq 0, \quad \forall \mathbf{x} \in \mathcal{D}$.

Entonces, $\mathbf{x}^* = 0$ es estable (E). Más aún, si (2.4) se sustituye por:

(ii)' $\dot{\mathcal{V}} < 0, \quad \forall \mathbf{x} \in \mathcal{D} \setminus \{0\}$.

Entonces, $\mathbf{x}^* = 0$ es asintóticamente estable (AE).

2.5. Sistemas Multiagente

Partiendo de la necesidad de cumplir con tareas más complejas, surge el término sistema multiagente (MAS por sus siglas en inglés), el cual es un sistema compuesto por múltiples agentes inteligentes que interactúan entre si [43].

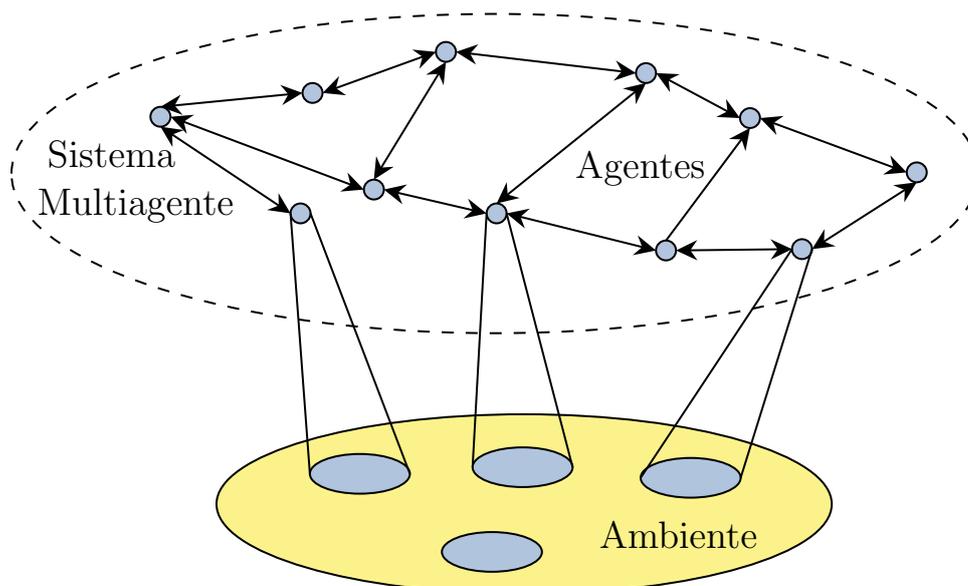


Figura 2.8: Esquema de un Sistema Multiagente.

En la Figura 2.8 se puede observar que los agentes son capaces de actuar en un ambiente; donde cada agente actuará o influenciará diferentes partes del ambiente. Los sistemas multiagente pueden ser utilizados para resolver problemas que son difíciles o imposibles de resolver para un agente individual [43].

Modelado y Control de un Robot Móvil

La deducción de un modelo matemático preciso es una parte crucial del diseño del sistema de control de cualquier planta física, como lo son los robots móviles. En esta sección se presenta el diseño y construcción de la plataforma experimental, como la función de transferencia que aproxima su dinámica, además se define la cinemática de un robot terrestre sobre ruedas de accionamiento diferencial tomando en cuenta las diferencias entre el modelo no holonómico y el modelo holonómico.

3.1. Plataforma Experimental: Diseño y Construcción del Robot

En esta sección se presenta el diseño y construcción de la plataforma experimental, desde la selección de sus componentes mecánicos y electrónicos así como su diseño en dibujo asistido por computadora, por último se muestra la plataforma construida.

3.1.1. Actuadores

El robot móvil terrestre sobre ruedas en configuración diferencial, tiene dos grados de libertad, el cual posee movimientos lineales (adelante y atrás) y el giro del robot (sentido horario y sentido antihorario).

Para la construcción del robot se seleccionaron dos motores de corriente directa, los cuales poseen una caja de engranes reductora y sus sensores de velocidad (encoder) ya incluidos. El modelo del motor que se eligió es de la marca CHIHAI MOTOR: CHR-GM25-370 (véase la Figura 3.1) con características de velocidad de 600 RPM, un voltaje de alimentación de 12 V y un consumo de corriente de 0.2 A.



Figura 3.1: Motor CHIHAI CHR-GM25-370.

Su datasheet se muestra en la Figura 3.2, en cual se observan sus características de rendimiento con carga y sin carga.

Voltage: 12V						
Reduction Ratio	1:4.4	1:10	1:20	1:34	1:20	1:45
No load Current(mA)	≤200	≤200	≤200	≤200	≤130	≤150
No load Speed(rpm)	2700	1290	600	350	220	140
Rated Torque(kg.cm)	0.8	1.7	3.5	5.8	2.9	4.3
Rated Torque(N.m)	0.08	0.17	0.34	0.57	0.28	0.42
Rated Speed(rpm)	1600	700	350	180	130	70
Rated Current(A)	≤1.5	≤1.5	≤1.5	≤1.7	≤1.0	≤0.8
Stall Current (A)	≤5.6	≤5.6	≤5.6	≤5.6	≤2.5	≤2.5
Stall Torque(kg.cm)	1.8	3.5	7.0	12.0	5.0	8.5
Hall resolution(PPR)	44	102.3	224.5	341.2	224.5	493.9
Length of gearbox	19	19	19	21	19	21

Figura 3.2: Datasheet del Motor CHR-GM25-370.

Que el robot posea sus propios encoders benefició la retroalimentación de las velocidades totales del robot y eso favoreció el rendimiento del controlador interno de velocidad (el cual quedó embebido en el robot).

3.1.2. Sensor de Velocidad

El propio motor ya posee un encoder magnético (véase la Figura 3.3) el cual permite detectar el sentido de giro y la ventaja es que ya está acoplado al eje del motor. El encoder de cuadratura es un tipo de encoder rotativo incremental el cual permite calcular la posición, velocidad y la dirección del movimiento del eje del motor.

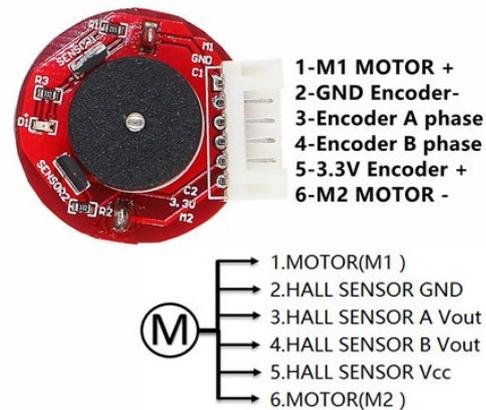


Figura 3.3: Encoder de Cuadratura del Motor CHR-GM25-370.

En la Figura anterior se observa que el encoder proporciona dos señales digitales para indicar el sentido de giro del motor, por lo que se utilizó una compuerta XOR para reducir el número de interrupciones necesarias para su lectura, de dos señales a solo una.

3.1.3. Driver de Potencia

Después de seleccionar el motor de corriente directa, se observa que existen varios modelos de drivers comerciales de potencia los cuales cumplen con las necesidades eléctricas propias del motor CHR-GM25-370 (véase la Figura 3.2).

El driver que se eligió es el módulo TB6612FNG (véase la Figura 3.4). El módulo TB6612FNG posee dos salidas tipo puente H independientes el cual permite controlar el giro y velocidad de dos motores de corriente directa de manera independiente. El módulo está conformado del mosfet TB6612FNG, diodos de protección y un regulador 78M05 (permite 5 V máximo). El driver soporta una alimentación entre 3 y 13.5 V, una corriente máxima de consumo de 1.2 A por puerto, por lo tanto cumple con lo que se necesita.

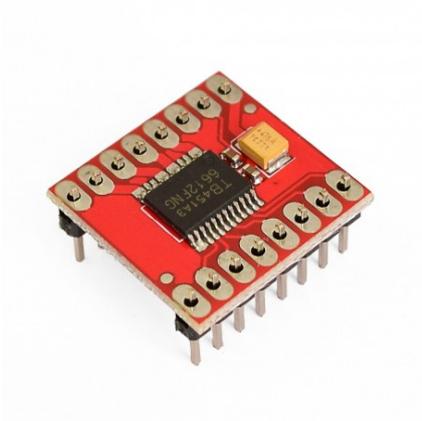


Figura 3.4: Módulo TB6612FNG.

El driver posee seis pines controlados por tecnología TTL, por lo que se necesitaron cuatro señales digitales para el sentido de giro, dos señales PWM para la velocidad.

3.1.4. Comunicación Inalámbrica

Para la comunicación inalámbrica se necesita monitorear señales del robot (velocidades) desde la PC. Se seleccionó el módulo BT HC05 (véase la Figura 3.5) el cual es un dispositivo de comunicación serial inalámbrica, este modelo es configurable para poderlo utilizar como esclavo o maestro.



Figura 3.5: BT HC05.

El modulo seleccionado necesitó un puerto serial (dos señales digitales TX y RX) además de un pin digital para el Enable (activación) del módulo.

3.1.5. Sistema de Control

El sistema de control permite realizar el procesamiento de las señales que recibe de los sensores y mediante el cálculo de algunas variables con el uso de los actuadores realizar un movimiento deseado en el robot (adelante, atrás, giro).

Para el sistema control se tuvo la necesidad de poder leer las señales y periféricos anteriormente mencionados, los cuales se tomaron en cuenta para la elección del mismo:

- 1 puerto serial (TX y RX) para la comunicación inalámbrica.
- 4 pines digitales (I/O) para el sentido de giro del robot.
- 2 pines PWM para el control de la velocidad del motor.
- 2 pines digitales (I/O) para la adquisición de las señales del encoder del motor.
- 2 pines para interrupciones generadas por los encoder.

Existen variedad de tarjetas de adquisición de datos (DAQ) y microcontroladores que cumplen con esas características. Se eligió una opción de bajo costo, fácil programación y de rendimiento regular como lo es la plataforma de desarrollo Arduino MEGA2560 (véase la Figura 3.6) el cual posee un microcontrolador de 8 bits de modelo ATMEGA2560.

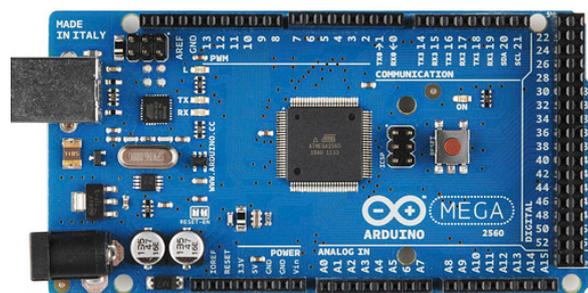


Figura 3.6: Arduino MEGA.

- Velocidad del oscilador 16MHz
- 54 pines I/O digitales
- 16 pines PWM

- 16 pines I/O analógicos
- 24 interrupciones
- 4 puertos seriales

PCB

Se realizó el diseño de la PCB para un fácil montaje en la tarjeta de control. Los componentes posibles a conectar es:

- 1 Bluetooth HC05.
- 1 Driver TB6612FNG.
- 1 Compuerta XOR 74LS86N.
- 2 Encoder de cuadratura.
- 1 Acelerómetro MPU6050.
- 3 Baterías 18650 3.7 V.

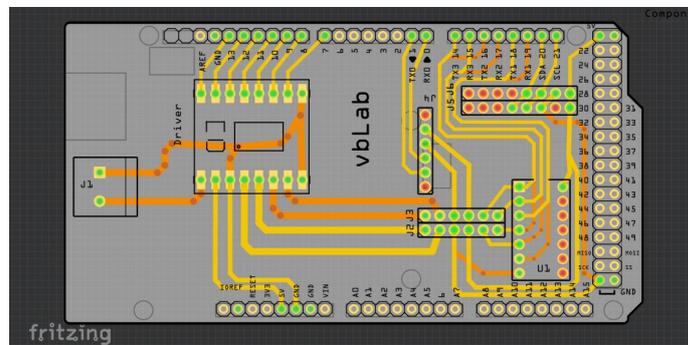


Figura 3.7: PCB Compatible con Arduino MEGA.

El diseño de la PC se realizó en el software fritzing[®] y se produjeron en una empresa local de Zacatecas, México.

Dibujo CAD

Para integrar todos los componentes en un chasis a la medida se diseñó un modelo de robot móvil sobre ruedas con tracción diferencial de acuerdo a las necesidades de espacio en el software SOLIDWORKS®. El resultado se muestra en la Figura 3.8.

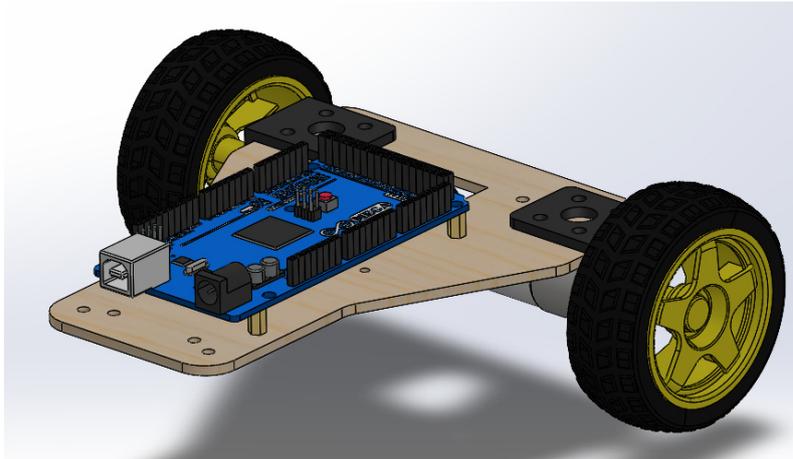


Figura 3.8: Robot en Dibujo Asistido por Computadora.

Robot Real

Por último, se construyó el robot mostrado en la Figura 3.9 agregándole una marca de color amarillo para su mejor identificación por el algoritmo de tracking del sistema de visión.



Figura 3.9: Robot Construido.

3.2. Identificación de la Dinámica del Motor

El objetivo de esta sección fue encontrar un modelo de orden reducido que represente el motor del robot a partir de la curva de reacción del proceso, para posteriormente utilizar este modelo para la sintonización de los controladores. Por lo tanto es necesario conocer el comportamiento dinámico del motor.

La aproximación que se tomó es de un modelo de primer orden con retardo. El modelo en el dominio de la frecuencia que se obtuvo es el siguiente:

$$G(s) := \frac{\Omega(s)}{V(s)} = \frac{ke^{-hs}}{Ts + 1} \quad (3.1)$$

donde $k > 0$, $h > 0$ y $T \in \mathbb{R} \setminus \{0\}$.

La función de transferencia (3.1) se obtuvo de la salida Ω (velocidad angular) entre la entrada V (voltaje de referencia). Teniendo el modelo anterior se deseaba identificar los parámetros constantes: la ganancia de lazo abierto k , el retardo h y la constante de tiempo T , todo se hizo mediante un algoritmo de identificación (Mínimos Cuadrados Recursivos, Optimización por Enjambre de Partículas, Caja Negra, etc). Cualquier algoritmo funciona a partir de la curva de respuesta del sistema en lazo abierto y puede funcionar en línea o fuera de línea, por lo tanto se le aplica un señal de entrada de tipo escalón por un tiempo definido, y a partir de esto se obtiene la velocidad angular del motor [rad/s].

Teniendo esto, se tuvo que identificar la zona muerta del motor. Se entiende como zona muerta de un motor a la mínima señal de entrada en este caso el ciclo de trabajo (DC de Duty Cycle por sus siglas en inglés) para que el sistema responda o de una salida ($y(t)$) diferente de cero.

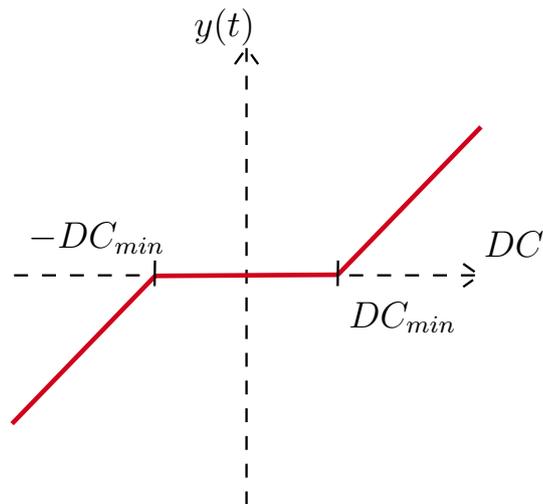


Figura 3.10: Zona Muerta del Motor.

La gráfica anterior representa la respuesta/salida (velocidad angular) vs la entrada (ciclo de trabajo). Esta zona muerta se debe compensar en el algoritmo de control para que el sistema se mueva con la velocidad de referencia a velocidades bajas.

Teniendo esto se programó el Arduino MEGA[®] para controlar el sentido de giro de los motores, la lectura de los encoders además del envío y recepción de datos vía serial, por otro lado en la PC por medio de *Python* y protocolo serial (véase la Figura 3.11), se envió la señal de referencia y se recibió la respuesta del motor), con los datos obtenidos fuera de línea se utilizó el algoritmo de identificación conocido como Optimización por Enjambres de Partículas (PSO por sus siglas en inglés), con el cual se obtuvieron las siguientes respuestas.

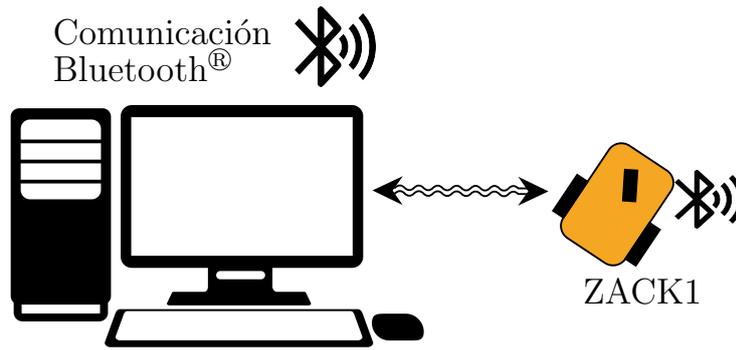


Figura 3.11: Diagrama de Comunicaciones para Experimentos de Identificación.

Robot No. 1

Las pruebas se realizaron con una normalización de 0 a 100 % en la velocidad de entrada y salida de cada uno de los motores, donde el rango de velocidad es de 0 a 60 rad/s.

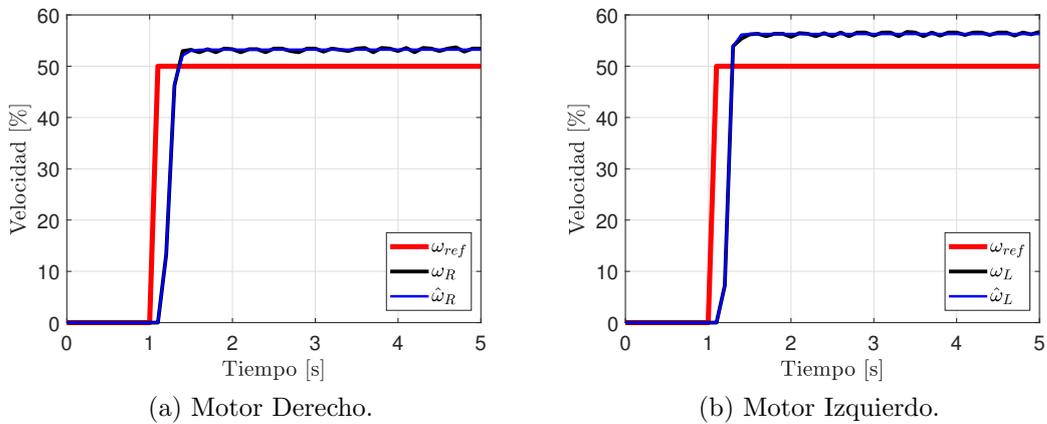


Figura 3.12: Respuesta en Lazo Abierto de los Motores del Robot No. 1.

Las constantes del modelo (3.1) que se obtuvieron para el Robot No. 1 se muestran en la Tabla 3.1:

Tabla 3.1: Constantes del Modelo del Robot No. 1.

Motor	Constantes del Modelo		
	k	h	T
Derecho	1.0646	0.084	0.056
Izquierdo	1.1257	0.0954	0.0333

Parametrizando el modelo (3.1) se obtuvieron las siguientes funciones de transferencia para el Robot No. 1:

$$G_{R1}(s) := \frac{1.0646e^{-0.084s}}{0.056s + 1}, \quad G_{L1}(s) := \frac{1.1257e^{-0.0954s}}{0.0333s + 1}. \quad (3.2)$$

donde G_{R1} representa el modelo del motor derecho y G_{L1} representa el modelo del motor izquierdo.

Robot No. 2

La identificación para el robot No. 2 se realizó con la misma metodología, los resultados se muestran a continuación:

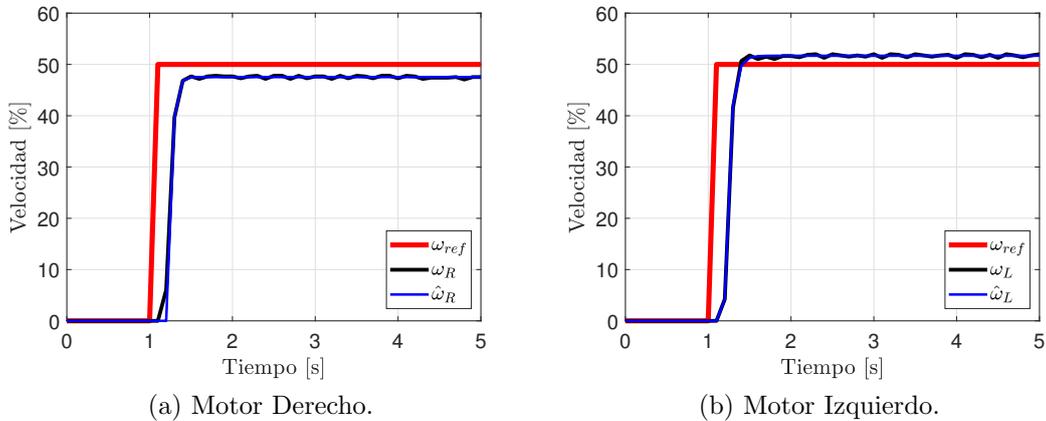


Figura 3.13: Respuesta en Lazo Abierto de los Motores del Robot No. 2.

Las constantes del modelo (3.1) que se obtuvieron del Robot No. 2 se muestran en la Tabla 3.2:

Tabla 3.2: Constantes del Modelo del Robot No. 2.

Motor	Constantes del Modelo		
	k	h	T
Derecho	0.9507	0.1257	0.0412
Izquierdo	1.0339	0.0947	0.0626

Parametrizando el modelo (3.1) se obtuvieron las siguientes funciones de transferencia para el Robot No. 2:

$$G_{R2}(s) := \frac{0.9507e^{-0.1257s}}{0.0412s + 1}, \quad G_{L2}(s) := \frac{1.0339e^{-0.0947s}}{0.0626s + 1}. \quad (3.3)$$

donde G_{R2} representa el modelo del motor derecho y G_{L2} representa el modelo del motor izquierdo.

3.3. Control de Velocidad del Motor

En esta sección se muestran las metodologías para diseñar diferentes controladores de bajo orden tipo PI, para manipular las variables de velocidad de cada uno de los motores, que a su vez por medio de la diferencias de velocidades se calcula la velocidad lineal y angular total del robot móvil. Este controlador se encontrará embebido en el microcontrolador de la plataforma experimental.

3.3.1. Controlador PI

Se define un controlador PI en su forma estándar con función de transferencia [44]:

$$C(s) := \left(K_p + \frac{K_i}{s} \right). \quad (3.4)$$

Y con diagrama de bloques mostrado en la Figura 3.14

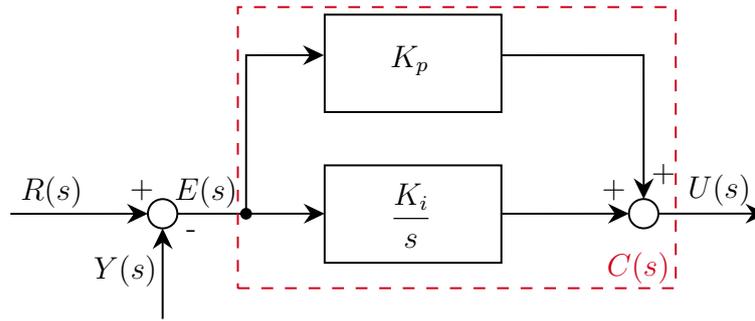


Figura 3.14: Diagrama de Bloques Controlador PI Estándar.

Controladores $PI\sigma$

Basándose en el concepto de σ -estabilidad y la teoría de \mathcal{D} -particiones [45, 46, 47, 48], se pueden diseñar controladores PI que permiten obtener el máximo decaimiento exponencial alcanzable en la respuesta de un sistema SISO (una entrada - una salida) en lazo cerrado. Dicho controlador se nombra $PI\sigma$, para el cálculo de las constantes K_i y K_p se utilizaron las fórmulas de diseño que se pueden encontrar en [49].

En las Figuras 3.15 y 3.16 se muestran las regiones de estabilidad de cada planta y cada robot respectivamente, las curvas de nivel describen regiones que contienen diferentes decaimientos exponenciales, siendo la más pequeña la región con las ganancias del controlador $PI\sigma$.

Robot No. 1

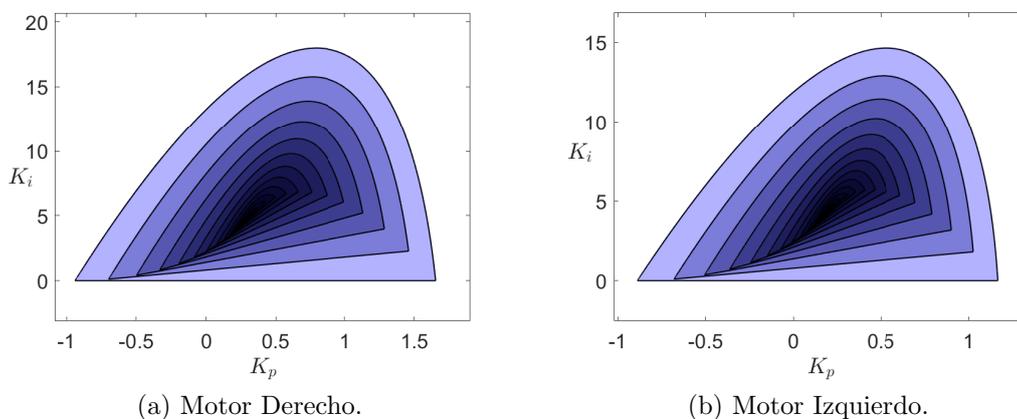


Figura 3.15: Región de Estabilidad en el Plano Paramétrico K_p vs K_i del Robot No. 1.

Las constantes de los controladores PI que se obtuvieron se muestran en la Tabla 3.3:

Tabla 3.3: Constantes del Control $PI\sigma$ Robot No. 1.

Motor	Constantes del Controlador $PI\sigma_1$	
	K_p	K_i
Derecho	0.2384	4.2144
Izquierdo	0.1519	3.8731

Robot No. 2

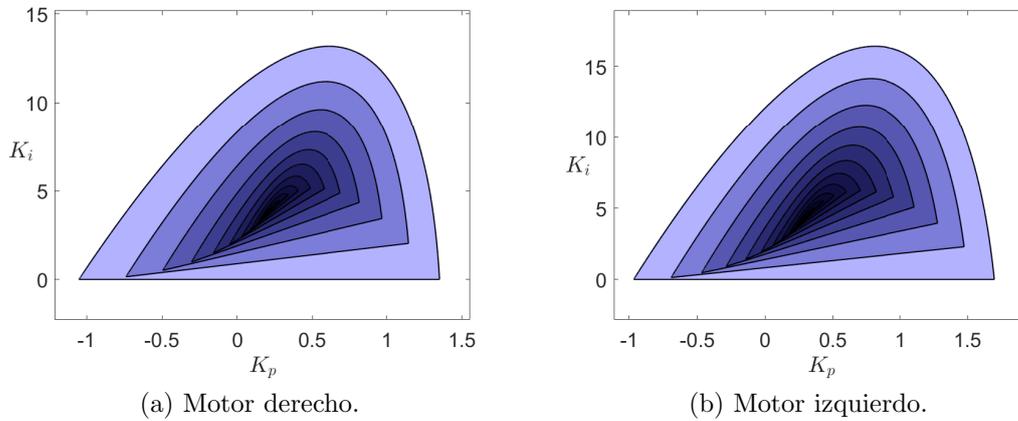


Figura 3.16: Región de Estabilidad en el Plano Paramétrico K_p vs K_i del Robot No. 2.

Las constantes de los controladores PI que se obtuvieron se muestran en la Tabla 3.4:

Tabla 3.4: Constantes del Control $PI\sigma$ Robot No. 2.

Motor	Constantes del Controlador $PI\sigma_2$	
	K_p	K_i
Derecho	0.1755	3.5183
Izquierdo	0.2438	3.8528

Controladores $PI\sigma$ Arbitrario

Utilizando las Figuras 3.15 y 3.16 se seleccionaron ganancias que se encuentran dentro de la región de estabilidad, para la sintonización de un nuevo controlador

que se nombrará PIa. Las constantes que se seleccionaron para el Robot No. 1 y Robot No. 2 se muestran en las Tablas 3.5 y 3.6 respectivamente.

Tabla 3.5: Constantes del Control PI σ Arbitrario Robot No. 1.

Motor	Constantes del Controlador PIa ₁	
	K_p	K_i
Derecho	0.7204	15.82
Izquierdo	0.4813	12.54

Tabla 3.6: Constantes del Control PI σ Arbitrario Robot No. 2.

Motor	Constantes del Controlador PIa ₂	
	K_p	K_i
Derecho	0.4584	9.486
Izquierdo	0.7056	12.27

Controladores PI λ

Un control que tome en cuenta la incertidumbre de los parámetros de estimación de la planta es de gran utilidad. Una alternativa común, es encontrar un controlador robusto, el cual se puede encontrar por la técnica de sintonización *Lambda*. A dicho controlador se le nombró PI λ , sus constantes se calcularon mediante las siguientes fórmulas [50, 51, 52]:

$$K_c := \frac{T}{k(h + \lambda)}, \quad (3.5)$$

$$T_i := T,$$

donde λ es un factor que afecta la velocidad en la respuesta, aumentando el valor disminuye la velocidad, y disminuyendo el valor aumenta la velocidad. El autor [51] recomienda $\lambda \approx 3T$ para obtener un controlador robusto, entre λ sea mas pequeño, la velocidad en la respuesta aumentará. Las constantes que se calcularon para el Robot No. 1 y Robot No. 2 se muestran en las Tablas 3.7 y 3.8 respectivamente.

Tabla 3.7: Constantes del Control PI λ Robot No. 1.

Motor	Constantes del Controlador PI λ_1	
	K_p	K_i
Derecho	0.2087	2.2361
Izquierdo	0.1515	2.7297

Tabla 3.8: Constantes del Control PI λ Robot No. 2.

Motor	Constantes del Controlador PI λ_2	
	K_p	K_i
Derecho	0.1738	2.5311
Izquierdo	0.2143	2.0540

Controladores PIH

Otro método popular de sintonización de controladores PI es el método de Haalman [53]. Esta técnica se diseñó para el uso en la industria de laminación de hierro con el fin de obtener una respuesta críticamente amortiguada con un buen rendimiento en la de seguimiento de referencia. Esta técnica funciona aceptablemente con sistemas del tipo (3.1). Las constantes del controlador PIH se obtienen de las siguientes ecuaciones [52]:

$$K_c := \frac{2T}{3kh}, \quad (3.6)$$

$$T_i := T.$$

La principal característica del método de ajuste de Haalman es que los polos y ceros del sistema se cancelan con los polos y ceros del controlador. No obstante, como inconveniente, a menudo se observan malos resultados en el rechazo de las perturbaciones de la carga cuando se utilizan las ganancias del controlador obtenidas por este método. Las constantes que se calcularon para el Robot No. 1 y Robot No. 2 se muestran en las Tablas 3.9 y 3.10 respectivamente.

Tabla 3.9: Constantes del Control PI Halman Robot No. 1.

Motor	Constantes del Controlador PIH $_1$	
	K_p	K_i
Derecho	0.4175	7.4554
Izquierdo	0.2067	6.2072

Tabla 3.10: Constantes del Control PI Halman Robot No. 2.

Motor	Constantes del Controlador PIH ₂	
	K_p	K_i
Derecho	0.2268	5.5777
Izquierdo	0.4262	6.8083

Después de que se realizó la sintonización de los diferentes controladores PI, se recabaron las constantes en la Tabla 3.11.

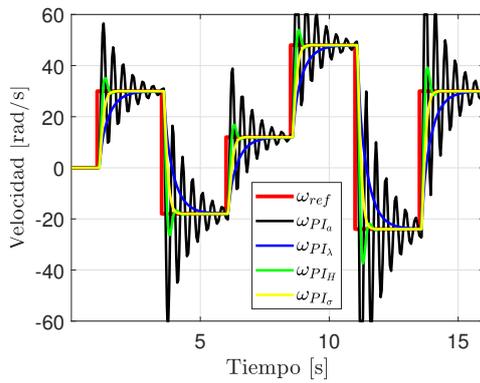
Tabla 3.11: Constantes de los Controladores PI

Motor	Constantes	Controladores PI				
		$PI\sigma$	PIa	$PI\lambda$	PIH	
Robot No. 1	Derecho	K_p	0.2384	0.7204	0.2087	0.4175
		K_i	4.2144	15.82	2.2361	7.4554
	Izquierdo	K_p	0.1519	0.4813	0.1515	0.2067
		K_i	3.8731	12.54	2.7297	6.2072
Robot No. 2	Derecho	K_p	0.1755	0.4584	0.1738	0.2298
		K_i	3.5183	9.486	2.5311	5.5777
	Izquierdo	K_p	0.2438	0.7056	0.2143	0.4262
		K_i	3.8528	12.273	2.9540	6.8083

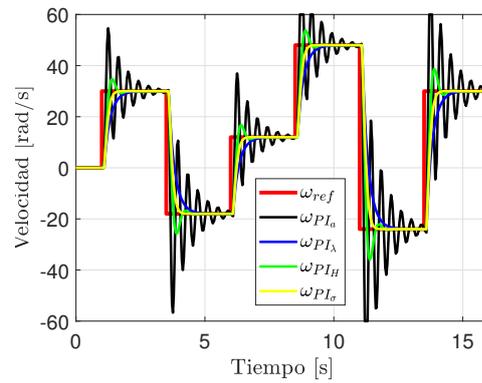
Posteriormente, se realizaron las simulaciones necesarias para analizar el rendimiento de la respuesta, tomando en cuenta propiedades como la velocidad de convergencia del error a cero además del máximo sobre impulso presente. La señal de prueba que se utilizó se define como:

$$\omega_{ref}(t) := \begin{cases} 0, & 0 < t \leq 1 \\ 30, & 1 < t \leq 3.5 \\ -18, & 3.5 < t \leq 6 \\ 12, & 6 < t \leq 8.5 \\ 48, & 8.5 < t \leq 11 \\ -24, & 11 < t \leq 13.5 \\ 30, & t \geq 13.5 \end{cases} .$$

Las respuestas de los diferentes controladores PI se muestran en las Figuras 3.17 y 3.18.

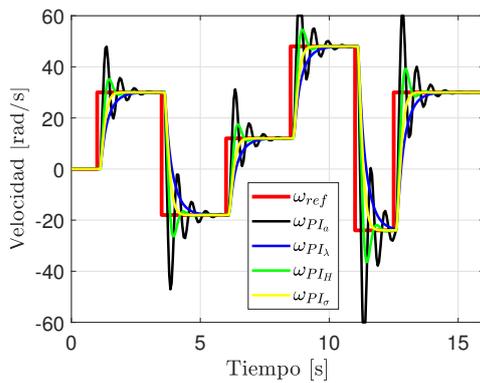


(a) Motor derecho.

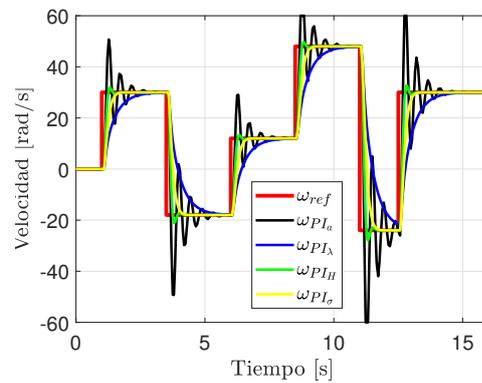


(b) Motor izquierdo.

Figura 3.17: Respuestas Numéricas del Robot No. 1 a Controladores PI.



(a) Motor derecho.



(b) Motor izquierdo.

Figura 3.18: Respuestas Numéricas del Robot No. 2 a Controladores PI.

A su vez, para realizar una validación experimental se programaron los controladores PI en el Arduino MEGA[®] de manera de dejarlos embebidos para el control de velocidad de cada uno de los motores. Para realizar las pruebas se utilizó la plataforma Python, desde la terminal serial de la computadora se envió la velocidad referencia correspondiente a un escalón de amplitud 15 rad/s (véase la Figura 3.11). Los resultados obtenidos para cada robot móvil se muestran en las Figuras 3.19 y 3.20.

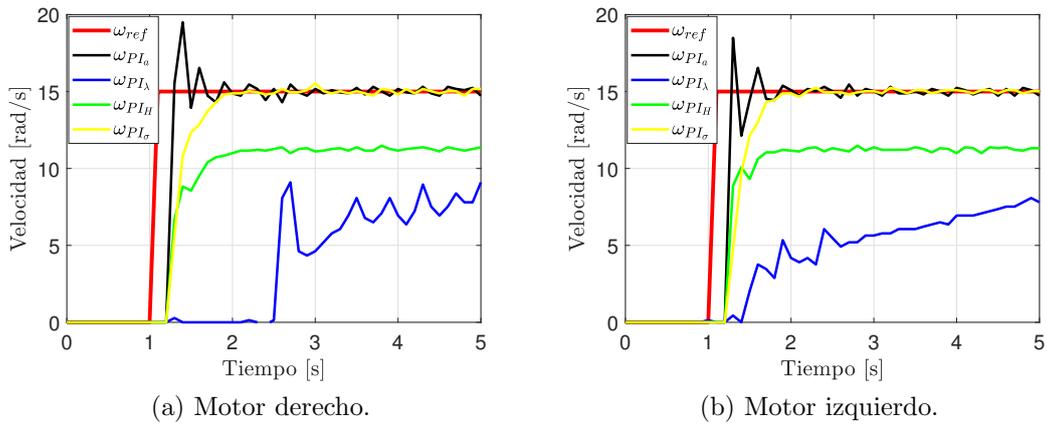


Figura 3.19: Respuestas Experimentales del Robot No. 1 a Controladores PI.

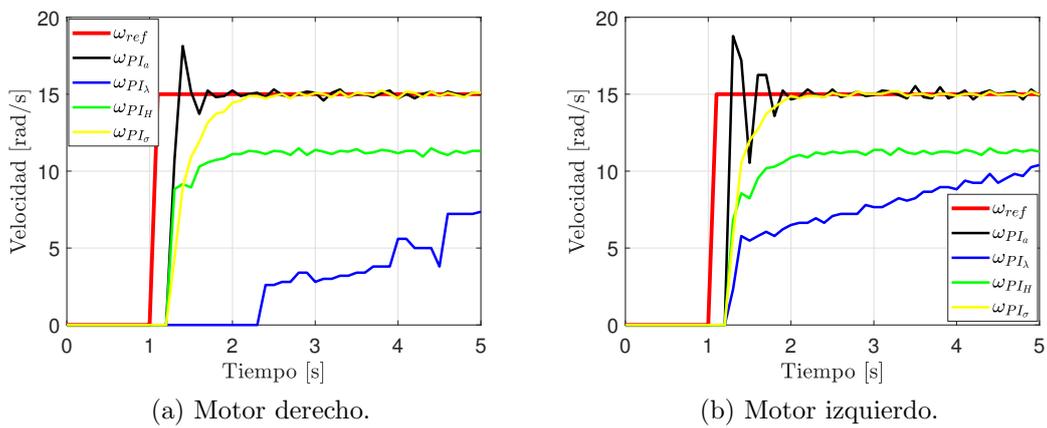


Figura 3.20: Respuestas Experimentales del Robot No. 2 a Controladores PI.

Tomando las respuestas experimentales de los controladores PI se diseñó la Tabla 3.12 donde se muestran los parámetros de rendimiento de cada una de las respuestas de manera numérica y experimental del motor derecho del Robot No. 1. Es evidente ver que la tendencia de las respuestas de los 2 motores de cada uno de los robots móviles es la misma, por lo que solo se muestran los resultados para el motor derecho del Robot 1 (G_{R1}).

Tabla 3.12: Parámetros de Rendimiento de los Controladores PI para G_{R1} .

Controlador	Simulación		Experimento			
	t_s [s]	M_p [%]	t_s [s]	M_p [%]	ISE ^a	ITAE ^b
PI σ	0.72	0	0.86	0	59.43	7.85
PIa	>2.5	93.33	0.75	30.33	46.99	6.86
PI λ	1.74	0	>4	0	501.91	113.52
PIH	0.70	16.66	>4	0	113.76	49.61

^aIntegral of Squared Error, ^b Integral of Time multiply Absolute Error

Con la finalidad de tener un criterio más objetivo para la elección de los controladores PI para los motores, se consideraron los criterios de desempeño ISE e ITAE. Adicionalmente, también se consideró el tiempo de asentamiento (t_s) con banda del 5% y el máximo sobre impulso (M_p), éstas dos últimas métricas son importantes dado que se buscaba que el robot siguiera lo más rápido posible a la señal de referencia y el sobreimpulso no es deseable en esta clase de robots, ya que el sobre impulso añade vibraciones en el movimiento del robot móvil.

Los resultados de este análisis se resumen en la Tabla 3.12, en donde se concluyó que el controlador más rápido y con ausencia de sobretiro en el seguimiento de la velocidad es el controlador PI σ .

3.4. Modelo Cinemático del Robot Móvil

Las tareas a ser ejecutadas por un robot móvil requieren cambios diferenciales en la localización (posición y orientación), es decir, en las coordenadas absolutas. Por tanto, es necesario encontrar los correspondientes cambios diferenciales en las coordenadas generalizadas y esto se logra mediante el cálculo de la matriz jacobiana y su inversa.

3.4.1. Restricciones Cinemáticas

Los vehículos de ruedas siempre suelen estar sujetos a una restricción. Por ejemplo, un coche de 4 ruedas en configuración Ackerman puede alcanzar cualquier configuración final en su plano, pero nunca podrá moverse de lado. Por lo tanto, dependiendo de la configuración final (uniciclo, bicicleta, triciclo, estructura Ackerman, etc), requiere realizar una serie de maniobras (como el

estacionamiento en paralelo) para alcanzar el estado deseado [54, 55].

Primero, hay que definir los sistemas holonómicos y no holonómicos. Consideremos un sistema mecánico con *coordenadas generalizadas* $\mathbf{q} \in C$, donde C es el espacio de configuración del sistema propuesto y coincide con \mathbb{R}^n . Para tal sistema, una restricción se denomina *cinemática* cuando sólo implica coordenadas generalizadas (\mathbf{q}) y velocidades ($\dot{\mathbf{q}}$).

Las restricciones cinemáticas se definen generalmente en *forma de Pfaffian* [34, 56]:

$$v_i^T(\mathbf{q}) \dot{\mathbf{q}} = 0 \quad i = 1, \dots, k < n, \quad (3.7)$$

donde los v_i son k vectores linealmente independientes.

La estructura cinemática de un robot móvil terrestre con ruedas RMR se puede considerar como un conjunto de cadenas cinemáticas cerradas, cuyo número esta dado por la cantidad de ruedas que estén en contacto con el suelo [56].

Normalmente se consideran las siguientes limitaciones para la construcción del modelo cinemático de un robot móvil diferencial [57]:

- El robot se mueve sobre una superficie plana.
- No existen elementos flexibles en la estructura del robot (incluidas las ruedas).
- Se considera que las ruedas poseen un eje de direccionamiento que siempre es perpendicular al suelo.
- No hay ningún deslizamiento entre el punto de contacto rueda-suelo.
- Se desprecia todo tipo de fricción en los elementos móviles del vehículo contra el suelo.

Una característica especial de los robots móviles es su naturaleza no holonómica [54], en contraste con los robots manipuladores. Las restricciones holonómicas se definen como un sistema de ecuaciones algebraicas con las variables de posición (grados de libertad GDL), tanto traslacionales como rotacionales, hacen que el sistema sea integrable sin involucrar ninguna variable de velocidad.

También se puede constituir un sistema de ecuaciones con variables de velocidad que logren integrarse, así de conducir el sistema de restricciones a uno con variables de posición.

Si el sistema de restricciones de velocidad no es integrable, se dice que las restricciones son *no holonómicas*. Por tanto, si un sistema mecánico está sujeto sólo a restricciones holonómicas, se dice que el sistema es holonómico; de lo contrario es no holonómico. Los manipuladores compuestos de pares prismáticos y de revolución son ejemplos de sistemas holonómicos, mientras que los robots móviles normalmente constituyen sistemas no holonómicos [54], [55].

La primera restricción obliga a que la rueda solo debe rodar cuando el movimiento tiene lugar en la dirección paralela al plano de la rueda. La Figura 3.21, representa el esquema de las ruedas fijas y el chasis, dicha figura indica su posición dentro del sistema de coordenadas global $\mathcal{G} = [X_G \ Y_G \ Z_G]^T$. El ángulo que el plano de la rueda forma con el chasis es constante, porque la rueda está fijada al chasis y no tiene desplazamientos laterales [58].

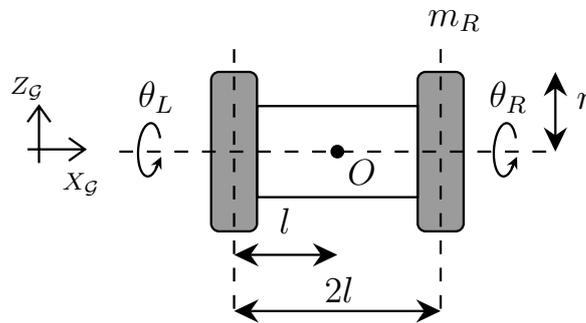


Figura 3.21: Esquema del Robot Móvil en Vista del Plano $\{X, Z\}_G$.

La rueda tiene un radio r y puede rotar con un ángulo θ sobre su eje horizontal (véase Figura 3.21), por lo que su posición rotacional es una función de tiempo $\theta(t)$. Cada rueda posee una velocidad angular $\dot{\theta}(t) = \omega(t)$. La distancia entre las dos ruedas de manera perpendicular está dada por el valor de $2l$.

Esta restricción rodante fuerza a que la cantidad de movimiento a lo largo de la dirección del plano de la rueda izquierda o derecha sea igual, al giro de la rueda sobre su eje horizontal, con el fin de obtener un puro rodamiento en el punto del contacto sin deslizamiento lateral.

La segunda restricción plantea que no haya ningún deslizamiento lateral, esto es, que la rueda no debe resbalar ortogonal al plano de la rueda [58].

Por lo tanto las restricciones no holonómicas del robot móvil (véase la Figura 3.22) se definen como [59]:

$$\begin{aligned} -\dot{x} \sin(\varphi) + \dot{y} \cos(\varphi) &= 0, \\ \dot{x} \cos(\varphi) + \dot{y} \sin(\varphi) &= -\frac{r(\omega_R + \omega_L)}{2}. \end{aligned} \quad (3.8)$$

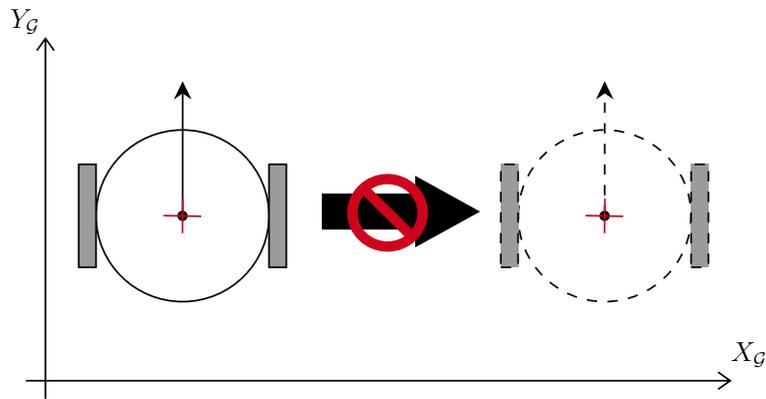


Figura 3.22: Restricciones Cinemáticas del Robot Diferencial.

3.4.2. Cinemática del Punto Central del Robot

Se tiene un robot móvil en configuración diferencial (véase la Figura 3.23), el cual posee un eje de coordenadas global $\mathcal{G} = [X_G \ Y_G]^T$. El carro está constituido por un chasis, un par de ruedas rígidas y una rueda libre. En el enfoque cinemático se omite el valor de la masa del robot. El modelo cinemático calculará la posición y orientación del punto O en el eje de coordenadas global.

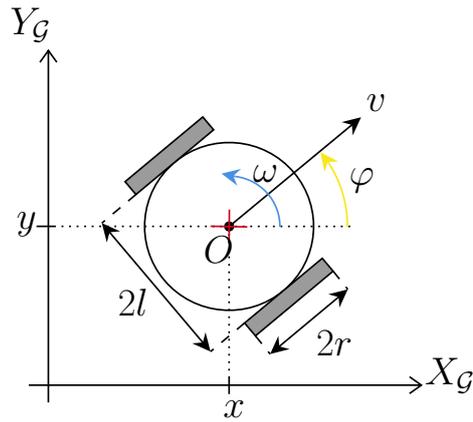


Figura 3.23: Esquema del Robot Móvil Diferencial para el Modelo No Holonómico.

En la Figura 3.23, x es la posición en el eje X_G , y es la posición en el eje Y_G , φ es la orientación del robot con respecto al eje X_G , ω_L es la velocidad angular de la rueda izquierda, ω_R es la velocidad angular de la rueda derecha, r el radio de las ruedas y $2l$ la distancia perpendicular entre las dos ruedas.

Para obtener el modelo cinemático primero se calcularon las velocidades lineales de las ruedas:

$$\begin{aligned} v_R &= r\omega_R, \\ v_L &= r\omega_L. \end{aligned} \tag{3.9}$$

Con las velocidades lineales se calculó la velocidad lineal instantánea total del robot:

$$v = \frac{v_R + v_L}{2}.$$

Posteriormente se calcularon las componentes de la velocidad lineal del robot:

$$\begin{aligned} v_x &= v \cos(\varphi), \\ v_y &= v \sen(\varphi). \end{aligned} \tag{3.10}$$

Se calculó la nueva posición para un t pequeño como:

$$\begin{aligned} x' &= x + v_x t, \\ y' &= y + v_y t. \end{aligned}$$

Posteriormente se calculó la velocidad angular del vehículo como:

$$\omega = \frac{v_R - v_L}{2l}.$$

Se calculó la nueva orientación del robot mediante:

$$\varphi' = \varphi + \omega t.$$

Tomando las ecuaciones anteriores entonces se obtuvo el modelo:

$$\begin{aligned} x' &= x + v \cdot \cos(\varphi) t, \\ y' &= y + v \cdot \operatorname{sen}(\varphi) t, \\ \varphi' &= \varphi + \omega t. \end{aligned} \quad (3.11)$$

Reescribiendo el sistema se obtuvo el modelo cinemático del robot diferencial:

$$\begin{aligned} \dot{x} &= v \cos(\varphi), \\ \dot{y} &= v \operatorname{sen}(\varphi), \\ \dot{\varphi} &= \omega. \end{aligned} \quad (3.12)$$

Definiendo un vector $\mathbf{p} \in \mathbb{R}^n$ como el vector de localización de $n = 3$ elementos (dos de posición y uno de orientación), que representa un punto en el espacio de las coordenadas globales se obtuvo:

$$\dot{\mathbf{p}} := \begin{bmatrix} \dot{x} & \dot{y} & \dot{\varphi} \end{bmatrix}^T, \quad \dot{\mathbf{q}} := \begin{bmatrix} v & \omega \end{bmatrix}^T. \quad (3.13)$$

Siendo $\mathbf{q} \in \mathbb{R}^m$ el correspondiente vector de $m = 2$ variables (velocidades lineales y angular) en el espacio de las coordenadas generalizadas del robot, siendo $n > m$ además, $\dot{\mathbf{p}}$ y $\dot{\mathbf{q}}$ son las derivadas temporales correspondientes. En lo posterior, se consideró que las variables se expresan en el sistema de las coordenadas globales.

El modelo diferencial directo en forma matricial es:

$$\dot{\mathbf{p}} = \mathbf{J}(\mathbf{q}) \dot{\mathbf{q}}. \quad (3.14)$$

Por lo tanto la cinemática directa se definió como:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\varphi} \end{bmatrix} = \begin{bmatrix} \cos(\varphi) & 0 \\ \sin(\varphi) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix}. \quad (3.15)$$

Estados: x, y, φ .

Entradas: v, ω .

Este modelo es de tipo *no holonómico*.

Se realizaron varias simulaciones para corroborar que el modelo cinemático diferencial del punto central sea correcto. En la Figura 3.24 se muestra una simulación con condiciones iniciales iguales a cero, con velocidades de referencia $v_{ref} = 0.5 \sin(0.1t)$ m/s y $\omega_{ref} = 0.8$ rad/s.

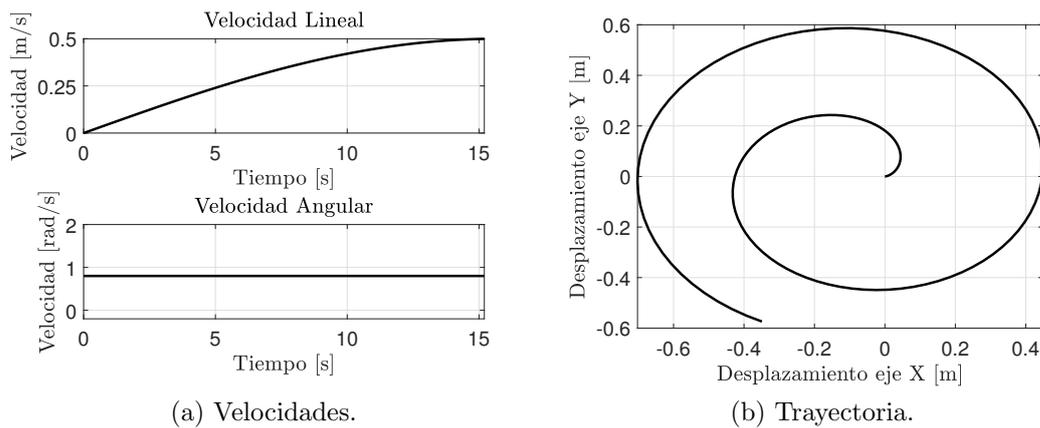


Figura 3.24: Respuesta Numérica de la CPC.

3.4.3. Cinemática de un Punto Desplazado del Robot

La cinemática de un punto desplazado (punto P) se estableció a partir de la posición y velocidad de un punto que está separado una distancia a (véase la Figura 3.25) del eje de tracción del robot (punto O).

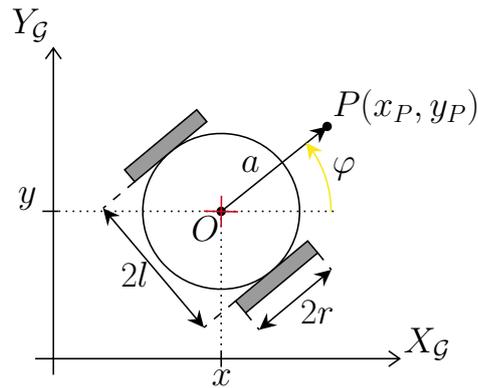


Figura 3.25: Esquema del Robot Móvil Diferencial para el Modelo Holonómico.

De esta forma las coordenadas del punto desplazado del robot móvil se definieron por:

$$\begin{bmatrix} \dot{x}_P \\ \dot{y}_P \\ \dot{\varphi} \end{bmatrix} = \begin{bmatrix} \cos(\varphi) & -a \sin(\varphi) \\ \sin(\varphi) & a \cos(\varphi) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix}. \quad (3.16)$$

Estados: x, y, φ .

Entradas: v, ω .

Este modelado es de tipo *holonómico*, por lo tanto el punto $P(x_P, y_P)$ no tiene restricciones de velocidad en el plano $\mathcal{G} = [X_G, Y_G]^T$. Este modelo es el que se utilizó para el diseño de los controladores.

Se realizaron una serie simulaciones para corroborar que el modelo cinemático diferencial del punto descentralizado sea correcto. En la Figura 3.26 se muestra una simulación con condiciones iniciales iguales a cero, con velocidades de referencia $v_{ref} = 0.5 \sin(0.1t)$ m/s y $\omega_{ref} = 0.8$ rad/s.

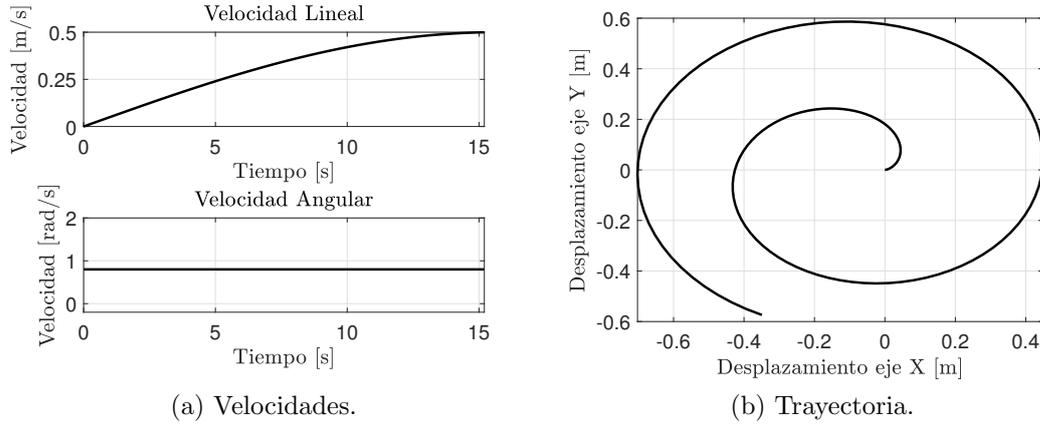


Figura 3.26: Respuesta Numérica de la CPD.

3.5. Control en Lazo Abierto

Las ecuaciones de movimiento que definen al sistema (robot en configuración diferencial) (3.15) en donde las velocidades lineal v y angular ω son las variables manipuladas, es decir, son las variables que se modifican con el objetivo de modificar las variables controladas, las cuales son las que se miden y se controlan y son representadas por:

$$\mathbf{p} = \begin{bmatrix} x & y & \varphi \end{bmatrix}^T. \quad (3.17)$$

Para una trayectoria deseada o de referencia $p_d(x_d(t), y_d(t))$ definida en un intervalo $t \in [0, T]$, se puede diseñar una ley de control en lazo abierto. Para obtener dicha ley, se debe derivar la expresión (3.15) con respecto al tiempo, donde se obtiene un vector de velocidad en x y uno de velocidad en y , para cada punto de la trayectoria deseada. Por lo tanto, la velocidad deseada del robot para cada punto de la trayectoria es obtenida al calcular el módulo de la suma de ambos vectores:

$$v_d = \pm \sqrt{\dot{x}_d^2 + \dot{y}_d^2}. \quad (3.18)$$

El ángulo de la tangente a cada punto de la trayectoria se define como:

$$\varphi_d = \arctan \left(\frac{\dot{y}_d}{\dot{x}_d} \right). \quad (3.19)$$

Por último, para obtener la velocidad angular, se deriva la posición angular respecto al tiempo para cada punto de la trayectoria:

$$\omega_d = \dot{\varphi}_d = \frac{\dot{x}_d \ddot{y}_d - \dot{y}_d \ddot{x}_d}{\dot{x}_d^2 + \dot{y}_d^2}. \quad (3.20)$$

Con las expresiones (3.18), (3.19) y (3.20), se define la trayectoria de referencia $\mathbf{p}_d = [x_d \ y_d \ \varphi_d]^T$ y las entradas al robot virtual v_d y ω_d . Para esta ley de control es requisito que la trayectoria sea de tipo \mathcal{C}^2 , es decir, que sus segundas derivadas sean continuas, también el error inicial debe ser cero y no deben existir perturbaciones.

3.6. Control en Lazo Cerrado

Utilizando la cinemática del punto descentralizado del robot se asignó una trayectoria de movimiento $\mathbf{p}_{xy} = [x \ y]^T$ mostrado en (3.16) al punto de interés en términos de $\dot{\mathbf{p}}_{xy}$ y sus condiciones iniciales de posición (x,y).

El objetivo es determinar un conjunto de velocidades $\dot{\mathbf{q}} = [v \ \omega]^T$ que reproduzca la trayectoria deseada, por lo que la cinemática directa se definió como:

$$\dot{\mathbf{p}}_{xy} = \mathbf{J}_{xy}(\mathbf{q}) \dot{\mathbf{q}}. \quad (3.21)$$

con

$$\mathbf{J}_{xy}(\mathbf{q}) := \begin{bmatrix} \cos(\varphi) & -a \sin(\varphi) \\ \sin(\varphi) & a \cos(\varphi) \end{bmatrix} \quad (3.22)$$

Considerando que se tienen el mismo número de ecuaciones e incógnitas, las velocidades de los actuadores se obtuvieron mediante la simple inversión de la matriz Jacobiana:

$$\dot{\mathbf{q}} = \mathbf{J}_{xy}^{-1}(\mathbf{q}) \dot{\mathbf{p}}_{xy}. \quad (3.23)$$

Dado que \mathbf{J}_{xy} es no singular siempre que $a \neq 0$ esto implica que la inversa de \mathbf{J} siempre existirá asumiendo la misma condición, ya que $\det(\mathbf{J}) = a$. A partir de ahora se omitió el argumento \mathbf{q} de \mathbf{J}_{xy} .

El esquema de solución para el Control en Lazo Cerrado (CLC) se resolvió mediante el cálculo del error de posición y orientación del punto de interés P deseado y el actual.

$$\mathbf{e} = \mathbf{p}_{xy} - \mathbf{p}_{xyd}. \quad (3.24)$$

La dinámica del error se definió como:

$$\dot{\mathbf{e}} = \dot{\mathbf{p}}_{xy} - \dot{\mathbf{p}}_{xyd}. \quad (3.25)$$

Realizando un despeje sencillo $\dot{\mathbf{p}}_{xy} = \dot{\mathbf{p}}_{xyd} + \dot{\mathbf{e}}$, y sustituyendo en (3.23) se obtuvo:

$$\dot{\mathbf{q}} = \mathbf{J}_{xy}^{-1} (\dot{\mathbf{p}}_{xyd} + \dot{\mathbf{e}}). \quad (3.26)$$

Teniendo lo anterior, se utilizó la Teoría de Lyapunov para asegurar que este controlador sea asintóticamente estable. Por lo tanto, los términos cambiarán según el objetivo/tarea: Control de Posición sin Orientación (CPO), Control de Posición con Ganancia Adaptativa (CPOGA), Control de Seguimiento de Trayectoria (CST) o Control de Seguimiento de Caminos (CSC).

3.6.1. Control de Posición sin Orientación

Teorema 3.6.1: Control de Posición sin Orientación (CPO)

Sea la ley de control

$$\dot{\mathbf{q}}_{ref} = \mathbf{J}_{xy}^{-1} (-\mathbf{K}\mathbf{e}), \quad (3.27)$$

donde $\dot{\mathbf{q}}_{ref} = \begin{bmatrix} v_{ref} & \omega_{ref} \end{bmatrix}^T$.

El sistema (3.21) es AE si $\mathbf{K} \in \mathbb{R}^{2 \times 2}$ es una matriz diagonal definida positiva.

Prueba. Partiendo de la la expresión general (3.26) para el movimiento del robot diferencial, en este caso \mathbf{p}_{xyd} es constante y por lo tanto $\dot{\mathbf{p}}_{xyd} = 0$, ya que solo se desea que el robot se mueva de una posición inicial a una posición final (véase la Figura 3.27), entonces:

$$\dot{\mathbf{q}} = \mathbf{J}_{xy}^{-1} \left(\dot{\mathbf{p}}_{xyd} + \dot{\mathbf{e}} \right).$$

Por lo tanto se obtiene:

$$\dot{\mathbf{q}} = \mathbf{J}_{xy}^{-1} (\dot{\mathbf{e}}). \quad (3.28)$$

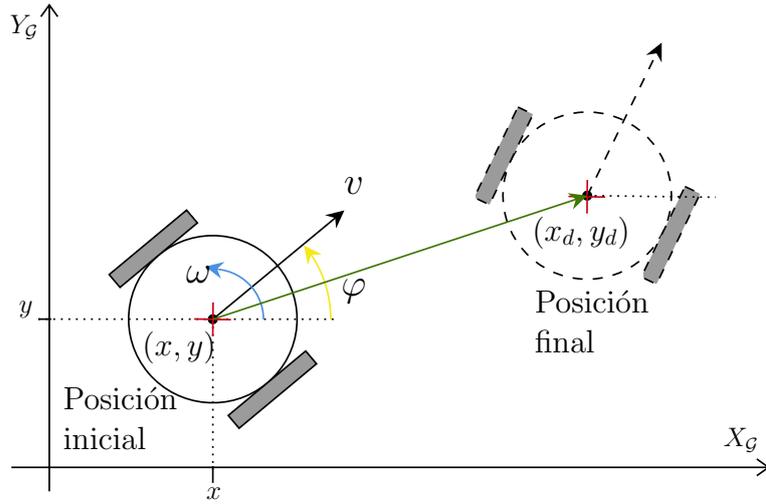


Figura 3.27: Esquema para la Posición del Robot Diferencial.

Teniendo esto se hizo el análisis de estabilidad por la teoría de Lyapunov. Para el análisis de estabilidad se asumió un seguimiento perfecto de la velocidad, esto quiere decir que las velocidades de referencia (las enviadas por el controlador) son iguales a las velocidades actuales del robot:

$$\dot{\mathbf{q}}_{ref} = \dot{\mathbf{q}}. \quad (3.29)$$

Posteriormente se propuso la función candidata de Lyapunov:

$$\mathcal{V}(\mathbf{e}) = \frac{\mathbf{e}^T \mathbf{e}}{2}. \quad (3.30)$$

Derivando el sistema se obtuvo:

$$\dot{\mathcal{V}} = \mathbf{e}^T \dot{\mathbf{e}}. \quad (3.31)$$

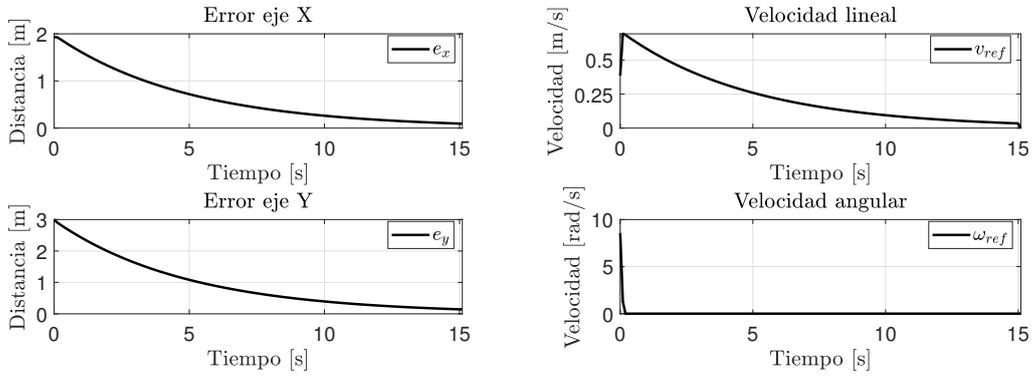
Para asegurar que el sistema sea AE, la expresión anterior debe ser definida negativa. Para esto se realizó un cambio de variable para el valor de la dinámica del error $\dot{\mathbf{e}} = -\mathbf{K}\mathbf{e}$ por lo que se obtuvo una función definida negativa:

$$\dot{\mathcal{V}} = -\mathbf{e}^T \mathbf{K} \mathbf{e} < 0, \quad (3.32)$$

Por lo tanto el sistema (3.21) es AE, si \mathbf{K} es una matriz diagonal definida positiva. ■

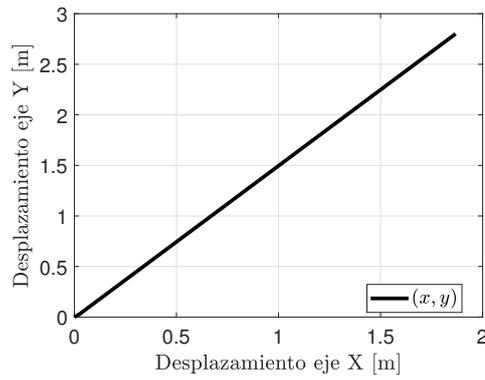
Después del análisis teórico se validó numéricamente la estabilidad de los controladores por medio de simulaciones a diferentes condiciones iniciales y posiciones finales.

1. El robot tenía condiciones iniciales iguales a cero y se deseó llevar a las coordenadas $x_d = 2$, $y_d = 3$, con una matriz de ganancias $\mathbf{K} = \text{diag}(0.2, 0.2)$.



(a) Errores de posición.

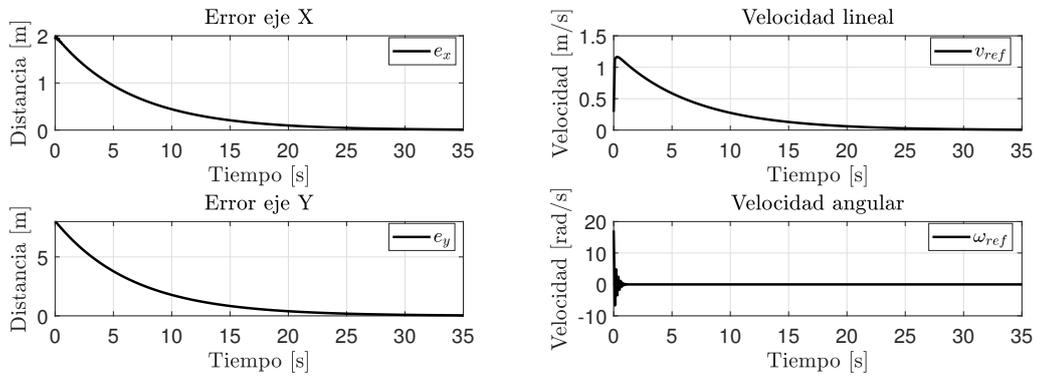
(b) Velocidades.



(c) Trayectoria.

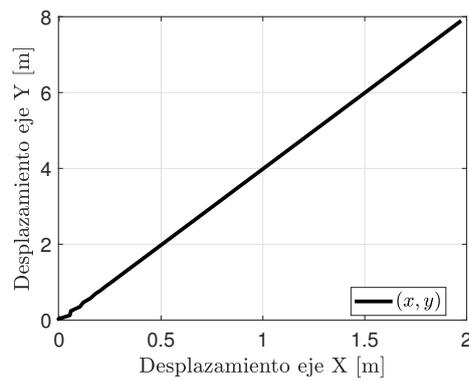
Figura 3.28: Respuesta Numérica No. 1 del CPO.

2. El robot estuvo en condiciones iniciales iguales a cero y se deseó llevar a las coordenadas $x_d = 2$, $y_d = 8$, con una matriz de ganancias $\mathbf{K} = \text{diag}(0.15, 0.15)$.



(a) Errores de posición.

(b) Velocidades.



(c) Trayectoria.

Figura 3.29: Respuesta Numérica No. 2 del CPO.

3. El robot tenía condiciones iniciales iguales a cero y se deseó llevar a las coordenadas $x_d = -2$, $y_d = 10$, con una matriz de ganancias $\mathbf{K} = \text{diag}(0.18, 0.18)$.

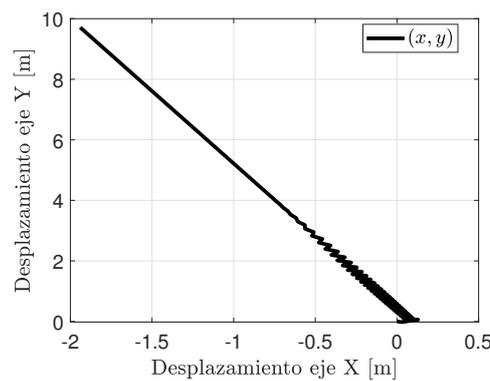
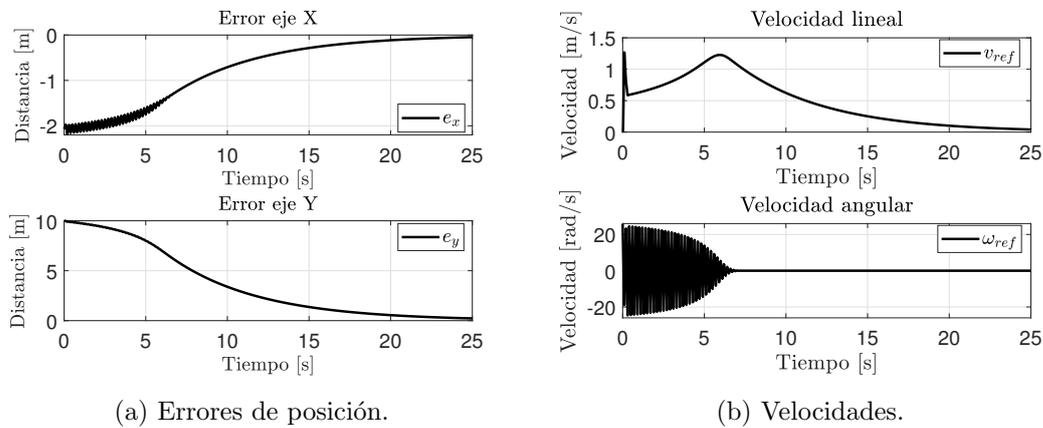


Figura 3.30: Respuesta Numérica No. 3 del CPO.

3.6.2. Control de Posición sin Orientación con Ganancia Adaptativa

Con resultados obtenidos en la sección anterior, se pudo ver que al inicio del movimiento del robot se dieron respuestas en la velocidad con valores altas mientras que el error iba convergiendo a cero las velocidades también se iban haciendo cero. Dichos rangos de velocidades altas y bajas el robot no las podría realizar en la validación experimental, por lo que se propuso la siguiente estrategia de control.

Teorema 3.6.2: Control de Posición sin Orientación con Ganancia Adaptativa (CPOGA)

Sea la ley de control

$$\dot{\mathbf{q}}_{ref} = \mathbf{J}_{xy}^{-1} (-\mathbf{K}\mathbf{e}), \quad (3.33)$$

donde $\dot{\mathbf{q}}_{ref} = \begin{bmatrix} v_{ref} & \omega_{ref} \end{bmatrix}^T$.

El sistema (3.21) es AE si $\mathbf{K} \in \mathbb{R}^{2 \times 2}$ está dada por:

$$\mathbf{K} = \begin{bmatrix} \frac{k_{max}}{1 + k_v d} & 0 \\ 0 & \frac{k_{max}}{1 + k_v d} \end{bmatrix}, \quad (3.34)$$

con $d > 0$, $k_{max} > 0$ y $k_v > 0$.

Prueba. La distancia entre posición inicial y final, se conoce como distancia euclidiana y se define como:

$$d = \sqrt{(x_d - x)^2 + (y_d - y)^2}, \quad (3.35)$$

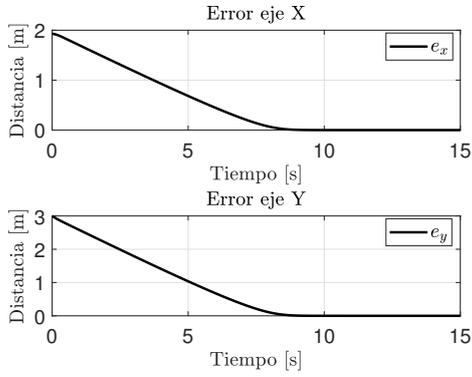
la cual por definición siempre es positiva.

En la definición de \mathbf{K} las constantes $k_{max} > 0$ es el valor máximo al que puede llegar esa ganancia y $k_v > 0$ es un factor que determina la velocidad con la que el error converge a cero.

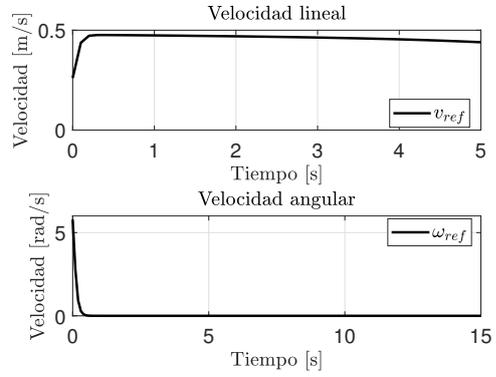
La demostración de la estabilidad del sistema se hace de la misma manera que el Teorema 3.6.1, y se concluye que el sistema (3.21) es AE si $k_{max} > 0$ y $k_v > 0$. ■

Se realizaron simulaciones con las condiciones iniciales mostradas en la Subsección 3.6.1, se pudo ver que los esfuerzos de control se redujeron considerablemente, además se muestra como se comportó el valor de las ganancias con respecto de la distancia euclidiana entre la posición inicial y la posición final (véase la Figura 3.27).

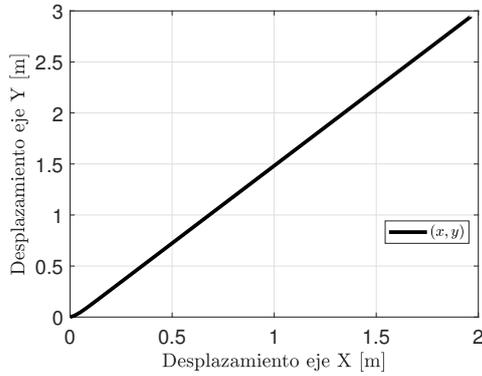
1. Condiciones iniciales iguales a cero y se deseó llevar a las coordenadas $x_d = 2$, $y_d = 3$, con $k_{max} = 10$ y $k_v = 5$.



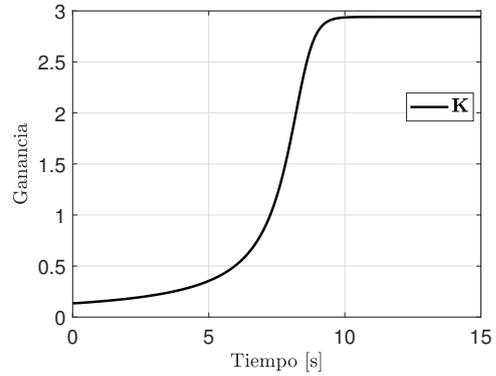
(a) Errores de posición.



(b) Velocidades.



(c) Trayectoria.



(d) Ganancia.

Figura 3.31: Respuesta Numérica No. 1 del CPOGA.

2. Condiciones iniciales iguales a cero y se deseó llevar a las coordenadas $x_d = 2$, $y_d = 8$, con $k_{max} = 10$ y $k_v = 5$.

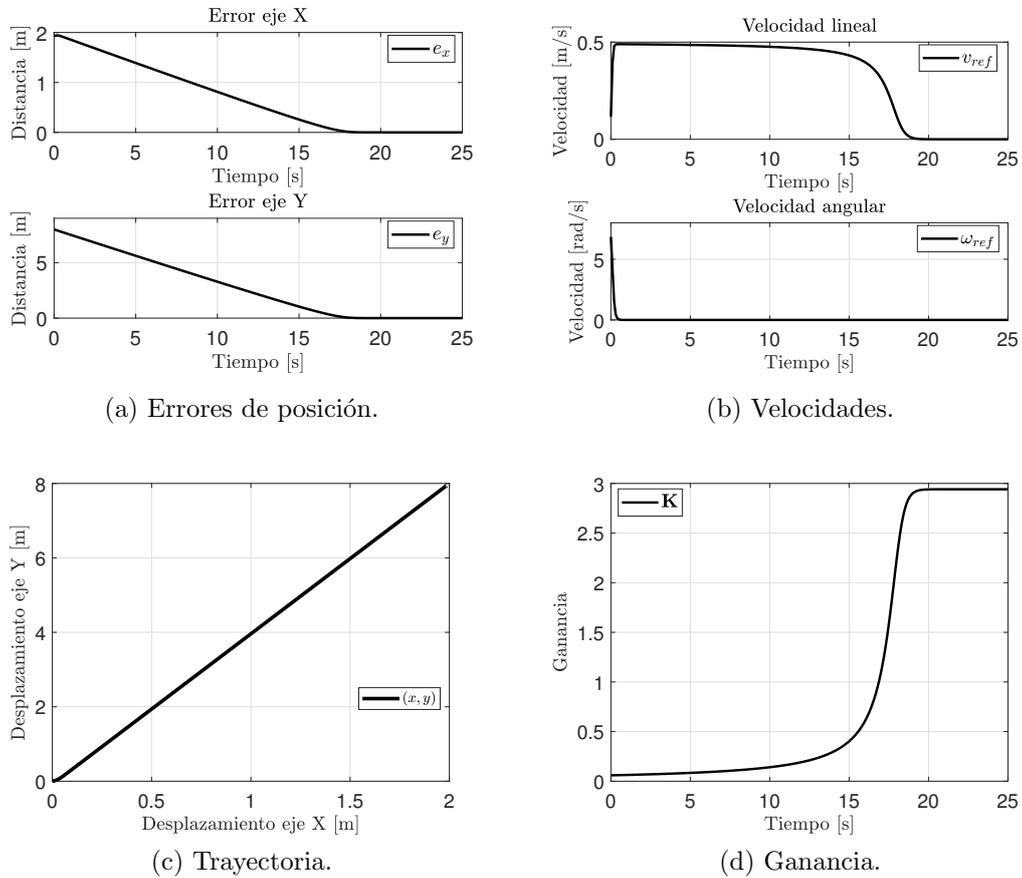


Figura 3.32: Respuesta Numérica No. 2 del CPOGA.

3. Condiciones iniciales iguales a cero y se deseó llevar a las coordenadas $x_d = -2$, $y_d = 10$, con $k_{max} = 10$ y $k_v = 5$.

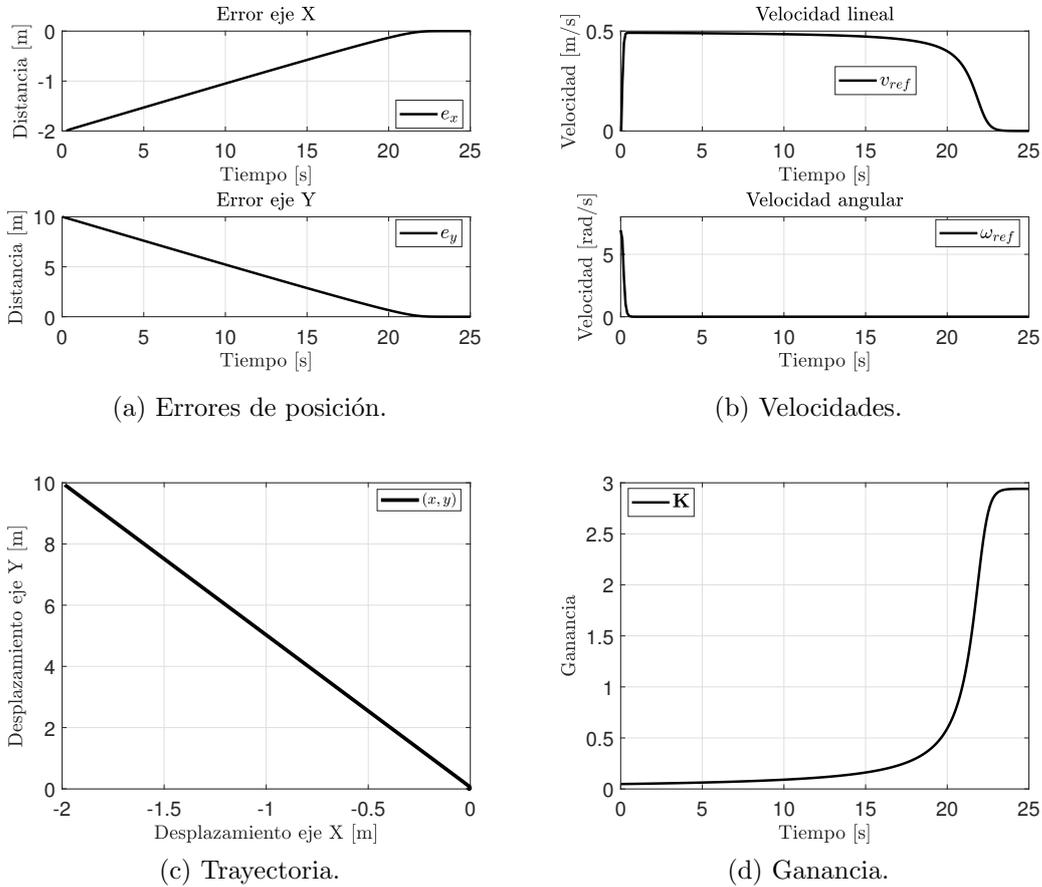


Figura 3.33: Respuesta Numérica No. 3 del CPOGA.

3.6.3. Control de Seguimiento de Trayectoria

La tarea de control de seguimiento de trayectorias en robots móviles consiste en hacer que el robot siga como referencia una curva parametrizada en el tiempo con errores iguales a cero. El objetivo de este tipo de controlador, es conseguir que mediante una ley de control que el robot alcance y siga con error cero estados deseados variantes en el tiempo, dichos estados describen la trayectoria deseada a seguir.

Anteriormente se definió el vector de estados y el vector de entradas, por lo que si se tiene el mismo número de ecuaciones e incógnitas, las velocidades del robot se expresa como en (3.21).

Tomando la expresión (3.28) del control de posición sin orientación final, se consideró que la velocidad $\dot{\mathbf{p}}_{xy_d} = 0$, pero en este caso actual $\dot{\mathbf{p}}_{xy_d} \neq 0$, por lo tanto se obtiene:

$$\dot{\mathbf{q}} = \mathbf{J}_{xy}^{-1} (\dot{\mathbf{p}}_{xy_d} + \dot{\mathbf{e}}). \quad (3.36)$$

Teorema 3.6.3: Control de Seguimiento de Trayectoria (CST)

Sea la ley de control

$$\dot{\mathbf{q}}_{ref} = \mathbf{J}_{xy}^{-1} (\dot{\mathbf{p}}_{xy_d} - \mathbf{K}\mathbf{e}), \quad (3.37)$$

donde $\dot{\mathbf{q}}_{ref} := \begin{bmatrix} v_{ref} & \omega_{ref} \end{bmatrix}^T$.

El sistema (3.21) es AE si $\mathbf{K} \in \mathbb{R}^{2 \times 2}$ es una matriz diagonal definida positiva.

Prueba. Primeramente para el análisis de estabilidad se asumió que existe el seguimiento perfecto de la velocidad ($\dot{\mathbf{q}}_{ref} = \dot{\mathbf{q}}$).

Se eligió la siguiente función candidata de Lyapunov:

$$\mathcal{V}(\mathbf{e}) = \frac{\mathbf{e}^T \mathbf{e}}{2}. \quad (3.38)$$

Derivando el sistema se obtuvo:

$$\dot{\mathcal{V}} = \mathbf{e}^T \dot{\mathbf{e}}. \quad (3.39)$$

Para determinar que la función anterior sea definida negativa, se sustituyó la ley de control (3.37) en el modelo matemático del robot (3.21) y se obtuvo:

$$\dot{\mathbf{p}}_{xy} = \mathbf{J}_{xy} \mathbf{J}_{xy}^{-1} (\dot{\mathbf{p}}_{xy_d} - \mathbf{K}\mathbf{e}). \quad (3.40)$$

La dinámica del error se definió como $\dot{\mathbf{e}} = \dot{\mathbf{p}}_{xy} - \dot{\mathbf{p}}_{xy_d}$, tomando en cuenta el seguimiento perfecto de velocidad lo anterior se sustituyó en (3.40) se obtuvo:

$$\begin{aligned} \dot{\mathbf{p}}_{xy_d} + \dot{\mathbf{e}} &= (\dot{\mathbf{p}}_{xy_d} + \mathbf{K}\mathbf{e}), \\ \dot{\mathbf{e}} &= -\mathbf{K}\mathbf{e}. \end{aligned}$$

Se obtuvo una función definida negativa:

$$\dot{\mathbf{y}} = -\mathbf{e}^T \mathbf{K} \mathbf{e} < 0. \tag{3.41}$$

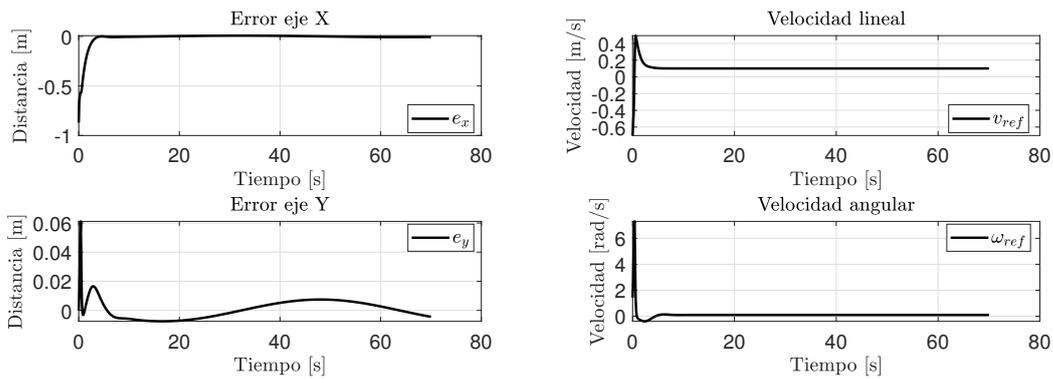
Por lo tanto el sistema (3.21) es AE, ssi \mathbf{K} es una matriz diagonal definida positiva. ■

Se realizaron simulaciones para diferentes trayectorias:

1. Condiciones iniciales $x_0 = 1.8$, $y_0 = 0$ y $\varphi_0 = 0^\circ$, y se deseó seguir la trayectoria de un círculo parametrizada en el tiempo:

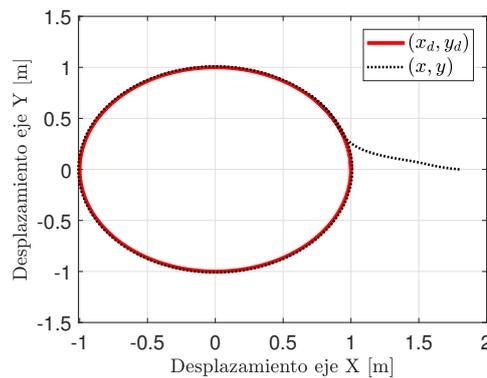
$$\begin{aligned} x_d(t) &= \cos(0.1t), \\ y_d(t) &= \text{sen}(0.1t), \end{aligned}$$

con una matriz de ganancias $\mathbf{K} = \text{diag}(1, 1)$.



(a) Errores de posición.

(b) Velocidades.



(c) Trayectoria.

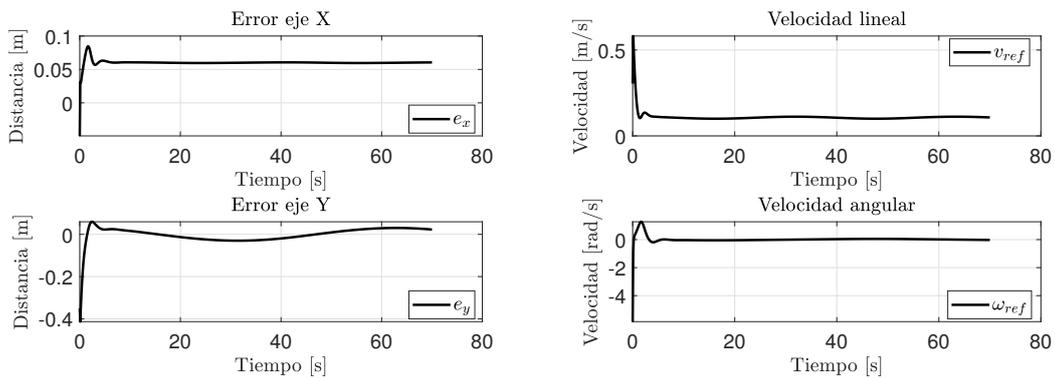
Figura 3.34: Respuesta Numérica No. 1 del CST.

2. Condiciones iniciales $x_0 = 0$, $y_0 = 0.4$ y $\varphi_0 = -45^\circ$, y se deseó seguir la trayectoria descrita por una función seno, la cual esta parametrizada en el tiempo:

$$x_d(t) = 0.1t,$$

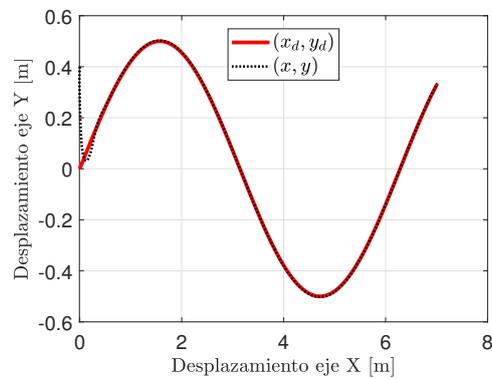
$$y_d(t) = 0.5 \text{ sen}(0.1t),$$

con una matriz de ganancias $\mathbf{K} = \text{diag}(1.5, 1.5)$.



(a) Errores de posición.

(b) Velocidades.



(c) Trayectoria.

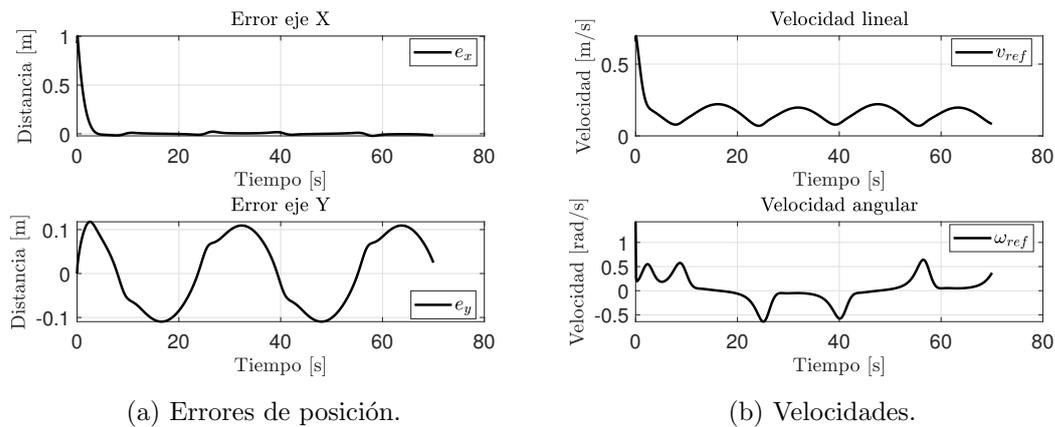
Figura 3.35: Respuesta Numérica No. 2 del CST.

3. Condiciones iniciales iguales a cero y se deseó seguir una trayectoria denominada como lemniscata de Bernoulli, la cual está parametrizada por:

$$x_d(t) = \cos(0.1t),$$

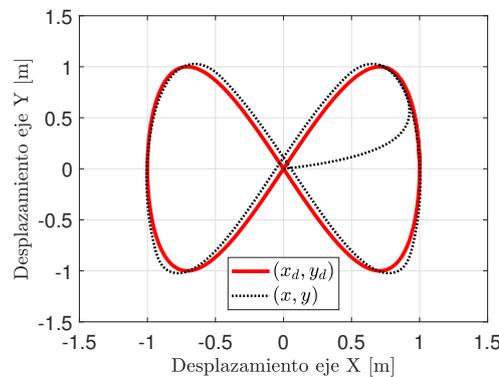
$$y_d(t) = \text{sen}(0.2t),$$

con una matriz de ganancias $\mathbf{K} = \text{diag}(0.9, 0.9)$.



(a) Errores de posición.

(b) Velocidades.



(c) Trayectoria.

Figura 3.36: Respuesta Numérica No. 3 del CST.

3.6.4. Control de Seguimiento de Caminos

A diferencia del control de seguimiento de trayectorias, el control de seguimiento de caminos implica que el robot siga el camino deseado sin ninguna especificación temporal (ecuación paramétrica). Un camino es una sucesión de líneas rectas que unen puntos ubicados en el espacio (plano \mathcal{G} en este caso). Un camino también se puede obtener de la discretización de una curva

continua (ecuación paramétrica) que interpola/une puntos de la misma.

Se utilizó la Teoría de Lyapunov al igual que en el diseño de los controladores anteriores. La ley de control que se propuso para esta tarea de control es la misma que se utilizó para el seguimiento de trayectorias, la cual está expresada en (3.37) excepción de un término ($\dot{\mathbf{p}}_r$).

Teorema 3.6.4: Control de Seguimiento de Caminos (CSC)

Sea la ley de control

$$\dot{\mathbf{q}}_{ref} = \mathbf{J}_{xy}^{-1} (\dot{\mathbf{p}}_r - \mathbf{K}\mathbf{e}), \quad (3.42)$$

donde $\dot{\mathbf{q}}_{ref} := \begin{bmatrix} v_{ref} & \omega_{ref} \end{bmatrix}^T$.

El sistema (3.21) es AE si:

(i) \mathbf{K} es una matriz diagonal definida positiva.

(ii) $\mathbf{K} > \frac{|\dot{\boldsymbol{\delta}}|}{|\mathbf{e}|}$, $\dot{\boldsymbol{\delta}} := \dot{\mathbf{p}}_d - \dot{\mathbf{p}}_r$.

donde $\dot{\boldsymbol{\delta}}$ es el error de velocidad.

Prueba. En este caso se consideró que el robot tendrá una velocidad diferente a la deseada ($\dot{\mathbf{q}}_{ref} \neq \dot{\mathbf{q}}$), por lo tanto la ley de control inicial cambió en un término, es decir, $\dot{\mathbf{p}}_r \neq \dot{\mathbf{p}}_{xy_d}$.

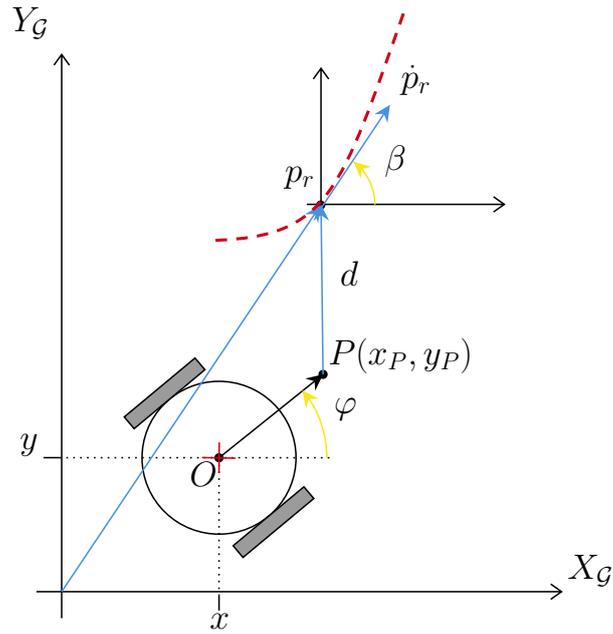


Figura 3.37: Esquema para el CSC.

En la Figura 3.37 se observa que utilizando el modelo cinemático de punto desplazado, el robot debe seguir el camino (curva roja) mediante el cálculo de la distancia euclidiana d y el ángulo β . De la Figura 3.37 se pudieron obtener las componentes de la velocidad $\dot{\mathbf{p}}_r$:

$$\begin{aligned} \dot{x}_r &= |\dot{\mathbf{p}}_r| \cos(\beta), \\ \dot{y}_r &= |\dot{\mathbf{p}}_r| \sin(\beta), \end{aligned} \quad (3.43)$$

donde:

$$\beta = \tan^{-1} \left(\frac{\Delta x_r}{\Delta y_r} \right). \quad (3.44)$$

La velocidad $\dot{\mathbf{p}}_r$ puede ser cualquier cosa, pero se le puso un límite en valor absoluto el cual se definió como:

$$|\dot{\mathbf{p}}_r| = v_{max}. \quad (3.45)$$

Para el análisis de estabilidad se consideró el seguimiento perfecto de la velocidad ($\dot{\mathbf{q}}_{ref} = \dot{\mathbf{q}}$). Se seleccionó la siguiente función candidata de Lyapunov:

$$\mathcal{V}(\mathbf{e}) = \frac{\mathbf{e}^T \mathbf{e}}{2}. \quad (3.46)$$

Derivando el sistema se obtuvo:

$$\dot{\mathcal{V}} = \mathbf{e}^T \dot{\mathbf{e}}. \quad (3.47)$$

Para determinar que la función anterior sea definida negativa, se sustituyó la ley de control (3.42) en el modelo matemático del robot (3.21) y se obtuvo:

$$\begin{aligned} \dot{\mathbf{p}}_{xy} &= \mathbf{J}_{xy} \mathbf{J}_{xy}^{-1} (\dot{\mathbf{p}}_r - \mathbf{K}\mathbf{e}), \\ \dot{\mathbf{p}}_{xy} &= \dot{\mathbf{p}}_r - \mathbf{K}\mathbf{e}. \end{aligned} \quad (3.48)$$

La dinámica del error de posición se definió como $\dot{\mathbf{e}} = \dot{\mathbf{p}}_{xy} - \dot{\mathbf{p}}_{xy_d}$ y también se definió un error de velocidad $\dot{\boldsymbol{\delta}} = \dot{\mathbf{p}}_r - \dot{\mathbf{p}}_{xy_d}$ tomando en la hipótesis del seguimiento perfecto de velocidad lo anterior se sustituyó en (3.48) y se obtuvo:

$$\dot{\mathbf{p}}_{xy_d} + \dot{\mathbf{e}} = \dot{\mathbf{p}}_r - \mathbf{K}\mathbf{e}.$$

Moviendo las velocidades del lado izquierdo de la igualdad (error de velocidad) y los errores del lado derecho se obtuvo:

$$\dot{\boldsymbol{\delta}} = \dot{\mathbf{e}} + \mathbf{K}\mathbf{e}. \quad (3.49)$$

Para poder sustituir los valores de la función candidata de Lyapunov se despejó la dinámica del error de posición de la expresión (3.49) y se obtuvo.

$$\dot{\mathbf{e}} = \dot{\boldsymbol{\delta}} - \mathbf{K}\mathbf{e}.$$

Por lo tanto la derivada temporal de la candidata de Lyapunov fue:

$$\begin{aligned} \dot{\mathcal{V}} &= \mathbf{e}^T (\dot{\boldsymbol{\delta}} - \mathbf{K}\mathbf{e}), \\ \dot{\mathcal{V}} &= \mathbf{e}^T \dot{\boldsymbol{\delta}} - \mathbf{e}^T \mathbf{K}\mathbf{e}. \end{aligned} \quad (3.50)$$

La expresión (3.50) debe ser definida negativa para que el sistema sea AE, por lo tanto se definieron las condiciones para que esto siempre se cumpla.

Si se toma el término derecho de la igualdad se pudo asegurar que $\dot{\mathcal{V}} < 0$ mientras se cumpla que:

$$\left| \mathbf{e}^T \dot{\boldsymbol{\delta}} \right| < \left| \mathbf{e}^T \mathbf{K}\mathbf{e} \right|. \quad (3.51)$$

Desarrollando la expresión anterior se demostró que (3.42) es AE si y solo si:

$$\mathbf{K} > \frac{|\dot{\delta}|}{|e|}, \tag{3.52}$$

para toda \mathbf{K} diagonal definida positiva. ■

Se realizaron simulaciones para diferentes caminos.

1. Condiciones iniciales $x_0 = 0$, $y_0 = 0$ y $\varphi_0 = -30^\circ$ y se deseó seguir el camino descrito por un triángulo con coordenadas:

$$x_d = \begin{bmatrix} -0.5 & 0.5 & 0 & -0.5 \end{bmatrix}^T, \\ y_d = \begin{bmatrix} 0 & 0 & 0.5 & 0 \end{bmatrix}^T$$

con una matriz de ganancias $\mathbf{K} = \text{diag}(11, 11)$ y una $v_{max} = 0.15$ m/s.

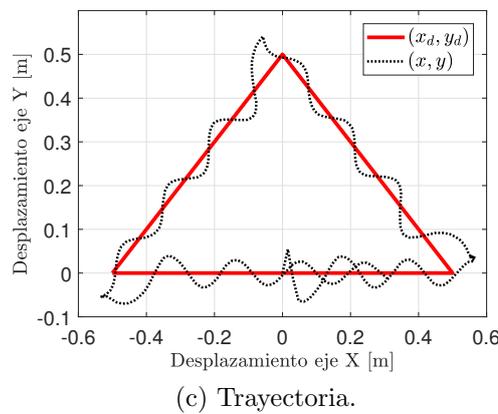
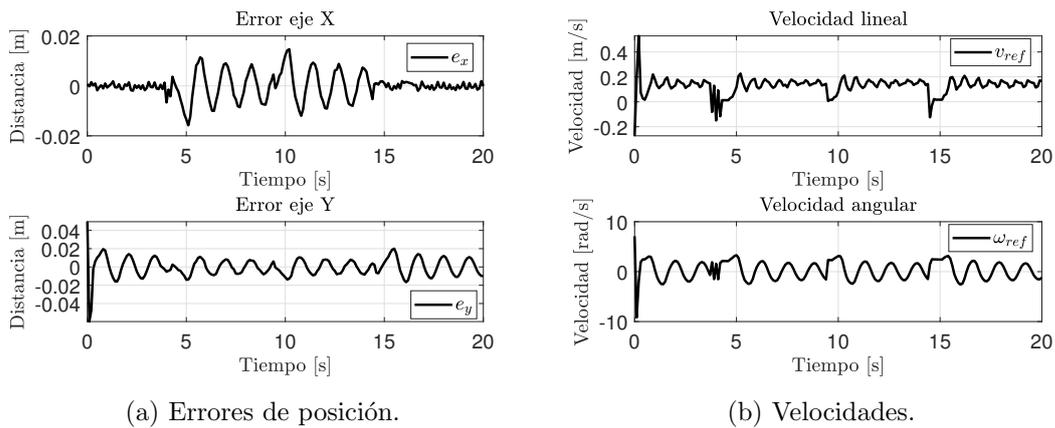


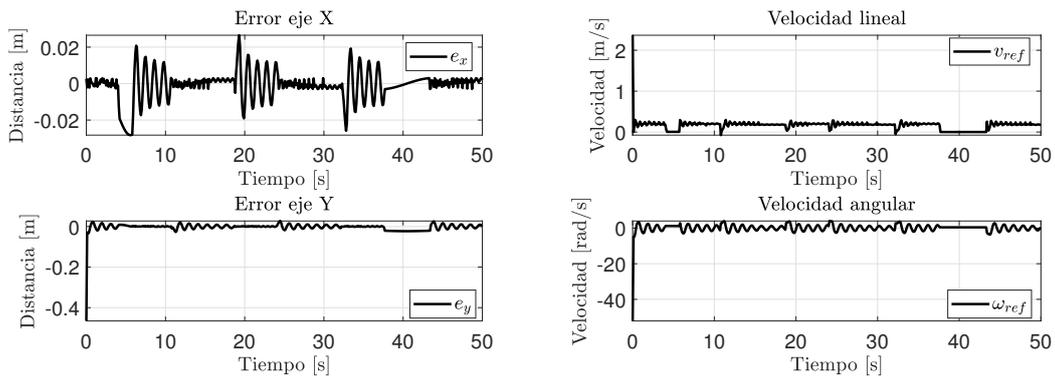
Figura 3.38: Respuesta Numérica No. 1 del CSC.

2. Condiciones iniciales $x_0 = 0$, $y_0 = 0$ y $\varphi_0 = -45^\circ$, y se deseó seguir el camino descrito por un rectángulo con coordenadas:

$$x_d = \begin{bmatrix} -0.75 & 0.75 & 0.75 & -0.75 & -0.75 \end{bmatrix}^T,$$

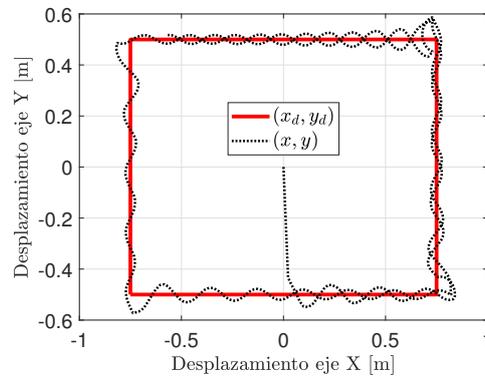
$$y_d = \begin{bmatrix} -0.5 & -0.5 & 0.5 & 0.5 & -0.5 \end{bmatrix}^T,$$

con una matriz de ganancias $\mathbf{K} = \text{diag}(10, 10)$ y una $v_{max} = 0.2$ m/s.



(a) Errores de posición.

(b) Velocidades.



(c) Trayectoria.

Figura 3.39: Respuesta Numérica No. 2 del CSC.

3. Condiciones iniciales $x_0 = 0$, $y_0 = 0$ y $\varphi_0 = -45^\circ$, y se deseó seguir el camino descrito por un hexágono con coordenadas:

$$x_d = [-1 \quad -0.5 \quad 0.5 \quad 1 \quad 1 \quad 0.5 \quad -0.5 \quad -1 \quad -1]^T,$$

$$y_d = [0.25 \quad 0.5 \quad 0.5 \quad 0.25 \quad -0.25 \quad -0.5 \quad -0.5 \quad -0.25 \quad 0.25]^T,$$

con una matriz de ganancias $\mathbf{K} = \text{diag}(1, 1)$ y una $v_{max} = 0.2$ m/s.

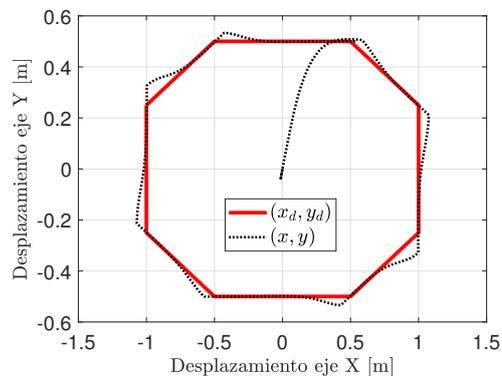
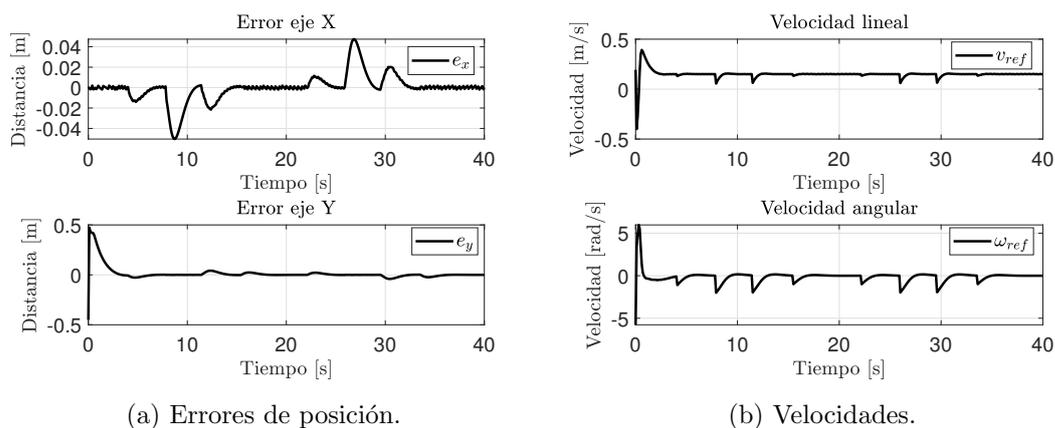


Figura 3.40: Respuesta Numérica No. 3 del CSC.

3.6.5. Control de Posición con Orientación Final

Existe una forma más de representar el modelo cinemático de un robot terrestre sobre ruedas en configuración diferencial, éste es en términos de coordenadas polares para poder definir la posición del robot.

Dicho modelo involucra el error de distancia m , error de ángulo ρ y su orientación final β con respecto al sistema de coordenadas relativo $\mathcal{R} = [X_{\mathcal{R}}, Y_{\mathcal{R}}]^T$

(véase la Figura 3.41). Dicho modelo cinemático en forma matricial es el siguiente [34, 40, 61]:

$$\begin{bmatrix} \dot{m} \\ \dot{\rho} \\ \dot{\beta} \end{bmatrix} = \begin{bmatrix} -\cos(\rho) & 0 \\ \frac{\sin(\rho)}{m} & -1 \\ \frac{\sin(\rho)}{m} & 0 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix}. \quad (3.53)$$

Estados: m, ρ, β .

Entradas v, ω .

Tomar en cuenta que este modelo es válido solo para $|m| > 0$, ya que cuando $m = 0$, los ángulos ρ y β son indefinidos.

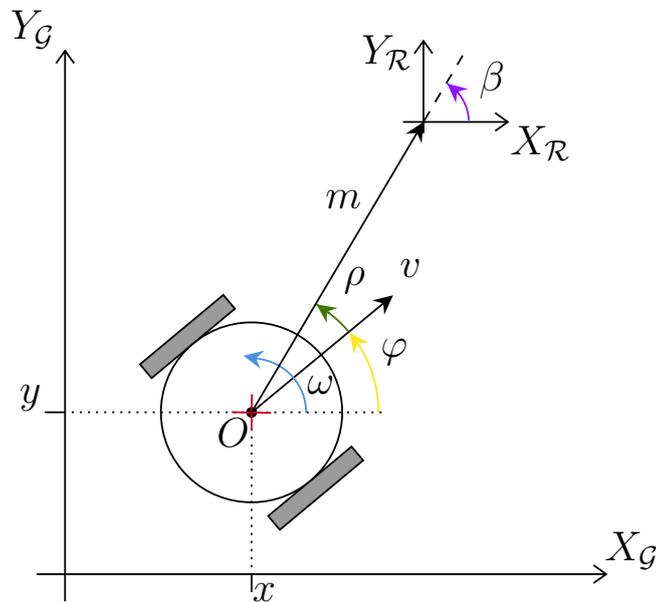


Figura 3.41: Esquema de la Postura del Robot en Coordenadas Polares.

Teorema 3.6.5: Control de Posición con Orientación (CPCO)

Sean las leyes de control

$$\begin{aligned} v_{ref} &= k_1 m \cos(\rho), \\ \omega_{ref} &= k_2 \rho + k_1 \cos(\rho) \sin(\rho) \left(\frac{\rho + \beta}{\rho} \right). \end{aligned} \quad (3.54)$$

El sistema (3.53) es AE si $k_1 > 0$ y $k_2 > 0$.

Prueba. Se propuso una función candidata de Lyapunov:

$$\mathcal{V}(\mathbf{p}) = \frac{1}{2} \mathbf{p}^T \mathbf{p}, \quad (3.55)$$

donde $\mathbf{p} = \begin{bmatrix} m & \rho & \beta \end{bmatrix}^T$.

Derivando la función candidata de Lyapunov se obtuvo:

$$\dot{\mathcal{V}} = \frac{1}{2} \mathbf{p}^T \dot{\mathbf{p}}. \quad (3.56)$$

Sustituyendo valores se obtuvo:

$$\dot{\mathcal{V}} = \underbrace{m\dot{m}}_{\dot{\mathcal{V}}_1} + \underbrace{\rho\dot{\rho} + \beta\dot{\beta}}_{\dot{\mathcal{V}}_2}. \quad (3.57)$$

Separando las dinámicas lineales y angulares de la derivada de la función candidata de Lyapunov, se volvió más sencillo el análisis y diseño del controlador.

Tomando el término $\dot{\mathcal{V}}_1$:

$$\dot{\mathcal{V}}_1 = m\dot{m} = -mv \cos(\rho). \quad (3.58)$$

Teniendo lo anterior, se debía asegurar que sea una función definida negativa para que el sistema sea AE. Para ello se eligió una velocidad lineal haga cumplir dicha condición [40]:

$$v = k_1 m \cos(\rho). \quad (3.59)$$

Sustituyendo (3.59) en (3.58) se obtuvo:

$$\dot{\mathcal{V}}_1 = -k_1 m^2 \cos^2(\rho) < 0. \quad (3.60)$$

Con esta definición m puede ser negativa o positiva y el ángulo ρ también, y al estar al cuadrado siempre serán positivas por lo que se cumple la definición negativa de la función. La ganancia k_1 ayudará a regular el comportamiento del controlador (velocidad de convergencia). Por lo tanto el sistema (3.59) es AE si $k_1 > 0$.

Para el segundo término $\dot{\mathcal{V}}_2$:

$$\dot{\mathcal{V}}_2 = \rho\dot{\rho} + \beta\dot{\beta} = \rho \left(-\omega + \frac{v}{m} \sin(\rho) \right) + \beta \left(\frac{v}{m} \sin(\rho) \right). \quad (3.61)$$

Sustituyendo el valor de la velocidad lineal v (3.59) en (3.61) se obtuvo:

$$\dot{\mathcal{V}}_2 = \rho \left(-\omega + \frac{k_1 m \cos(\rho)}{m} \operatorname{sen}(\rho) \right) + \beta \left(\frac{k_1 m \cos(\rho)}{m} \operatorname{sen}(\rho) \right). \quad (3.62)$$

Realizando las simplificaciones posibles se obtuvo:

$$\begin{aligned} \dot{\mathcal{V}}_2 &= \rho (-\omega + k_1 \cos(\rho) \operatorname{sen}(\rho)) + \beta (k_1 \cos(\rho) \operatorname{sen}(\rho)) \\ \dot{\mathcal{V}}_2 &= -\rho\omega + k_1 \cos(\rho) \operatorname{sen}(\rho) (\rho + \beta) \end{aligned}$$

La expresión anterior se reescribió como:

$$\dot{\mathcal{V}}_2 = \rho \left(-\omega + k_1 \cos(\rho) \operatorname{sen}(\rho) \left(\frac{\rho + \beta}{\rho} \right) \right). \quad (3.63)$$

Para asegurar que $\dot{\mathcal{V}}_2$ sea definida negativa, se eligió una velocidad angular ω de tal manera que siempre se cumpliera. Por lo tanto se propuso la velocidad angular [40]:

$$\omega = k_2 \rho + k_1 \cos(\rho) \operatorname{sen}(\rho) \left(\frac{\rho + \beta}{\rho} \right). \quad (3.64)$$

Sustituyendo (3.64) en (3.63) se obtuvo:

$$\dot{\mathcal{V}}_2 = -k_2 \rho^2 < 0. \quad (3.65)$$

con $k_2 > 0$, el sistema siempre será AE. ■

Después del diseño analítico del controlador se realizaron una serie de simulaciones para la validación numérica con diferentes condiciones iniciales y finales.

- Condiciones iniciales iguales a cero y se deseó llevar a las coordenadas $x_d = 2$, $y_d = 3$, $\varphi_d = 45^\circ$, con $k_1 = 0.7$ y $k_2 = 0.8$.

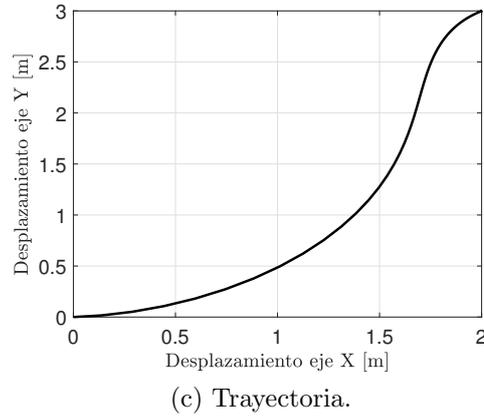
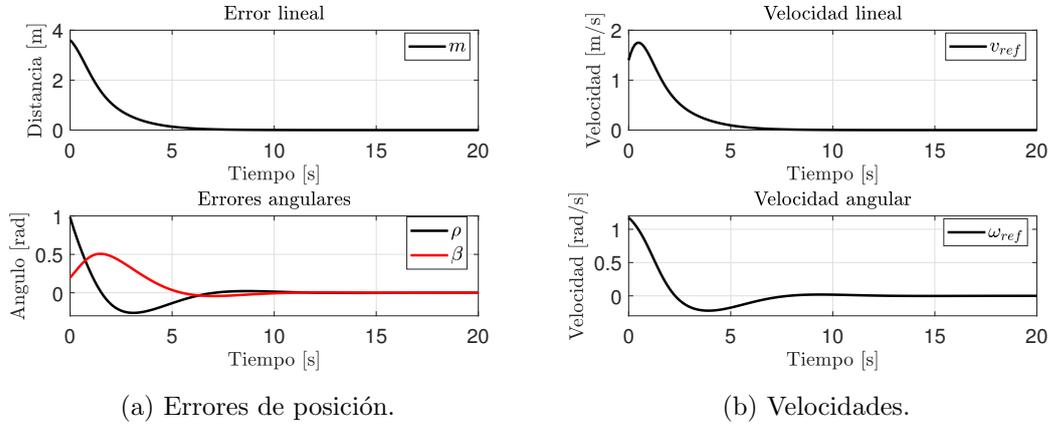


Figura 3.42: Respuesta Numérica No. 1 del CPCO.

- Condiciones iniciales iguales a cero y se deseó llevar a las coordenadas $x_d = 2$, $y_d = 8$, $\varphi_d = 60^\circ$, con $k_1 = 1.2$ y $k_2 = 1$.

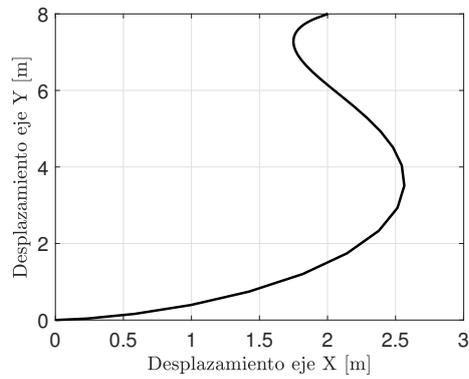
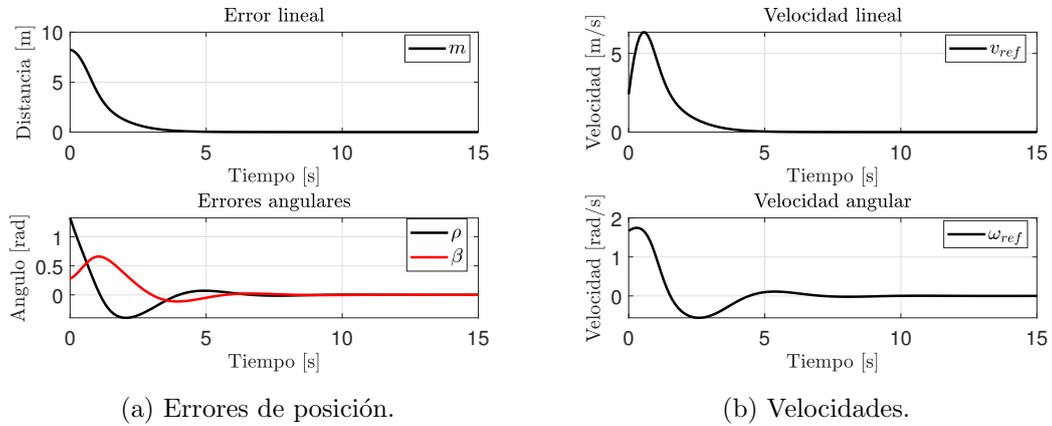


Figura 3.43: Respuesta Numérica No. 2 del CPCO.

- Condiciones iniciales iguales a cero y se deseó llevar a las coordenadas $x_d = -2$, $y_d = 10$, $\varphi_d = 120^\circ$, con $k_1 = 0.8$ y $k_2 = 1.5$.

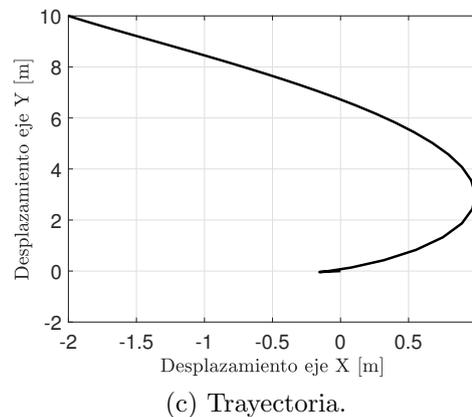
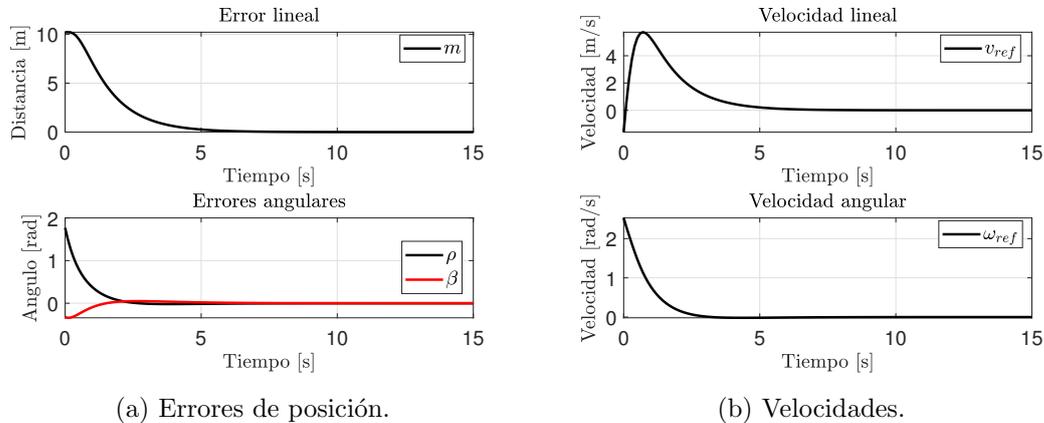


Figura 3.44: Respuesta Numérica No. 3 del CPCO.

3.6.6. Control de Seguimiento de Trayectoria con Retardo

Después de que se realizó el análisis y diseño de los controladores anteriores se tomaron en cuenta condiciones ideales, donde no existe ningún tipo de retardo en el intercambio de información. Por otro lado, se sabe que al momento de validar experimentalmente, el sensado de la posición del robot móvil se hará por medio de un sistema de visión monocular y el uso de algoritmos de tracking para el procesamiento de la imagen, este procesamiento conlleva un cálculo y por lo tanto un retardo no tomado en cuenta en las leyes de control anteriores.

En la Figura 3.45 se puede observar en donde está inducido el retardo τ por parte del cálculo de la localización del robot por el uso de la cámara.

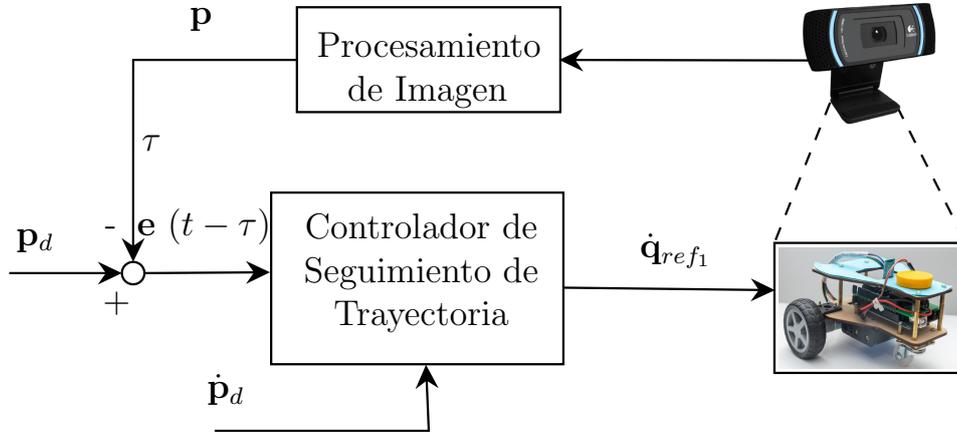


Figura 3.45: Esquema del CSTR.

Por lo tanto, a continuación se presenta una estrategia de control considerando el retardo τ dando un margen de estabilidad para diferentes ganancias por medio de los límites permisibles de la matriz de ganancias \mathbf{K} .

Teorema 3.6.6: Control de Seguimiento de Trayectoria con Retardo (CSTR)

Sea la ley de control

$$\dot{\mathbf{q}}_{ref} = \mathbf{J}_{xy}^{-1}(\mathbf{q}) (\dot{\mathbf{p}}_{xy_d} - \mathbf{K}\mathbf{e}(t - \tau)), \quad (3.66)$$

donde $\tau > 0$ es el retardo inducido por el procesamiento de la imagen y $\dot{\mathbf{q}}_{ref} := [v_{ref} \ \omega_{ref}]^T$.

El sistema (3.21) es AE si $\mathbf{K} \in \mathbb{R}^{2 \times 2}$ está dada por:

$$\mathbf{K} := \begin{bmatrix} k_1 & 0 \\ 0 & k_2 \end{bmatrix},$$

con $k_{1,2} \in \left(0, \frac{\pi}{2\tau}\right)$. Más aún, si $k_1 = k_2 = \frac{e^{-1}}{\tau}$ entonces la respuesta del sistema alcanzará su máximo decaimiento exponencial.

Prueba. Se tiene que el sistema (3.21) con la estrategia de control (3.66) resulta en el siguiente sistema en lazo cerrado:

$$\dot{\mathbf{e}}(t) = -\mathbf{K}\mathbf{e}(t - \tau). \quad (3.67)$$

Se observó que para analizar la estabilidad del sistema (3.67) es suficiente con garantizar que las soluciones de la siguiente función:

$$\Delta(s, \tau) = s + k_i e^{-\tau s}, \quad i \in \{1, 2\}, \quad (3.68)$$

se encontraran en el semiplano izquierdo del plano complejo (definido de ahora en adelante como \mathbb{C}_-).

Para garantizar lo anterior, sea $s = i\omega$, con $\omega \geq 0$ se obtuvieron las siguientes ecuaciones:

$$k_i \cos(\tau\omega) = 0, \quad (3.69)$$

$$\omega = k_i \sin(\tau\omega). \quad (3.70)$$

De (3.69) se tiene que ω debe ser igual a $\omega = \frac{\pi}{2} + 2\ell$ con $\ell \in \mathbb{Z}$. Por tanto, dado que para $\tau = 0$ el sistema es estable con $k_i > 0$, entonces aplicando la continuidad de las raíces [60] de (3.70) se obtuvo que Δ tiene sus soluciones en \mathbb{C}_- ssi $k_i \in \left(0, \frac{\pi}{2\tau}\right)$, demostrando con esto la primera parte del resultado.

Para garantizar el máximo decaimiento exponencial, se procedió de manera similar pero considerando $s = -\sigma + i\omega$ de donde se obtuvieron las siguientes expresiones:

$$-\sigma + k_i e^{\tau\sigma} \cos(\tau\omega) = 0, \quad (3.71)$$

$$\omega = k_i e^{\tau\sigma} \sin(\tau\omega). \quad (3.72)$$

Siguiendo un argumento similar a lo antes expuesto se concluyó que para $k_i = \frac{e^{-1}}{\tau}$ la respuesta del sistema alcanza su máximo decaimiento exponencial. ■

Se realizaron una serie de simulaciones para diferentes ganancias permisibles obtenidas anteriormente. Se consideró un retardo $\tau = 0.1$ s.

1. Condiciones iniciales $x_0 = 1.8$, $y_0 = 0$ y $\varphi_0 = 0^\circ$, y se deseó seguir la trayectoria de un círculo parametrizada en el tiempo:

$$x_d(t) = \cos(0.1t),$$

$$y_d(t) = \text{sen}(0.1t),$$

con una matriz de ganancias $\mathbf{K} = \text{diag}(8, 8)$.

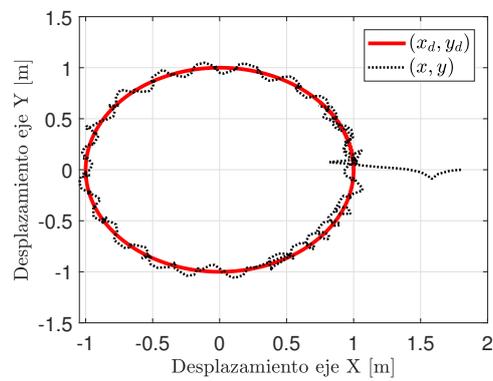
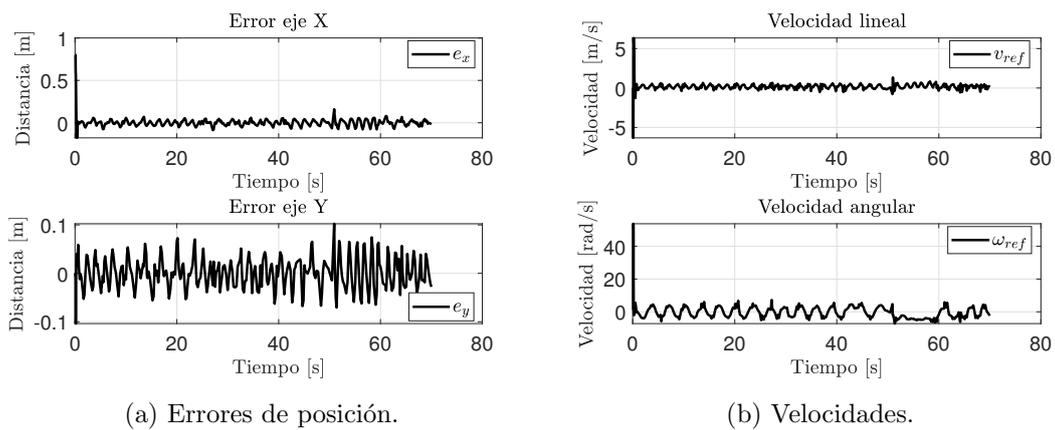


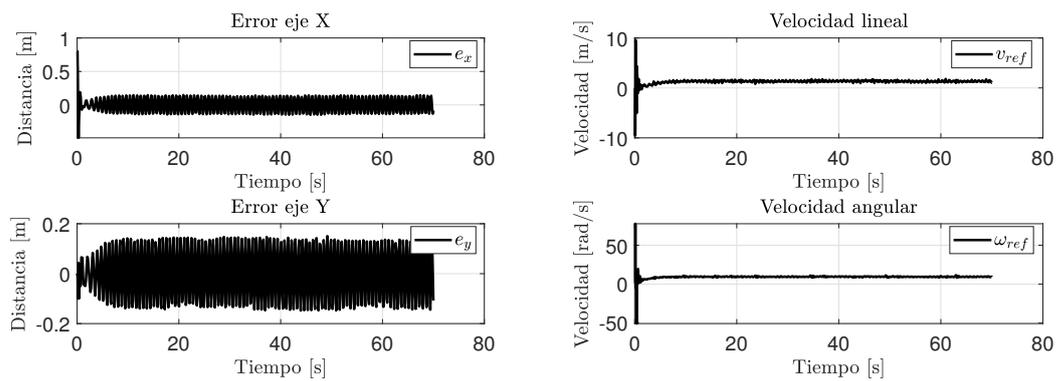
Figura 3.46: Respuesta Numérica No. 1 del CSTR.

2. Condiciones iniciales $x_0 = 1.8$, $y_0 = 0$ y $\varphi_0 = 0^\circ$, y se deseó seguir la trayectoria de un círculo parametrizada en el tiempo:

$$x_d(t) = \cos(0.1t),$$

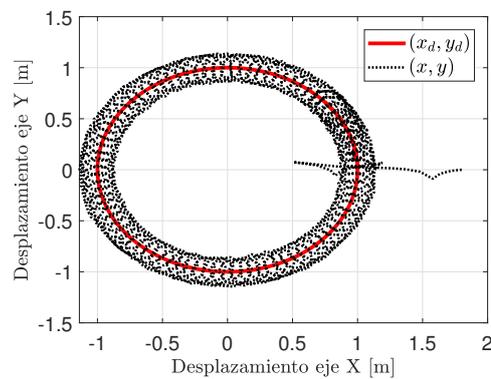
$$y_d(t) = \sin(0.1t),$$

con una matriz de ganancias $\mathbf{K} = \text{diag}(12, 12)$.



(a) Errores de posición.

(b) Velocidades.



(c) Trayectoria.

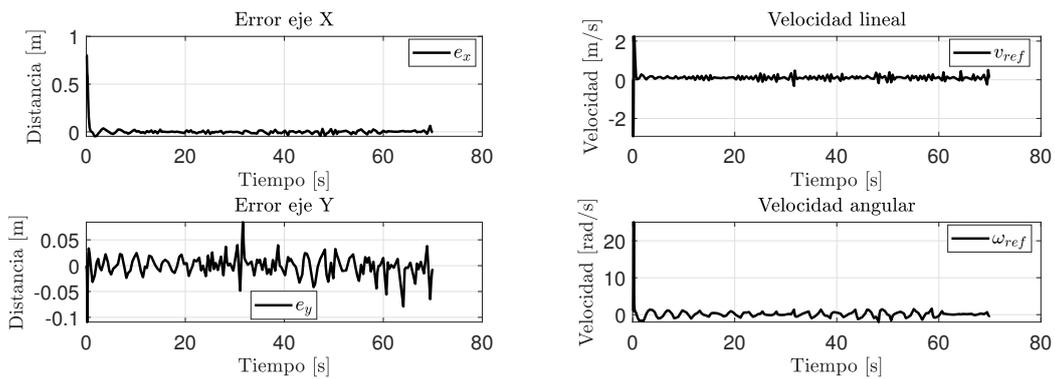
Figura 3.47: Respuesta Numérica No. 2 del CSTR.

3. Condiciones iniciales $x_0 = 1.8$, $y_0 = 0$ y $\varphi_0 = 0^\circ$, y se deseó seguir la trayectoria de un círculo parametrizada en el tiempo:

$$x_d(t) = \cos(0.1t),$$

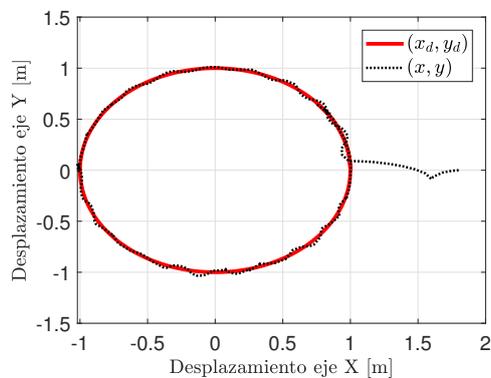
$$y_d(t) = \text{sen}(0.1t),$$

con una matriz de ganancias $\mathbf{K} = \text{diag}\left(\frac{e^{-1}}{0.1}, \frac{e^{-1}}{0.1}\right)$.



(a) Errores de posición.

(b) Velocidades.



(c) Trayectoria.

Figura 3.48: Respuesta Numérica No. 3 del CSTR.

Modelado y Control para la Formación de un Sistema Multiagente

4.1. Formación de Dos Robots

En esta sección se muestra el modelo geométrico y la cinemática de la formación de dos robots móviles de tracción diferencial. Posteriormente se presenta el diseño del control de posición y el control de seguimiento de trayectoria de la formación, para así mantener la distancia y el ángulo que se forma entre ellos.

4.1.1. Modelo Cinemático

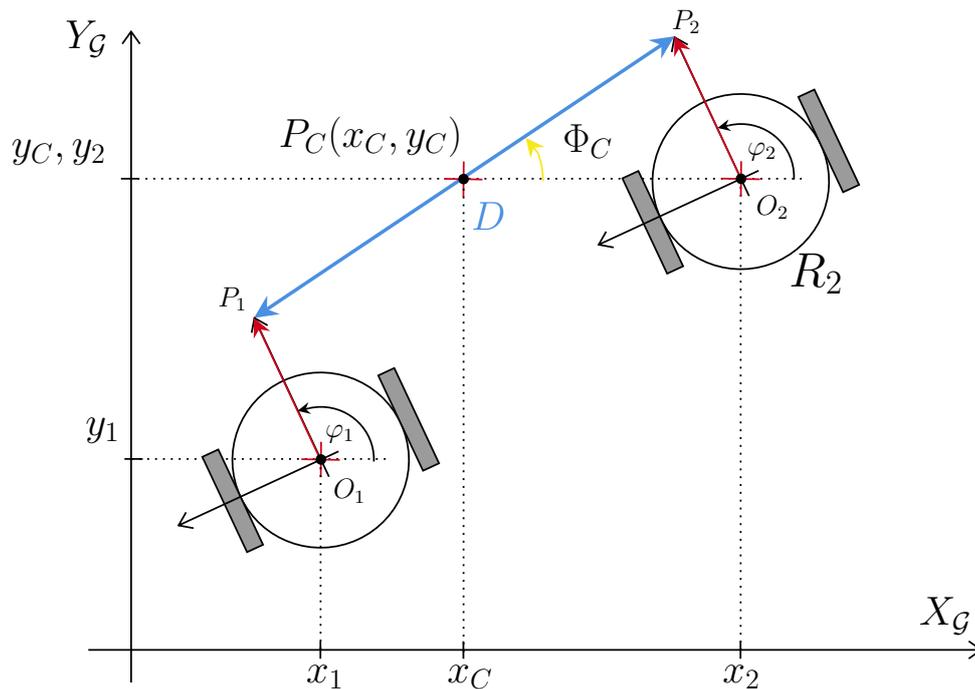


Figura 4.1: Esquema de la Formación de Dos Robots.

La distancia que une a los puntos P_i los cuales representan el punto descentralizado de los robots 1 y 2 es un vector de distancia con magnitud D , y que su centro está representado por el punto $P_C(x_C, y_C)$, dichas coordenadas se definieron mediante la ecuación de punto medio:

$$\begin{aligned} x_C &= \frac{x_1 + x_2}{2}, \\ y_C &= \frac{y_1 + y_2}{2}. \end{aligned} \tag{4.1}$$

Para las ecuaciones de forma es necesario determinar la distancia entre los robots D la cual se obtuvo por la fórmula de distancia euclidiana entre dos puntos:

$$D = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}. \tag{4.2}$$

Para el ángulo de orientación con respecto al eje X_G , se completa el triángulo rectángulo y se tienen los dos catetos conocidos, por lo que el ángulo se definió como:

$$\Phi_C = \tan^{-1} \left(\frac{y_2 - y_1}{x_2 - x_1} \right). \tag{4.3}$$

Por lo tanto el modelo geométrico en forma matricial quedó definido como:

$$\begin{bmatrix} x_C \\ y_C \\ D \\ \Phi_C \end{bmatrix} = \begin{bmatrix} \frac{x_1 + x_2}{2} \\ \frac{y_1 + y_2}{2} \\ \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \\ \tan^{-1} \left(\frac{y_2 - y_1}{x_2 - x_1} \right) \end{bmatrix}. \tag{4.4}$$

Con el modelo geométrico ya obtenido, se obtuvo el modelo cinemático diferencial de la formación por medio de la derivadas temporales de cada una de las ecuaciones anteriores mediante la regla de la cadena.

$$\begin{aligned} \frac{dx_C}{dt} &= \frac{\partial x_C}{\partial x_1} \frac{dx_1}{dt} + \frac{\partial x_C}{\partial y_1} \frac{dy_1}{dt} + \frac{\partial x_C}{\partial x_2} \frac{dx_2}{dt} + \frac{\partial x_C}{\partial y_2} \frac{dy_2}{dt}, \\ \dot{x}_C &= \frac{\partial x_C}{\partial x_1} \dot{x}_1 + \frac{\partial x_C}{\partial y_1} \dot{y}_1 + \frac{\partial x_C}{\partial x_2} \dot{x}_2 + \frac{\partial x_C}{\partial y_2} \dot{y}_2, \\ \dot{x}_C &= \frac{1}{2} \dot{x}_1 + \frac{1}{2} \dot{x}_2. \end{aligned} \tag{4.5}$$

$$\begin{aligned}
\frac{dy_C}{dt} &= \frac{\partial y_C}{\partial x_1} \frac{dx_1}{dt} + \frac{\partial y_C}{\partial y_1} \frac{dy_1}{dt} + \frac{\partial y_C}{\partial x_2} \frac{dx_2}{dt} + \frac{\partial y_C}{\partial y_2} \frac{dy_2}{dt}, \\
\dot{y}_C &= \frac{\partial y_C}{\partial x_1} \dot{x}_1 + \frac{\partial y_C}{\partial y_1} \dot{y}_1 + \frac{\partial y_C}{\partial x_2} \dot{x}_2 + \frac{\partial y_C}{\partial y_2} \dot{y}_2, \\
\dot{x}_C &= \frac{1}{2} \dot{y}_1 + \frac{1}{2} \dot{y}_2.
\end{aligned} \tag{4.6}$$

$$\begin{aligned}
\frac{dD}{dt} &= \frac{\partial D}{\partial x_1} \frac{dx_1}{dt} + \frac{\partial D}{\partial y_1} \frac{dy_1}{dt} + \frac{\partial D}{\partial x_2} \frac{dx_2}{dt} + \frac{\partial D}{\partial y_2} \frac{dy_2}{dt}, \\
\dot{D} &= \frac{\partial D}{\partial x_1} \dot{x}_1 + \frac{\partial D}{\partial y_1} \dot{y}_1 + \frac{\partial D}{\partial x_2} \dot{x}_2 + \frac{\partial D}{\partial y_2} \dot{y}_2, \\
\dot{D} &= \frac{x_1 - x_2}{\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}} \dot{x}_1 + \frac{y_1 - y_2}{\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}} \dot{y}_1, \\
&\quad - \frac{x_1 - x_2}{\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}} \dot{x}_2 - \frac{y_1 - y_2}{\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}} \dot{y}_2.
\end{aligned} \tag{4.7}$$

$$\begin{aligned}
\frac{d\Phi_C}{dt} &= \frac{\partial \Phi_C}{\partial x_1} \frac{dx_1}{dt} + \frac{\partial \Phi_C}{\partial y_1} \frac{dy_1}{dt} + \frac{\partial \Phi_C}{\partial x_2} \frac{dx_2}{dt} + \frac{\partial \Phi_C}{\partial y_2} \frac{dy_2}{dt}, \\
\dot{\Phi}_C &= \frac{\partial \Phi_C}{\partial x_1} \dot{x}_1 + \frac{\partial \Phi_C}{\partial y_1} \dot{y}_1 + \frac{\partial \Phi_C}{\partial x_2} \dot{x}_2 + \frac{\partial \Phi_C}{\partial y_2} \dot{y}_2, \\
\dot{\Phi}_C &= - \frac{y_1 - y_2}{(x_1 - x_2)^2 + (y_1 - y_2)^2} \dot{x}_1 + \frac{x_1 - x_2}{(x_1 - x_2)^2 + (y_1 - y_2)^2} \dot{y}_1, \\
&\quad + \frac{y_1 - y_2}{(x_1 - x_2)^2 + (y_1 - y_2)^2} \dot{x}_2 - \frac{x_1 - x_2}{(x_1 - x_2)^2 + (y_1 - y_2)^2} \dot{y}_2.
\end{aligned} \tag{4.8}$$

Por lo que la cinemática en forma matricial quedó como:

$$\underbrace{\begin{bmatrix} \dot{x}_C \\ \dot{y}_C \\ \dot{D} \\ \dot{\Phi}_C \end{bmatrix}}_{\dot{\mathbf{p}}} = \underbrace{\begin{bmatrix} \frac{\partial x_C}{\partial x_1} & \frac{\partial x_C}{\partial y_1} & \frac{\partial x_C}{\partial x_2} & \frac{\partial x_C}{\partial y_2} \\ \frac{\partial y_C}{\partial x_1} & \frac{\partial y_C}{\partial y_1} & \frac{\partial y_C}{\partial x_2} & \frac{\partial y_C}{\partial y_2} \\ \frac{\partial D}{\partial x_1} & \frac{\partial D}{\partial y_1} & \frac{\partial D}{\partial x_2} & \frac{\partial D}{\partial y_2} \\ \frac{\partial \Phi_C}{\partial x_1} & \frac{\partial \Phi_C}{\partial y_1} & \frac{\partial \Phi_C}{\partial x_2} & \frac{\partial \Phi_C}{\partial y_2} \end{bmatrix}}_{\mathbf{J}} \underbrace{\begin{bmatrix} \dot{x}_1 \\ \dot{y}_1 \\ \dot{x}_2 \\ \dot{y}_2 \end{bmatrix}}_{\dot{\mathbf{q}}} \tag{4.9}$$

donde:

$$\frac{\partial x_C}{\partial x_1} = \frac{1}{2}, \quad \frac{\partial x_C}{\partial y_1} = 0, \quad \frac{\partial x_C}{\partial x_2} = \frac{1}{2}, \quad \frac{\partial x_C}{\partial y_2} = 0,$$

$$\begin{aligned} \frac{\partial y_C}{\partial x_1} &= 0, & \frac{\partial y_C}{\partial y_1} &= \frac{1}{2}, & \frac{\partial y_C}{\partial x_2} &= 0, & \frac{\partial y_C}{\partial y_2} &= \frac{1}{2}, \\ \frac{\partial D}{\partial x_1} &= \frac{x_1 - x_2}{\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}}, & \frac{\partial D}{\partial y_1} &= \frac{y_1 - y_2}{\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}}, \\ \frac{\partial D}{\partial x_2} &= -\frac{x_1 - x_2}{\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}}, & \frac{\partial D}{\partial y_2} &= -\frac{y_1 - y_2}{\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}}, \\ \frac{\partial \Phi_C}{\partial x_1} &= -\frac{y_1 - y_2}{(x_1 - x_2)^2 + (y_1 - y_2)^2}, & \frac{\partial \Phi_C}{\partial y_1} &= \frac{x_1 - x_2}{(x_1 - x_2)^2 + (y_1 - y_2)^2}, \\ \frac{\partial \Phi_C}{\partial x_2} &= \frac{y_1 - y_2}{(x_1 - x_2)^2 + (y_1 - y_2)^2}, & \frac{\partial \Phi_C}{\partial y_2} &= -\frac{x_1 - x_2}{(x_1 - x_2)^2 + (y_1 - y_2)^2}. \end{aligned}$$

Estados: $\dot{x}_C, \dot{y}_C, \dot{D}, \dot{\Phi}_C$.

Entradas: $\dot{x}_1, \dot{y}_1, \dot{x}_2, \dot{y}_2$.

4.1.2. Control de Posición

Al tener el modelo cinemático de la formación (4.9) se continuó diseñando un Control de Posición de la Formación de dos Robots(CPF2R) de un punto a otro. En la Figura 4.2 se muestra el diagrama de bloques que se propuso para la solución de esta tarea.

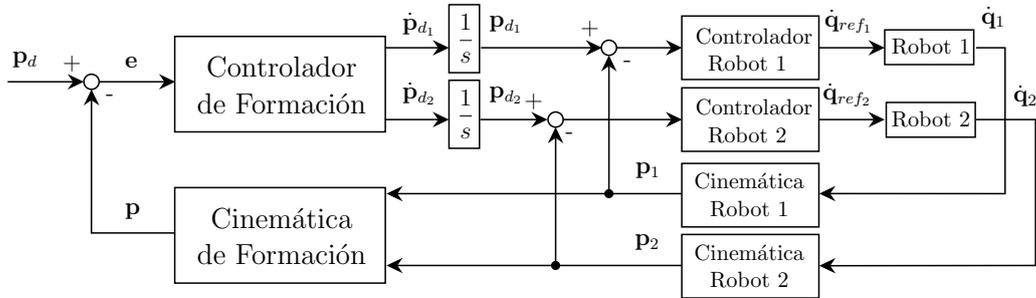


Figura 4.2: Diagrama de Bloques para el CPF2R.

Se deseó diseñar un controlador que permita llevar el punto P_C (véase la Figura 4.1) a una posición deseada, con una orientación deseada y una distancia deseada entre los robots.

Dadas las posiciones deseadas \mathbf{p}_d y la posición actual de la formación \mathbf{p} se calculó el error \mathbf{e} el cual funcionó como entrada para el controlador de formación el cual está basado en el modelo cinemático total de la formación, ese controlador determina las velocidades deseadas $\dot{\mathbf{p}}_1$ y $\dot{\mathbf{p}}_2$ correspondientes a cada uno

de los robots, se integran y se obtienen las posiciones deseadas, para poder así encontrar los errores de posición de cada robot.

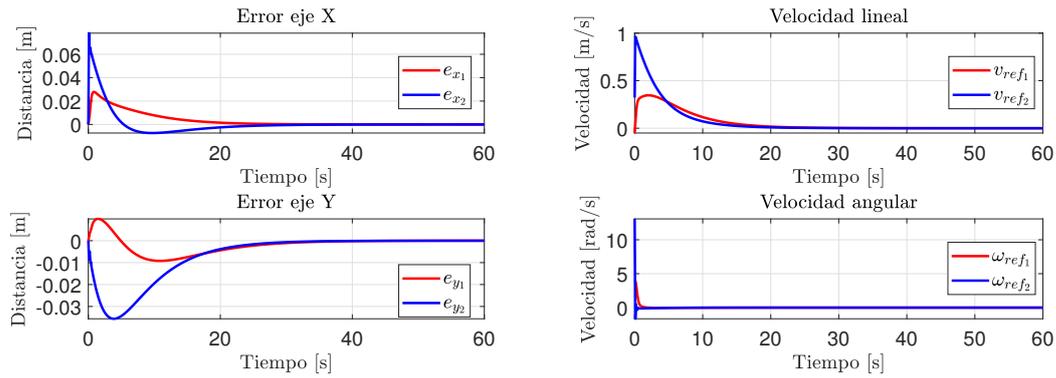
Esos errores de posición entran a un controlador para cada uno de los robots, el cual calcula velocidades de referencia para cada uno de los robots $\dot{\mathbf{q}}_{ref_1}$ y $\dot{\mathbf{q}}_{ref_2}$, entran a la planta (robots) y se obtienen las velocidades actuales de los robots $\dot{\mathbf{q}}_1$ y $\dot{\mathbf{q}}_2$. Finalmente estas velocidades entran a las cinemáticas de cada uno de los robots, para obtener posiciones actuales y así realimentar con la cinemática de la formación.

Tomando en cuenta las condiciones anteriormente mencionadas, al nivel de la formación se utilizó la ley de control para el CPO mostrada en (3.27), esta ley ya se demostró que es AE. Al nivel del robot, se utilizó la ley de control para el CST mostrada en (3.36), por lo tanto la ley para del i -ésimo robot es:

$$\dot{\mathbf{q}}_{ref_i} = \mathbf{J}_{xy_i}^{-1} \left(\dot{\mathbf{p}}_{xy_{d_i}} - \mathbf{K}_i \mathbf{e}_i \right) \quad (4.10)$$

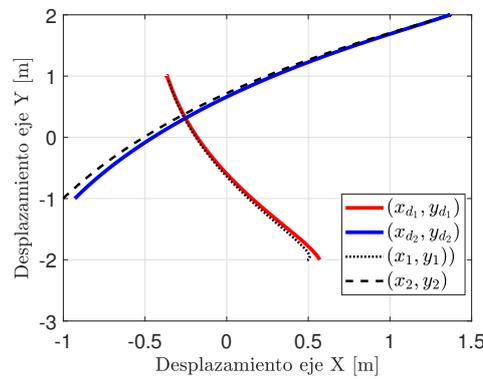
con $i \in \{1, 2\}$, esta ley de control también ya fue demostrada que presenta una dinámica AE. Por lo tanto, el diagrama de control de la Figura 4.2 representa un control en cascada.

- Se realizó una simulación para condiciones iniciales $x_{0_1} = 0.5, y_{0_1} = -2, \varphi_{0_1} = 0^\circ, x_{0_2} = -1, y_{0_2} = -1$ y $\varphi_{0_2} = 0^\circ$, para mover a las coordenadas deseadas el centro de la formación $x_d = 0.5, y_d = 1.5$, una distancia entre los robots de $D_d = 2$ y con ángulo de orientación en la formación de $\Phi_{C_d} = 30^\circ$. Las matrices de ganancia utilizadas fueron $\mathbf{K} = \text{diag}(0.2, 0.2, 0.2, 0.2), \mathbf{K}_1 = \text{diag}(0.2, 0.2), \mathbf{K}_2 = \mathbf{K}_1$.



(a) Errores de posición.

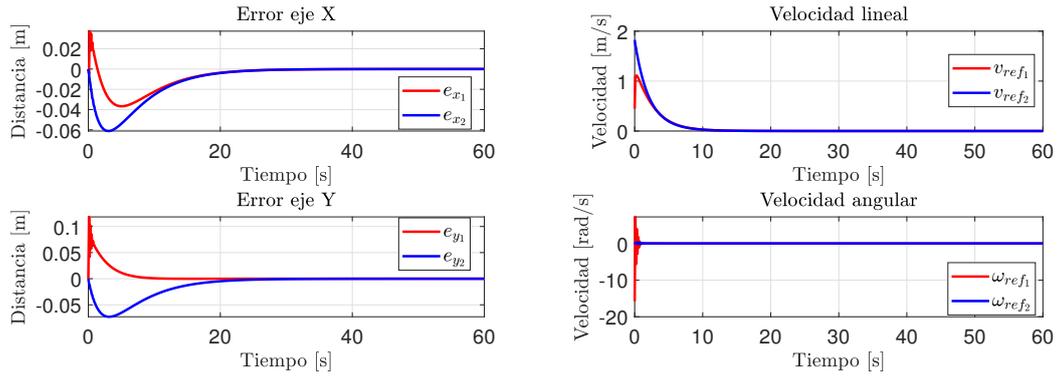
(b) Velocidades.



(c) Trayectoria.

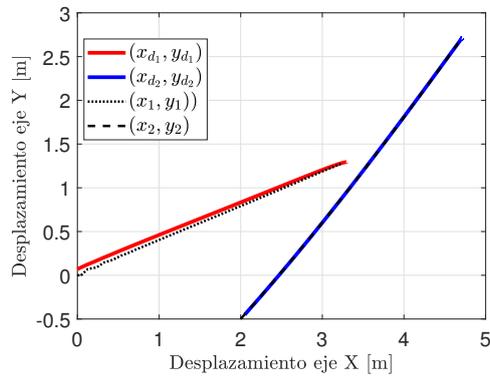
Figura 4.3: Respuestas Numéricas No. 1 del CPF2R.

2. Se realizó una simulación para condiciones iniciales $x_{0_1} = 0$, $y_{0_1} = 0$, $\varphi_{0_1} = 90^\circ$, $x_{0_2} = 2$, $y_{0_2} = -0.5$ y $\varphi_{0_2} = 45^\circ$, para mover a las coordenadas deseadas el centro de la formación $x_d = 4$, $y_d = 2$, una distancia entre los robots de $D_d = 2$ y con ángulo de orientación en la formación de $\Phi_{C_d} = 45^\circ$. Las matrices de ganancia utilizadas fueron $\mathbf{K} = \text{diag}(0.4, 0.4, 0.4, 0.4)$, $\mathbf{K}_1 = \text{diag}(0.2, 0.2)$, $\mathbf{K}_2 = \mathbf{K}_1$.



(a) Errores de posición.

(b) Velocidades.



(c) Trayectoria.

Figura 4.4: Respuestas Numéricas No. 2 del CPF2R.

4.1.3. Control de Seguimiento de Trayectoria

En la Figura 4.5 se propuso el diagrama de bloques para el Control de Seguimiento de Trayectoria de la Formación de dos Robots (CSTF2R).

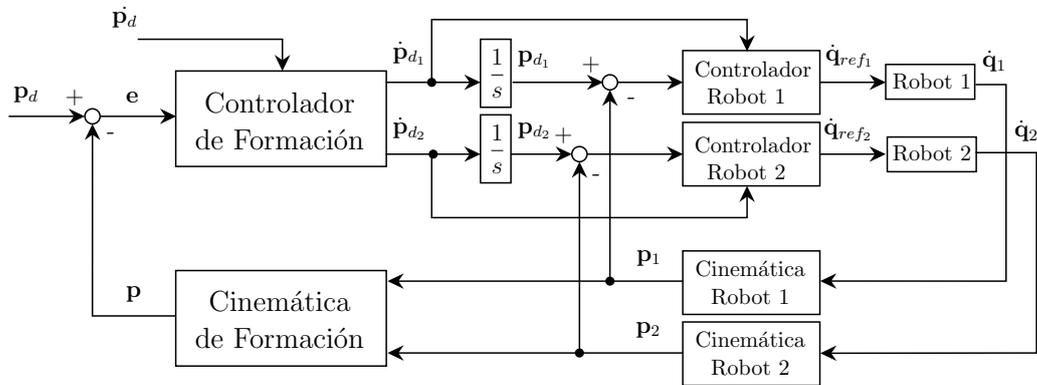


Figura 4.5: Diagrama de Bloques para el CSTF2R.

A diferencia del diagrama de bloques de la Figura 4.2, en el diagrama de la Figura 4.5 se ve que en el controlador de formación tuvo una entrada extra la cual es la velocidad deseada, también para el controlador de los robots se tomaron las velocidades deseadas como una entrada extra para cada uno de los mismos. Por lo tanto al nivel de formación, se utilizó la ley de control CST mostrada en (3.36) y al nivel de robot, se utilizó la ley de control mostrada en (4.10).

Para validar las leyes de control propuestas, se realizaron una serie de simulaciones mostradas a continuación:

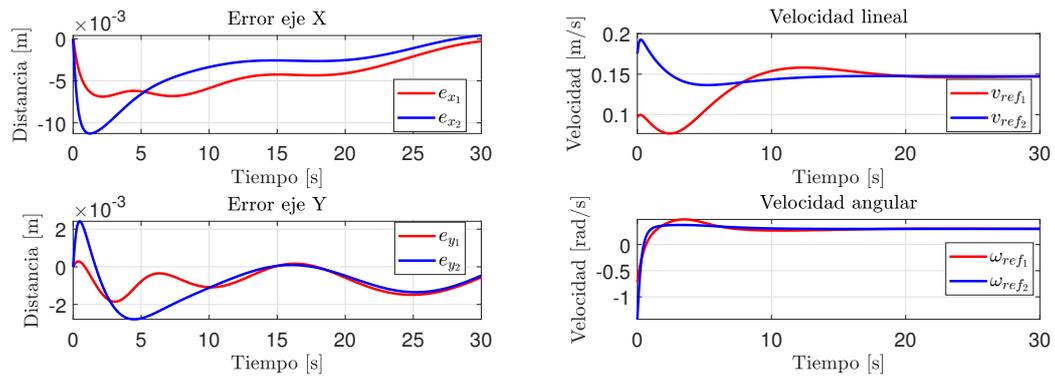
Círculo

Para la simulación se tomaron las siguientes condiciones iniciales $x_{0_1} = 0$, $y_{0_1} = 0$, $\varphi_{0_1} = 0^\circ$, $x_{0_2} = 0.25$, $y_{0_2} = 0$ y $\varphi_{0_2} = 0^\circ$, y se deseó que la formación siguiera la trayectoria deseada representada por una función seno la cual está parametrizada por:

$$x_d(t) = 0.5 \cos(0.3 \cdot t)$$

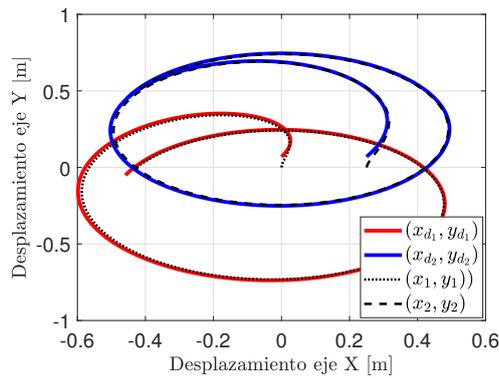
$$y_d(t) = 0.5 \operatorname{sen}(0.3 \cdot t)$$

con una distancia entre robots de $D_d = 0.5$, ángulo de orientación en la formación de $\Phi_{C_d} = 90^\circ$ y con unas matrices de ganancia $\mathbf{K} = \operatorname{diag}(0.2, 0.2, 0.2, 0.2)$, $\mathbf{K}_1 = \operatorname{diag}(0.1, 0.1)$, $\mathbf{K}_2 = \mathbf{K}_1$.



(a) Errores de posición.

(b) Velocidades.



(c) Trayectorias.

Figura 4.6: Respuestas Numéricas del CSTF2R al Seguimiento de un Círculo.

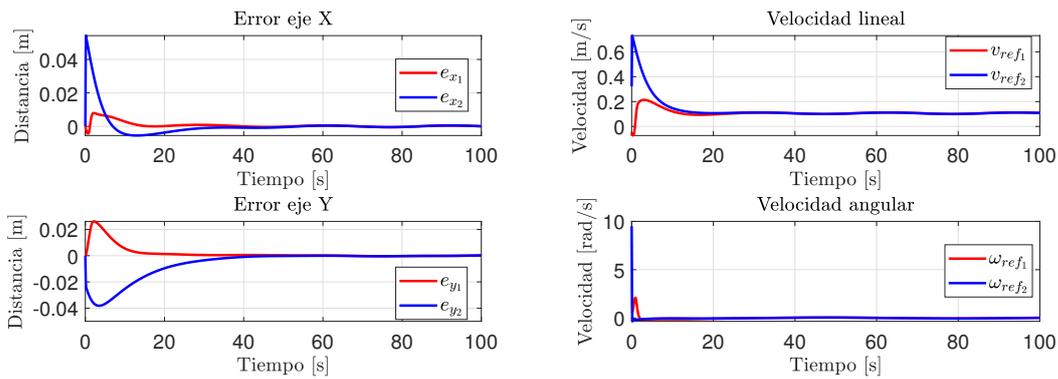
Función seno

Para la simulación se tomaron las siguientes condiciones iniciales $x_{0_1} = 0.5$, $y_{0_1} = -2$, $\varphi_{0_1} = 0^\circ$, $x_{0_2} = -1$, $y_{0_2} = -1$ y $\varphi_{0_2} = 0^\circ$, y se deseó que la formación siguiera la trayectoria deseada representada por una función seno la cual está parametrizada por:

$$x_d(t) = 0.1t$$

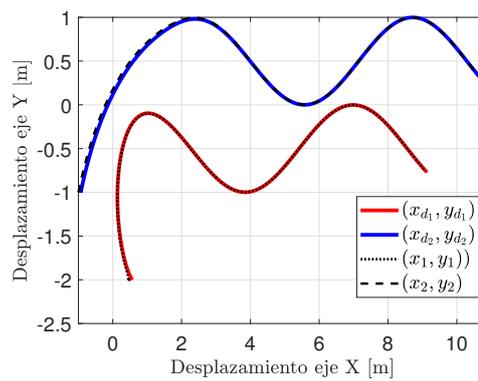
$$y_d(t) = 0.5 \text{sen}(0.1 \cdot t)$$

con una distancia entre robots de $D_d = 2$, ángulo de orientación en la formación de $\Phi_{C_d} = 30^\circ$ y con unas matrices de ganancia $\mathbf{K} = \text{diag}(0.2, 0.2, 0.2, 0.2)$, $\mathbf{K}_1 = \text{diag}(0.1, 0.1)$, $\mathbf{K}_2 = \mathbf{K}_1$.



(a) Errores de posición.

(b) Velocidades.



(c) Trayectorias.

Figura 4.7: Respuestas Numéricas del CSTF2R al Seguimiento de una Función Seno.

4.1.4. Control de Seguimiento de Trayectoria con Retardo

De igual manera que en la sección del robot móvil individual, para realizar un control de trayectoria más apegado a la realidad se debe tomar en cuenta el retardo inducido por el cálculo de las posiciones de cada uno de los robots por medio de la cámara utilizada como sensor. La Figura 4.8 muestra el diagrama de bloques de la implementación del controlador retardado.

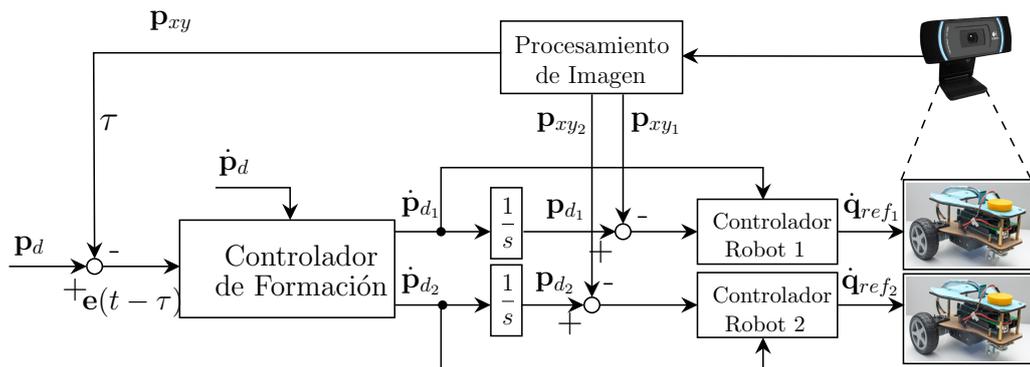


Figura 4.8: Esquema del CSTRR.

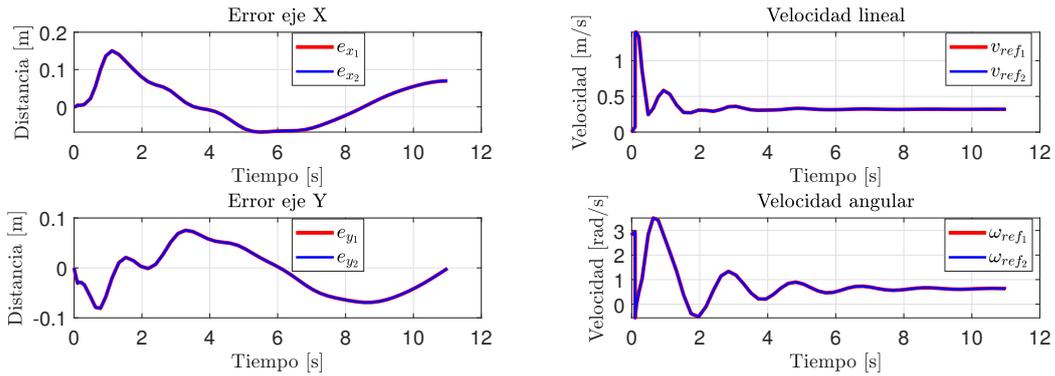
Implementando el diagrama de bloques mostrado en la figura anterior para el Controlador de Formación se utilizó la ley de control de seguimiento de trayectoria (3.66) pero con $\mathbf{K} \in \mathbb{R}^{4 \times 4}$ y para el Controlador Robot se utilizó la ley de control (4.10) con $i \in \{1, 2\}$. Como se demostró en secciones anteriores, las leyes de control utilizadas tienen un comportamiento asintóticamente estable.

Círculo

Para la simulación se tomaron las siguientes condiciones iniciales $x_{0_1} = 0$, $y_{0_1} = 0$, $\varphi_{0_1} = 0^\circ$, $x_{0_2} = 0.25$, $y_{0_2} = 0$ y $\varphi_{0_2} = 0^\circ$, y se deseó que la formación siguiera la trayectoria deseada representada por una función seno la cual está parametrizada por:

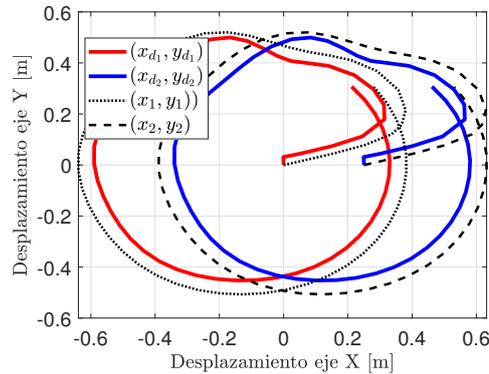
$$\begin{aligned} x_d(t) &= 0.5 \cos\left(\frac{\pi}{5}t\right) \\ y_d(t) &= 0.5 \sin\left(\frac{\pi}{5}t\right) \end{aligned}$$

con una distancia entre robots de $D_d = 0.25$, ángulo de orientación en la formación de $\Phi_{C_d} = 0^\circ$ y con unas matrices de ganancia $\mathbf{K} = \text{diag}\left(\frac{e^{-1}}{0.1}, \frac{e^{-1}}{0.1}, \frac{e^{-1}}{0.1}, \frac{e^{-1}}{0.1}\right)$, $\mathbf{K}_1 = \text{diag}(0.8, 0.8)$, $\mathbf{K}_2 = \mathbf{K}_1$.



(a) Errores de posición.

(b) Velocidades.



(c) Trayectorias.

Figura 4.9: Respuestas Numéricas del CSTF2RR al Seguimiento de un Círculo.

4.2. Formación de Tres Robots

En esta sección se muestra el modelo geométrico y la cinemática de la formación de tres robots móviles de tracción diferencial. Posteriormente se presenta el diseño del control de posición y control de seguimiento de trayectoria de la formación, para así mantener la distancia y el ángulo que se forma entre cada uno de ellos.

4.2.1. Modelo Cinemático

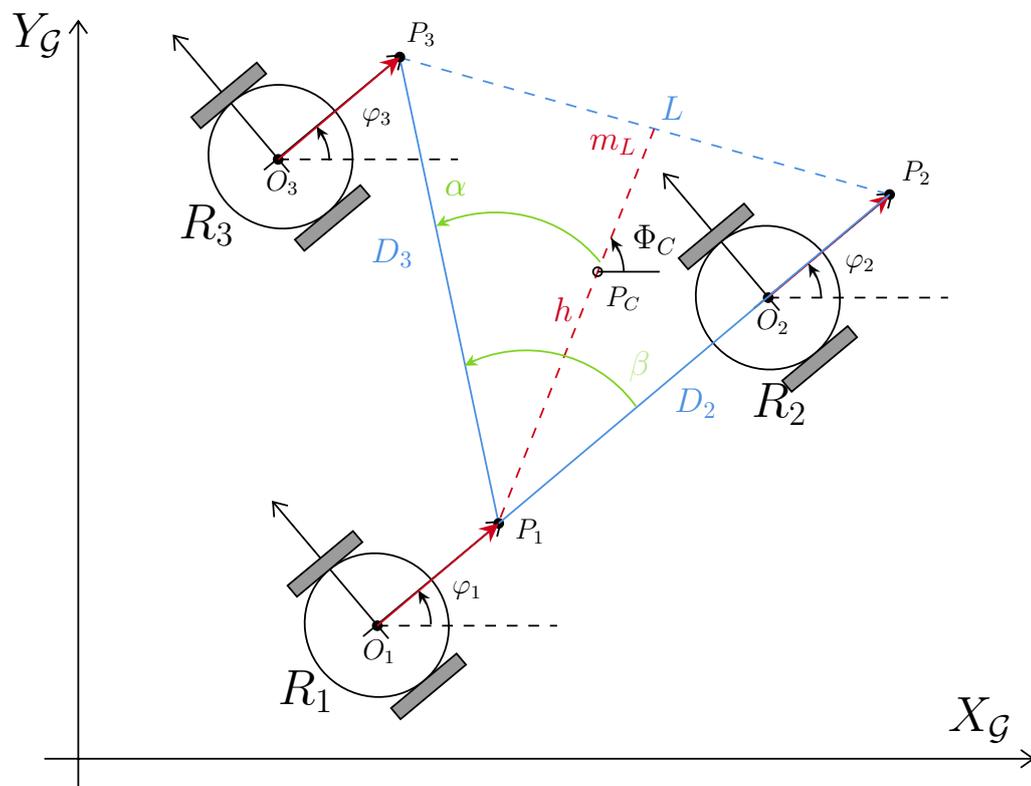


Figura 4.10: Esquema de la Formación de Tres Robots.

Donde los puntos P_i se definen como $P_i(x_i, y_i)$ y el punto central de la formación se define como $P_C(x_C, y_C)$, el cual es el centroide o baricentro del triángulo formado por los puntos P_i correspondiente a cada robot, por lo que el punto P_C es la intersección de las rectas que unen el punto P_i con el punto medio de la arista opuesta.

Las coordenadas del punto P_C definen como:

$$\begin{aligned} x_C &= \frac{x_1 + x_2 + x_3}{3} \\ y_C &= \frac{y_1 + y_2 + y_3}{3} \end{aligned} \quad (4.11)$$

El ángulo de orientación de la formación con respecto del eje X_G se calculó con la función arco tangente y se definió como:

$$\Phi_C = \tan^{-1} \left(\frac{-2y_1 + y_2 + 2y_3}{-2x_1 + x_2 + x_3} \right) \quad (4.12)$$

Para las ecuaciones de forma fue necesario determinar la distancia entre los robots, la cual es la distancia euclidiana entre los puntos P_i que representan el centro de cada uno de los robots, dichas distancias se definieron como:

$$\begin{aligned} D_2 &= \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \\ D_3 &= \sqrt{(x_1 - x_3)^2 + (y_1 - y_3)^2} \\ L &= \sqrt{(x_2 - x_3)^2 + (y_2 - y_3)^2} \end{aligned} \quad (4.13)$$

También se debió determinar el ángulo formado entre las distancias D_2 y D_3 el cual se definió como β :

$$\beta = \cos^{-1} \left(\frac{D_2^2 + D_3^2 + L^2}{2D_2D_3} \right) \quad (4.14)$$

Teniendo esto se representó el modelo geométrico en forma matricial:

$$\begin{bmatrix} x_C \\ y_C \\ \Phi_C \\ D_2 \\ D_3 \\ \beta \end{bmatrix} = \begin{bmatrix} \frac{x_1 + x_2 + x_3}{3} \\ \frac{y_1 + y_2 + y_3}{3} \\ \tan^{-1} \left(\frac{-2y_1 + y_2 + y_3}{-2x_1 + x_2 + x_3} \right) \\ \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \\ \sqrt{(x_1 - x_3)^2 + (y_1 - y_3)^2} \\ \cos^{-1} \left(\frac{D_2^2 + D_3^2 + L^2}{2D_2D_3} \right) \end{bmatrix} \quad (4.15)$$

Con el modelo geométrico ya obtenido, se obtuvo el modelo cinemático diferencial de la formación mediante la derivadas temporales, de cada una de las

ecuaciones anteriores mediante la regla de la cadena.

Por lo que la cinemática directa diferencial en forma matricial quedó como:

$$\underbrace{\begin{bmatrix} \dot{x}_C \\ \dot{y}_C \\ \dot{\Phi}_C \\ \dot{D}2 \\ \dot{D}3 \\ \dot{\beta} \end{bmatrix}}_{\dot{\mathbf{p}}} = \underbrace{\begin{bmatrix} \frac{\partial x_C}{\partial x_1} & \frac{\partial x_C}{\partial y_1} & \frac{\partial x_C}{\partial x_2} & \frac{\partial x_C}{\partial y_2} & \frac{\partial x_C}{\partial x_3} & \frac{\partial x_C}{\partial y_3} \\ \frac{\partial y_C}{\partial x_1} & \frac{\partial y_C}{\partial y_1} & \frac{\partial y_C}{\partial x_2} & \frac{\partial y_C}{\partial y_2} & \frac{\partial y_C}{\partial x_3} & \frac{\partial y_C}{\partial y_3} \\ \frac{\partial \Phi_C}{\partial x_1} & \frac{\partial \Phi_C}{\partial y_1} & \frac{\partial \Phi_C}{\partial x_2} & \frac{\partial \Phi_C}{\partial y_2} & \frac{\partial \Phi_C}{\partial x_3} & \frac{\partial \Phi_C}{\partial y_3} \\ \frac{\partial D2}{\partial x_1} & \frac{\partial D2}{\partial y_1} & \frac{\partial D2}{\partial x_2} & \frac{\partial D2}{\partial y_2} & \frac{\partial D2}{\partial x_3} & \frac{\partial D2}{\partial y_3} \\ \frac{\partial D3}{\partial x_1} & \frac{\partial D3}{\partial y_1} & \frac{\partial D3}{\partial x_2} & \frac{\partial D3}{\partial y_2} & \frac{\partial D3}{\partial x_3} & \frac{\partial D3}{\partial y_3} \\ \frac{\partial \beta}{\partial x_1} & \frac{\partial \beta}{\partial y_1} & \frac{\partial \beta}{\partial x_2} & \frac{\partial \beta}{\partial y_2} & \frac{\partial \beta}{\partial x_3} & \frac{\partial \beta}{\partial y_3} \end{bmatrix}}_{\mathbf{J}} \underbrace{\begin{bmatrix} \dot{x}_1 \\ \dot{y}_1 \\ \dot{x}_2 \\ \dot{y}_2 \\ \dot{x}_3 \\ \dot{y}_3 \end{bmatrix}}_{\dot{\mathbf{q}}} \quad (4.16)$$

donde:

$$\begin{aligned} \frac{\partial x_C}{\partial x_1} &= \frac{1}{3}, & \frac{\partial x_C}{\partial y_1} &= 0, & \frac{\partial x_C}{\partial x_2} &= \frac{1}{3}, & \frac{\partial x_C}{\partial y_2} &= 0, & \frac{\partial x_C}{\partial x_3} &= \frac{1}{3}, & \frac{\partial x_C}{\partial y_3} &= 0, \\ \frac{\partial y_C}{\partial x_1} &= 0, & \frac{\partial y_C}{\partial y_1} &= \frac{1}{3}, & \frac{\partial y_C}{\partial x_2} &= 0, & \frac{\partial y_C}{\partial y_2} &= \frac{1}{3}, & \frac{\partial y_C}{\partial x_3} &= 0, & \frac{\partial y_C}{\partial y_3} &= \frac{1}{3}, \\ \frac{\partial \Phi_c}{\partial x_1} &= \frac{2(-2y_1 + y_2 + 2y_3)}{(-2x_1 + x_2 + x_3)^2 + (-2y_1 + y_2 + 2y_3)^2}, \\ \frac{\partial \Phi_c}{\partial y_1} &= \frac{2(2x_1 - x_2 - x_3)}{(-2x_1 + x_2 + x_3)^2 + (-2y_1 + y_2 + 2y_3)^2}, \\ \frac{\partial \Phi_c}{\partial x_2} &= \frac{-2y_1 + y_2 + 2y_3}{(-2x_1 + x_2 + x_3)^2 + (-2y_1 + y_2 + 2y_3)^2}, \\ \frac{\partial \Phi_c}{\partial y_2} &= \frac{-2x_1 + x_2 + x_3}{(-2x_1 + x_2 + x_3)^2 + (-2y_1 + y_2 + 2y_3)^2}, \\ \frac{\partial \Phi_c}{\partial x_3} &= \frac{-2y_1 + y_2 + 2y_3}{(-2x_1 + x_2 + x_3)^2 + (-2y_1 + y_2 + 2y_3)^2}, \\ \frac{\partial \Phi_c}{\partial y_3} &= \frac{2(-2x_1 + x_2 + x_3)}{(-2x_1 + x_2 + x_3)^2 + (-2y_1 + y_2 + 2y_3)^2}, \\ \frac{\partial D2}{\partial x_1} &= \frac{x_1 - x_2}{\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}}, & \frac{\partial D2}{\partial y_1} &= \frac{y_1 - y_2}{\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}}, \end{aligned}$$

$$\begin{aligned} \frac{\partial D2}{\partial x_2} &= -\frac{x_1 - x_2}{\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}}, & \frac{\partial D2}{\partial y_2} &= -\frac{y_1 - y_2}{\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}}, \\ & \frac{\partial D2}{\partial x_3} = 0, & \frac{\partial D2}{\partial y_3} &= 0, \\ \frac{\partial D3}{\partial x_1} &= \frac{x_1 - x_3}{\sqrt{(x_1 - x_3)^2 + (y_1 - y_3)^2}}, & \frac{\partial D3}{\partial y_1} &= \frac{y_1 - y_3}{\sqrt{(x_1 - x_3)^2 + (y_1 - y_3)^2}}, \\ & \frac{\partial D3}{\partial x_2} = 0, & \frac{\partial D3}{\partial y_2} &= 0, \\ \frac{\partial D3}{\partial x_3} &= -\frac{x_1 - x_3}{\sqrt{(x_1 - x_3)^2 + (y_1 - y_3)^2}}, & \frac{\partial D3}{\partial y_3} &= -\frac{y_1 - y_3}{\sqrt{(x_1 - x_3)^2 + (y_1 - y_3)^2}}, \\ \frac{\partial \beta}{\partial x_1} &= \frac{\partial}{\partial x_1} \left(\cos^{-1} \left(\frac{(x_1 - x_2)^2 + (x_1 - x_3)^2 + (x_2 - x_3)^2 + (y_1 - y_2)^2 + (y_1 - y_3)^2 + (y_2 - y_3)^2}{2\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \sqrt{(x_1 - x_3)^2 + (y_1 - y_3)^2}} \right) \right), \\ \frac{\partial \beta}{\partial y_1} &= \frac{\partial}{\partial y_1} \left(\cos^{-1} \left(\frac{(x_1 - x_2)^2 + (x_1 - x_3)^2 + (x_2 - x_3)^2 + (y_1 - y_2)^2 + (y_1 - y_3)^2 + (y_2 - y_3)^2}{2\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \sqrt{(x_1 - x_3)^2 + (y_1 - y_3)^2}} \right) \right), \\ \frac{\partial \beta}{\partial x_2} &= \frac{\partial}{\partial x_2} \left(\cos^{-1} \left(\frac{(x_1 - x_2)^2 + (x_1 - x_3)^2 + (x_2 - x_3)^2 + (y_1 - y_2)^2 + (y_1 - y_3)^2 + (y_2 - y_3)^2}{2\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \sqrt{(x_1 - x_3)^2 + (y_1 - y_3)^2}} \right) \right), \\ \frac{\partial \beta}{\partial y_2} &= \frac{\partial}{\partial y_2} \left(\cos^{-1} \left(\frac{(x_1 - x_2)^2 + (x_1 - x_3)^2 + (x_2 - x_3)^2 + (y_1 - y_2)^2 + (y_1 - y_3)^2 + (y_2 - y_3)^2}{2\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \sqrt{(x_1 - x_3)^2 + (y_1 - y_3)^2}} \right) \right), \\ \frac{\partial \beta}{\partial x_3} &= \frac{\partial}{\partial x_3} \left(\cos^{-1} \left(\frac{(x_1 - x_2)^2 + (x_1 - x_3)^2 + (x_2 - x_3)^2 + (y_1 - y_2)^2 + (y_1 - y_3)^2 + (y_2 - y_3)^2}{2\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \sqrt{(x_1 - x_3)^2 + (y_1 - y_3)^2}} \right) \right), \\ \frac{\partial \beta}{\partial y_3} &= \frac{\partial}{\partial y_3} \left(\cos^{-1} \left(\frac{(x_1 - x_2)^2 + (x_1 - x_3)^2 + (x_2 - x_3)^2 + (y_1 - y_2)^2 + (y_1 - y_3)^2 + (y_2 - y_3)^2}{2\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \sqrt{(x_1 - x_3)^2 + (y_1 - y_3)^2}} \right) \right). \end{aligned}$$

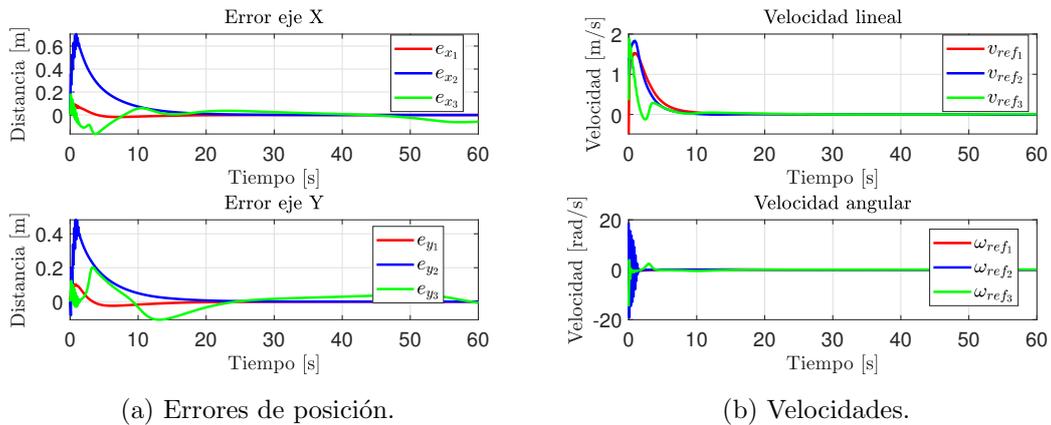
Estados: $\dot{x}_C, \dot{y}_C, \dot{\Phi}_C, \dot{D}2, \dot{D}3, \dot{\beta}$.

Entradas: $\dot{x}_1, \dot{y}_1, \dot{x}_2, \dot{y}_2, \dot{x}_3, \dot{y}_3$.

4.2.2. Control de Posición

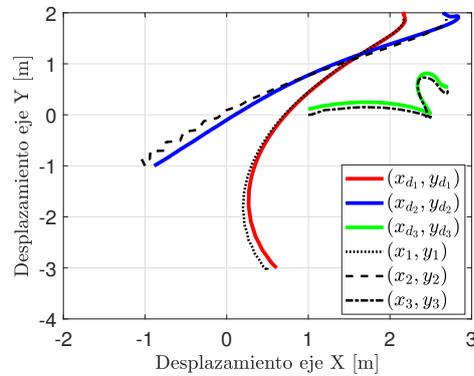
Utilizando el diagrama de la Figura 4.2 se implementó el control de posición para la formación de tres robots, se utilizaron las leyes de control mostradas en la Subsección 4.1.2, pero con $i \in \{1, 2, 3\}$. El análisis de estabilidad se realizó anteriormente, por lo que se realizaron una serie de simulaciones para una validación numérica.

1. Se realizó una simulación para condiciones iniciales $x_{0_1} = -0.5$, $y_{0_1} = 3$, $\varphi_{0_1} = 0^\circ$, $x_{0_2} = -1$, $y_{0_2} = -1$ y $\varphi_{0_2} = 0^\circ$, $x_{0_3} = 1$, $y_{0_3} = 0$ y $\varphi_{0_3} = 90^\circ$, para mover a las coordenadas deseadas el centro de la formación $x_d = 2.5$, $y_d = 1.5$, ángulo de orientación en la formación de $\Phi_{C_d} = 30^\circ$, una distancia entre los robots de $D_{2_d} = 0.5$, $D_{3_d} = 0.5$ y con un ángulo entre los robots de $\beta_d = 60^\circ$. Las matrices de ganancia que se utilizaron fueron $\mathbf{K} = \text{diag}(0.4, 0.4, 0.4, 0.4, 0.4, 0.4)$, $\mathbf{K}_1 = \text{diag}(0.2, 0.2)$, $\mathbf{K}_2 = \mathbf{K}_1$, $\mathbf{K}_3 = \mathbf{K}_2$.



(a) Errores de posición.

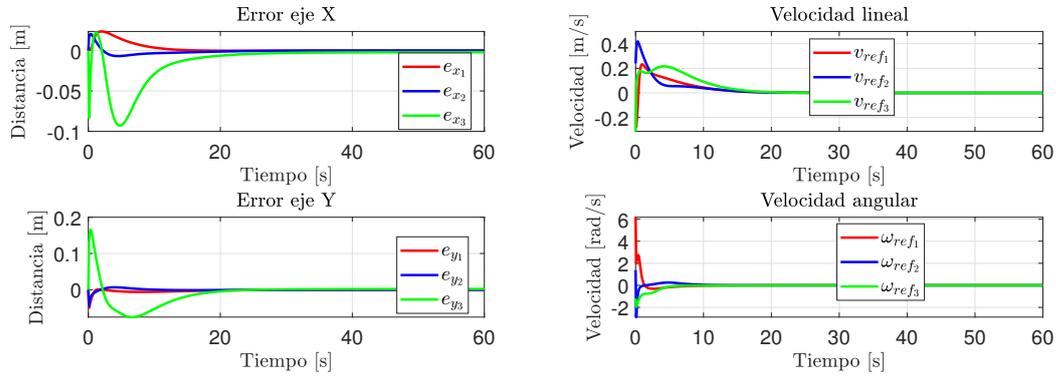
(b) Velocidades.



(c) Trayectoria.

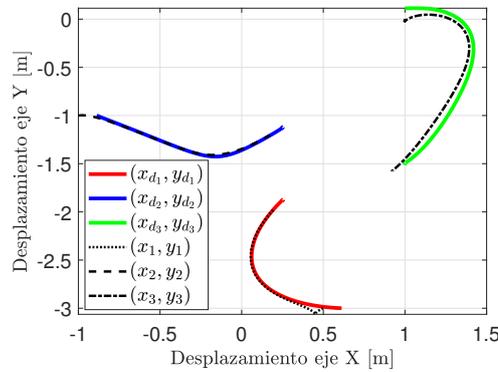
Figura 4.11: Respuestas Numéricas No. 1 del CPF3R.

2. Se realizó una simulación para condiciones iniciales $x_{0_1} = -0.5$, $y_{0_1} = 3$, $\varphi_{0_1} = 0^\circ$, $x_{0_2} = -1$, $y_{0_2} = -1$ y $\varphi_{0_2} = 0^\circ$, $x_{0_3} = 1$, $y_{0_3} = 0$ y $\varphi_{0_3} = 90^\circ$, para mover a las coordenadas deseadas el centro de la formación $x_d = 0.5$, $y_d = -1.5$, ángulo de orientación en la formación de $\Phi_{C_d} = 90^\circ$, una distancia entre los robots de $D_{2_d} = 0.75$, $D_{3_d} = 0.75$ y con un ángulo entre los robots de $\beta_d = 60^\circ$. Las matrices de ganancia que se utilizaron fueron $\mathbf{K} = \text{diag}(0.4, 0.4, 0.4, 0.4, 0.4, 0.4)$, $\mathbf{K}_1 = \text{diag}(0.2, 0.2)$, $\mathbf{K}_2 = \mathbf{K}_1$, $\mathbf{K}_3 = \mathbf{K}_2$.



(a) Errores de posición.

(b) Velocidades.



(c) Trayectoria.

Figura 4.12: Respuestas Numéricas No. 2 del CPF3R.

4.2.3. Control de Seguimiento de Trayectoria

Utilizando el diagrama de la Figura 4.5 se diseñó el Control de Seguimiento de Trayectoria para la Formación de tres Robots (CSTF3R), al nivel de formación se utilizó la ley de control (3.37) y al nivel del robot se utilizó la ley mostrada en (4.10), pero con $i \in \{1, 2, 3\}$.

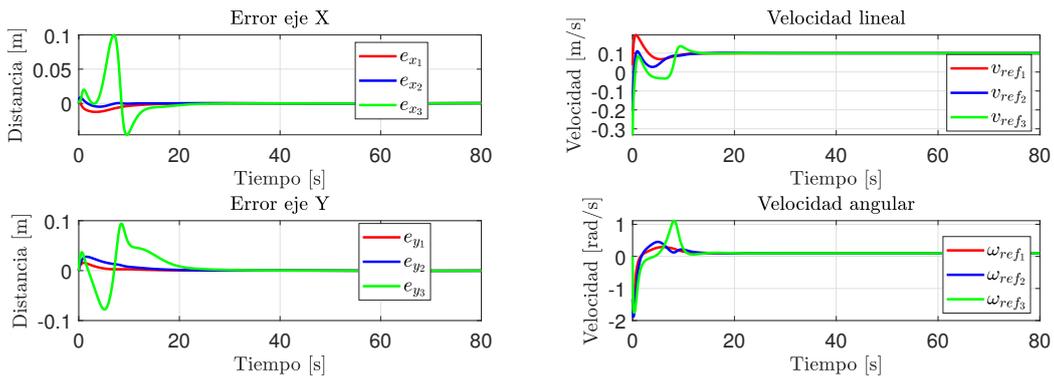
Círculo

Para la simulación se tomaron las siguientes condiciones iniciales $x_{0_1} = 0$, $y_{0_1} = 0$, $\varphi_{0_1} = 90^\circ$, $x_{0_2} = 0.5$, $y_{0_2} = 0$, $\varphi_{0_2} = 90^\circ$, $x_{0_3} = 1$, $y_{0_3} = 1$, $\varphi_{0_3} = 90^\circ$, y se deseó que la formación siguiera la trayectoria de un círculo la cual está parametrizada por:

$$x_d(t) = \cos(0.1t)$$

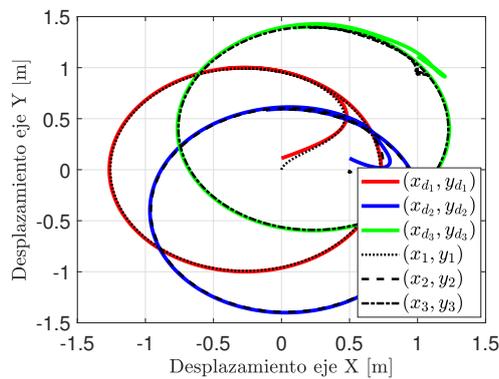
$$y_d(t) = \sin(0.1t)$$

con una distancia entre robots de $D_{2_d} = 0.5$, $D_{3_d} = 0.5$, ángulo entre los robots de $\beta_d = 60^\circ$, ángulo de orientación en la formación de $\text{Phi}_d = 0^\circ$ y con unas matrices de ganancia $\mathbf{K} = \text{diag}(0.4, 0.4, 0.4, 0.4, 0.4, 0.4)$, $\mathbf{K}_1 = \text{diag}(0.2, 0.2)$, $\mathbf{K}_2 = \mathbf{K}_1$, $\mathbf{K}_3 = \mathbf{K}_2$.



(a) Errores de posición.

(b) Velocidades.



(c) Trayectorias.

Figura 4.13: Respuestas Numéricas del CSTF3R al Seguimiento de un Círculo.

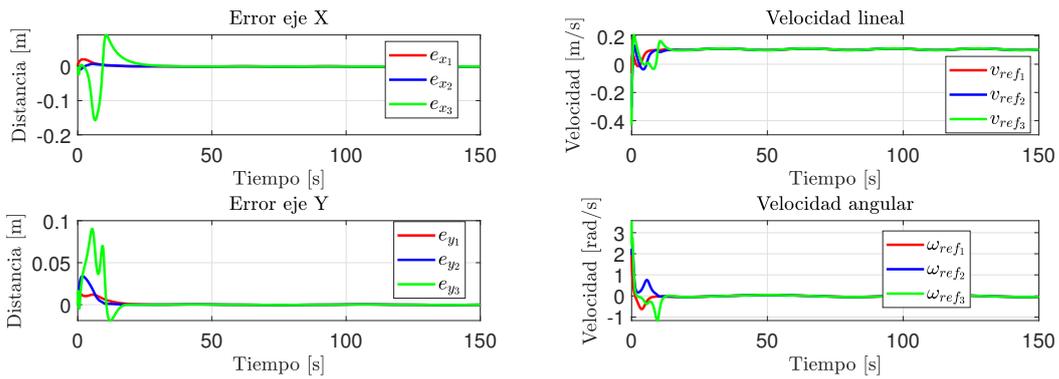
Función seno

Para la simulación se tomaron las siguientes condiciones iniciales $x_{0_1} = 0$, $y_{0_1} = 0$, $\varphi_{0_1} = 90^\circ$, $x_{0_2} = 0.5$, $y_{0_2} = 0$, $\varphi_{0_2} = 90^\circ$, $x_{0_3} = 1$, $y_{0_3} = 1$, $\varphi_{0_3} = 90^\circ$, y se deseó que la formación siguiera la trayectoria de una función seno la cual está parametrizada por:

$$x_d(t) = 0.1t$$

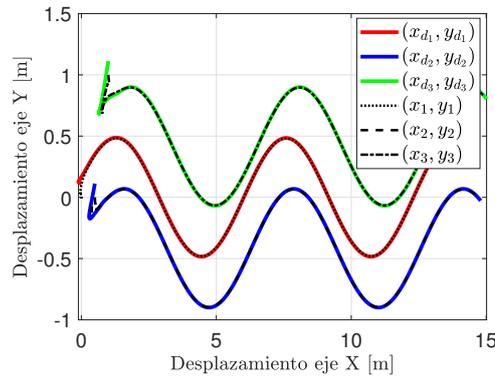
$$y_d(t) = 0.5 \text{sen}(0.1t)$$

con una distancia entre robots de $D_{2_d} = 0.5$, $D_{3_d} = 0.5$, ángulo entre los robots de $\beta_d = 60^\circ$, ángulo de orientación en la formación de $\text{Phi}_d = 0^\circ$ y con unas matrices de ganancia $\mathbf{K} = \text{diag}(0.4, 0.4, 0.4, 0.4, 0.4, 0.4)$, $\mathbf{K}_1 = \text{diag}(0.2, 0.2)$, $\mathbf{K}_2 = \mathbf{K}_1$, $\mathbf{K}_3 = \mathbf{K}_2$.



(a) Errores de posición.

(b) Velocidades.



(c) Trayectorias.

Figura 4.14: Respuestas Numéricas del CSTF3R al Seguimiento de una Función Seno.

Validación Experimental

5.1. Sistema de Adquisición de Datos

Teniendo la programación de la tarjeta de control lista para mandar las velocidades actuales del robot (v_{Robot} y ω_{Robot}) y también para recibir las señales de control o velocidades de referencia (v_{ref} y ω_{ref}), se procedió a realizar la programación en Python del algoritmo de control deseado (CPO, CPOGA, CST, CSC, etc) y del tracking de los robots por medio de una cámara.

En la Figura 5.1 se muestra el diagrama de comunicaciones involucradas en las pruebas experimentales. Se tiene un control central el cual se calcula en la PC, con ayuda de la cámara se realimentan las posiciones actuales y con esto obtener las velocidades de referencia idóneas para el control de velocidad del robot programado en el Arduino MEGA®.

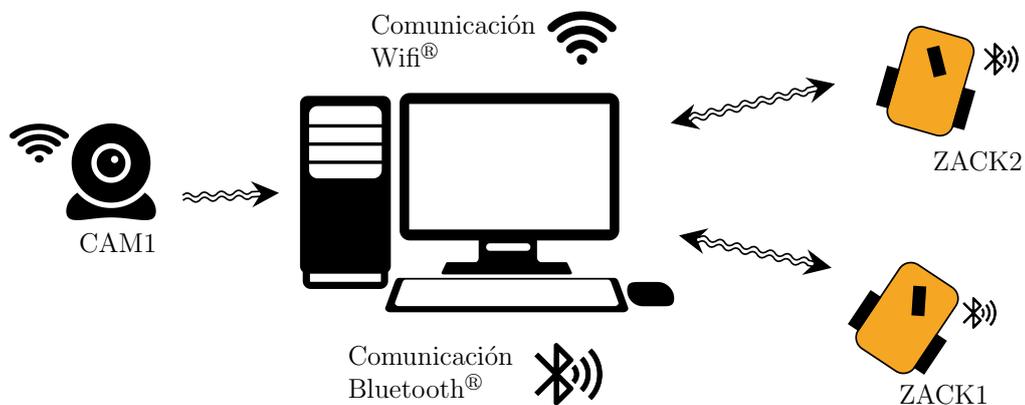


Figura 5.1: Diagrama de Comunicaciones.

Se puede observar en la Figura 5.1 que la comunicación de los dispositivos fue de manera inalámbrica: la cámara por conexión Wifi (direccional) y los robots por conexión Bluetooth (bidireccional).

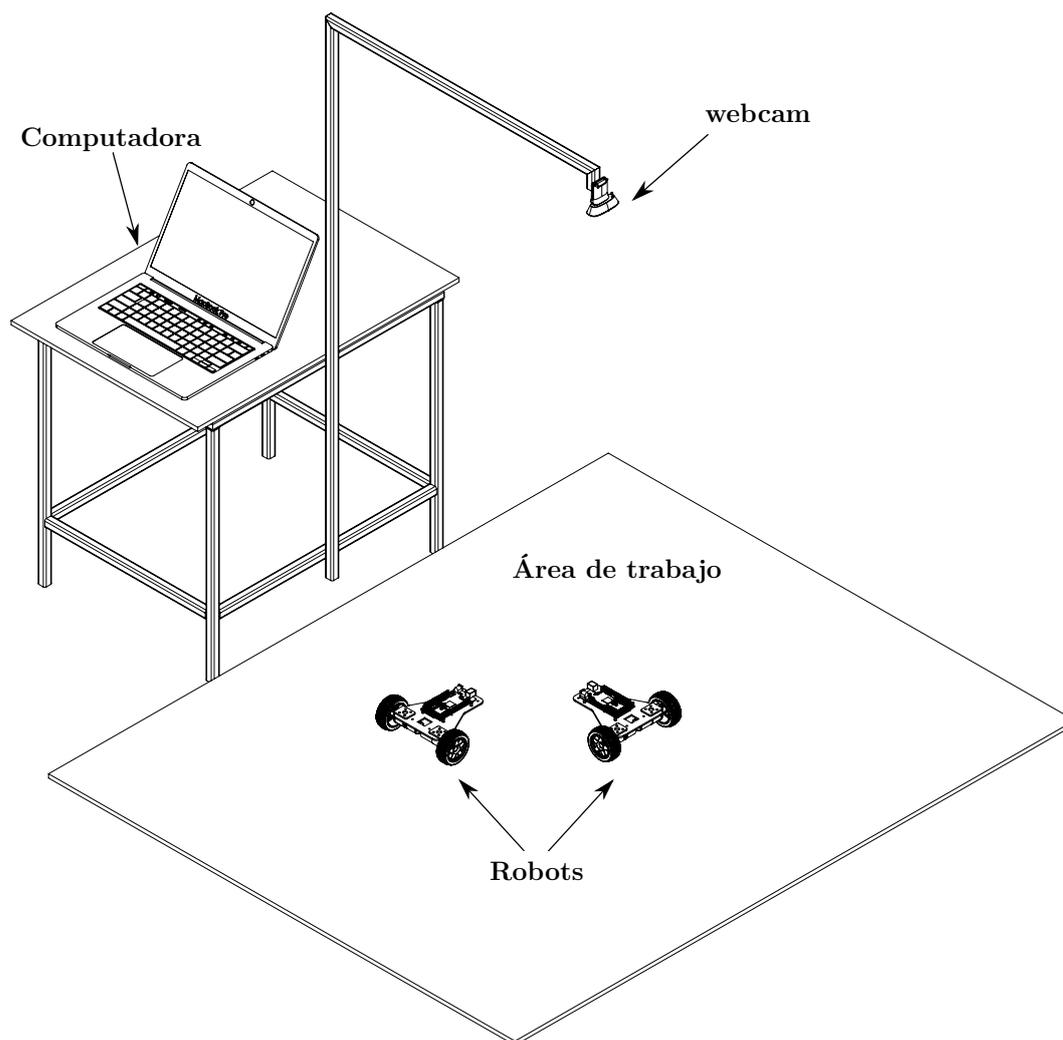


Figura 5.2: Área de Trabajo en los Experimentos.

Sistema de Visión Artificial

Para el diseño del Sistema de Visión Artificial (SVA) se utilizó la biblioteca de acceso libre OpenCV la cual fue desarrollada por Intel[®]. OpenCV significa Open Computer Vision (Visión Artificial Abierta). Esta librería está desarrollada en el lenguaje orientado a objetos *C++*, aunque tiene conectores para otros lenguajes como lo es *Python*.

La versión de OpenCV utilizada es la 3.4.9 y la cámara utilizada durante las pruebas experimentales es de un celular MOTO G8 Power, la resolución configurada del video es 640x480 pixeles a 60 FPS.

Por lo tanto, en la Figura 5.3 se muestra como se intercambió la información

entre la Computadora, la Cámara y los Robots, los lenguajes de programación mencionados y los programas utilizados.

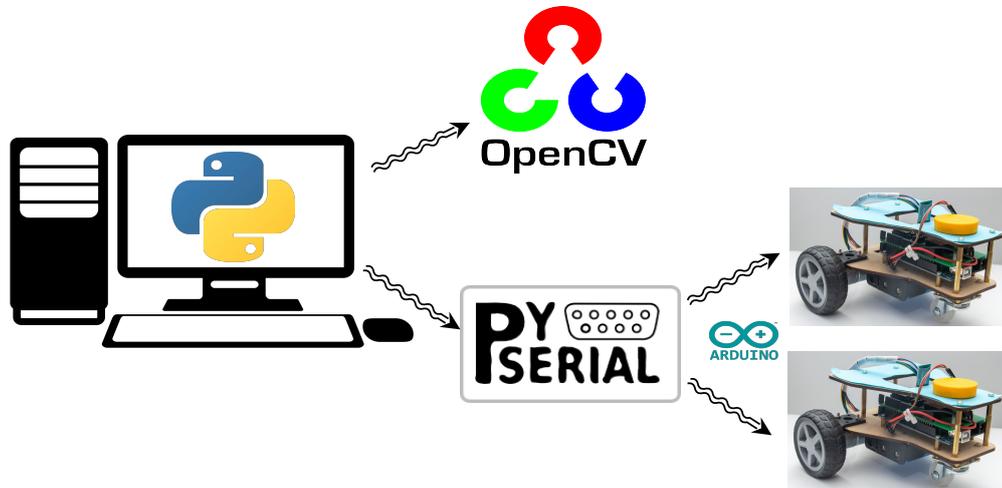


Figura 5.3: Programas Utilizados en los Experimentos.

5.2. Pruebas Experimentales de un Robot

Posteriormente ya con el SAD terminando, se realizaron pruebas experimentales para las diferentes leyes de control diseñadas y validadas numéricamente en la Sección 3.6 tales resultados se muestran a continuación. En las Figuras posteriores se muestran las respuestas numéricas (color rojo) y las respuestas experimentales (color negro) donde es fácil concluir que tienen la misma tendencia.

5.2.1. Cinemática de un Punto Central del Robot

La prueba para comprobar la cinemática directa fue tomar el robot en condiciones iniciales iguales a cero y poner una $v_{ref} = 0.5 \cdot \sin(0.1 \cdot t)$ m/s y $\omega_{ref} = 0.8$ rad/s, por 15 segundos de simulación, los resultados obtenidos fueron los siguientes:

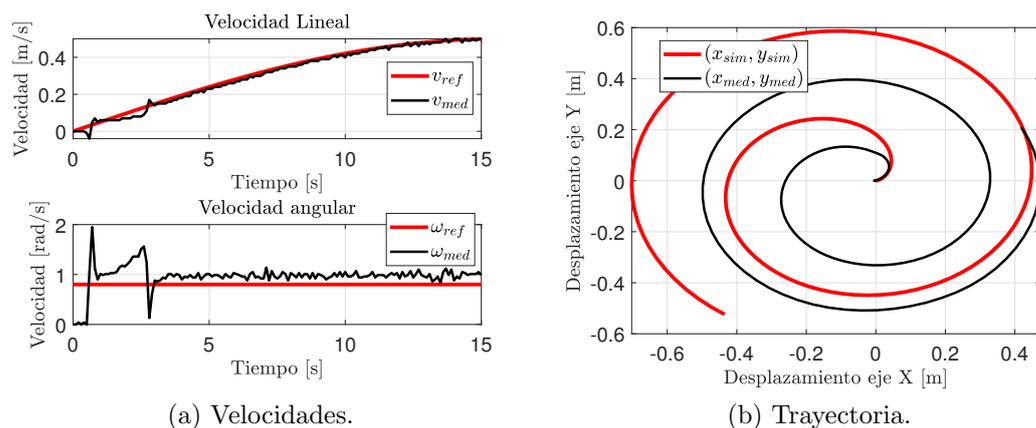


Figura 5.4: Respuesta Experimental del Robot No. 1 a la CPC.

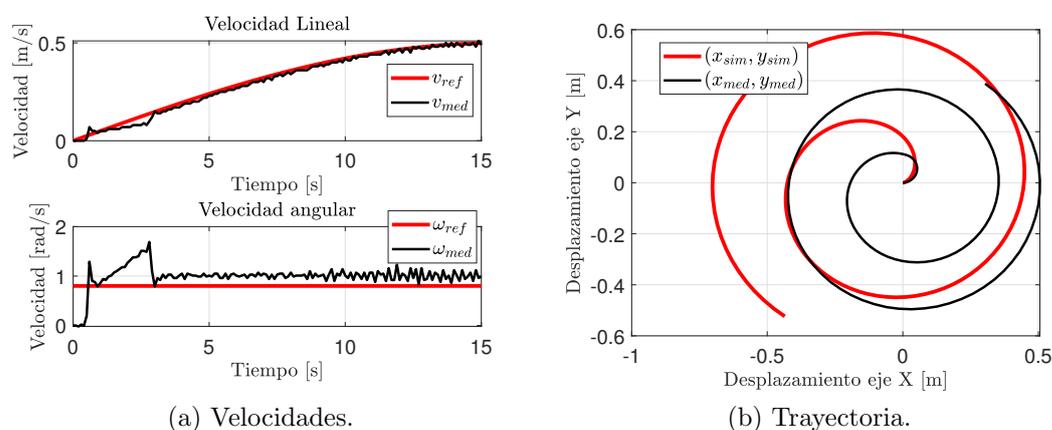


Figura 5.5: Respuesta Experimental del Robot No. 2 a la CPC.

5.2.2. Cinemática de un Punto Descentralizado del Robot

La prueba para comprobar la cinemática directa fue tomar el robot en condiciones iniciales iguales a cero y poner una $v_{ref} = 0.5 \cdot \text{sen}(0.1 \cdot t)$ m/s y $\omega_{ref} = 0.8$ rad/s, por 15 segundos de simulación, los resultados obtenidos fueron los siguientes:

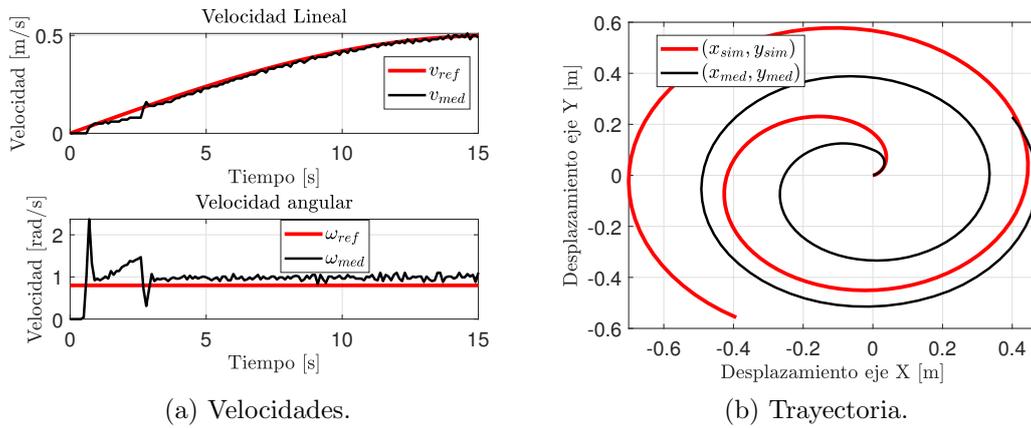


Figura 5.6: Respuesta Experimental del Robot No. 1 a la CPD.

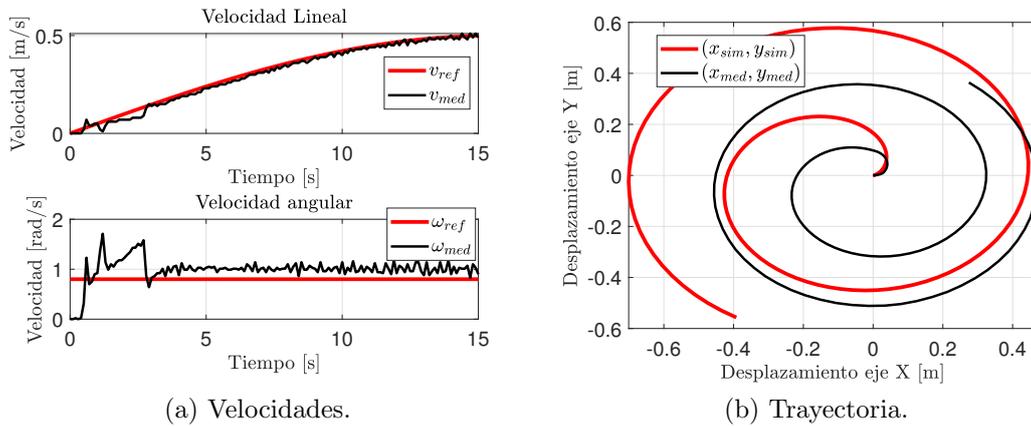
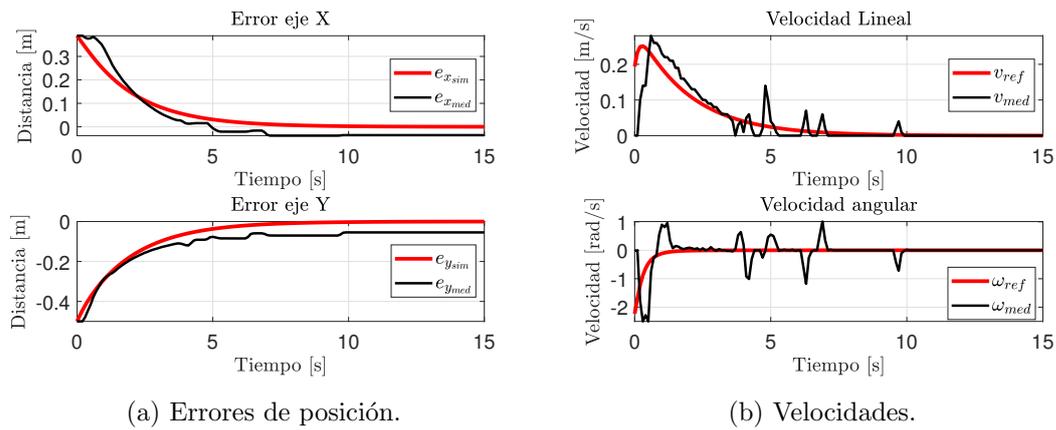


Figura 5.7: Respuesta Experimental del Robot No. 2 a la CPD.

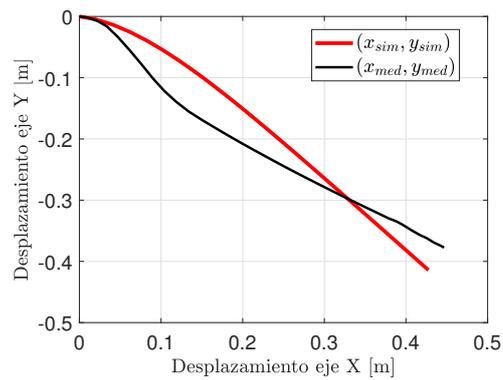
5.2.3. Control de Posición sin Orientación

La prueba para comprobar experimentalmente el algoritmo de control fue tomar el robot con condiciones iguales a cero, y la posiciones deseadas $x_d = 0.5$ m y $y_d = -0.5$ m, los resultados fueron los siguientes:



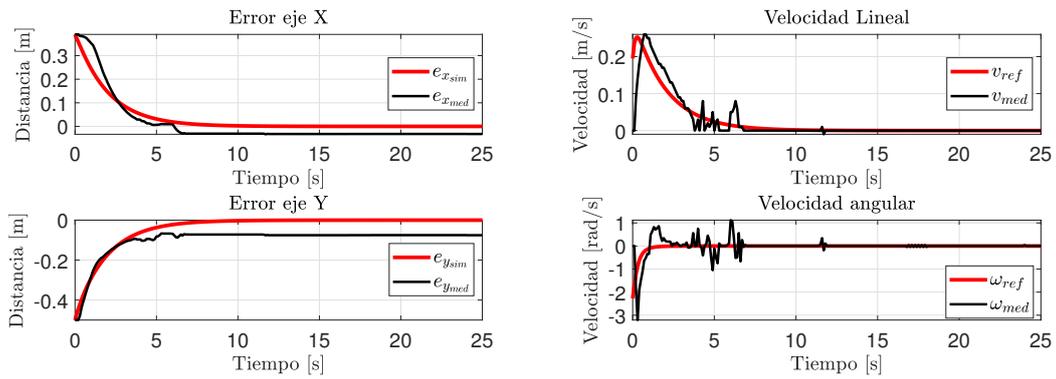
(a) Errores de posición.

(b) Velocidades.



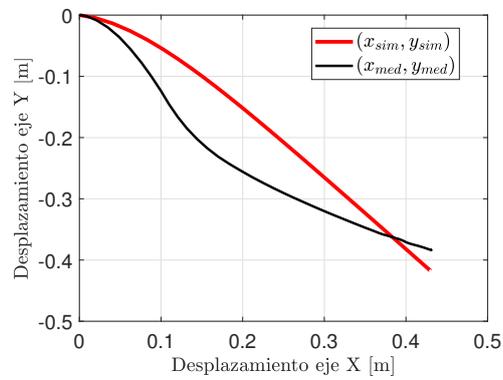
(c) Trayectoria.

Figura 5.8: Respuesta Experimental del Robot No. 1 al CPO.



(a) Errores de posición.

(b) Velocidades.



(c) Trayectoria.

Figura 5.9: Respuesta Experimental del Robot No. 2 al CPO.

5.2.4. Control de Posición sin Orientación con Ganancia Adaptativa

La prueba para comprobar experimentalmente el algoritmo de control fue tomar el robot con condiciones iguales a cero, y la posiciones deseadas $x_d = 0.5$ m y $y_d = -0.5$ m, los resultados fueron los siguientes:

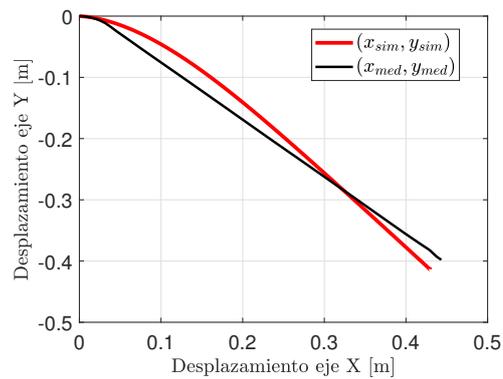
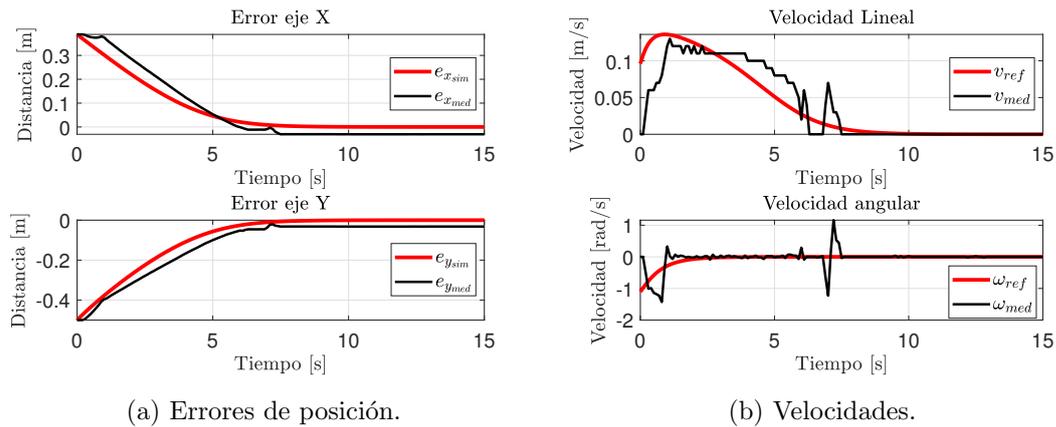


Figura 5.10: Respuesta Experimental del Robot No. 1 al CPOGA.

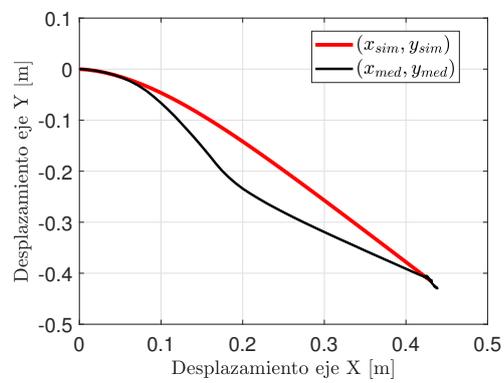
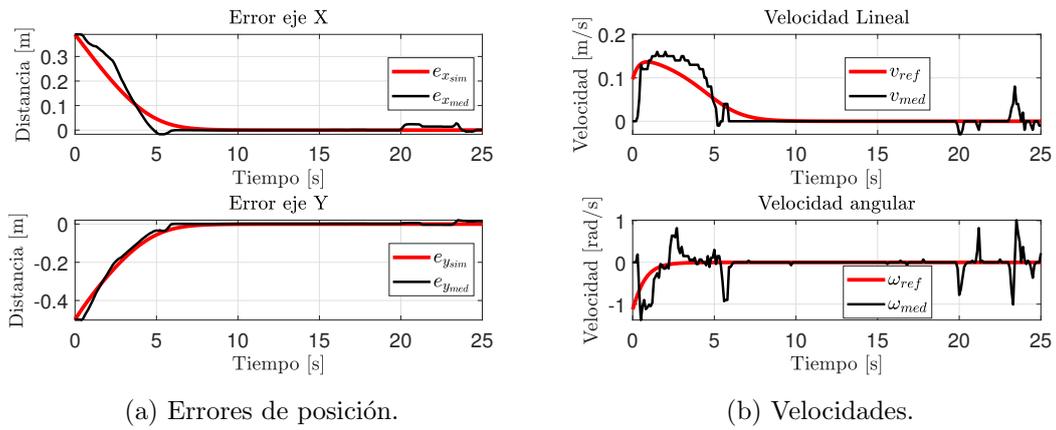


Figura 5.11: Respuesta Experimental del Robot No. 2 al CPOGA.

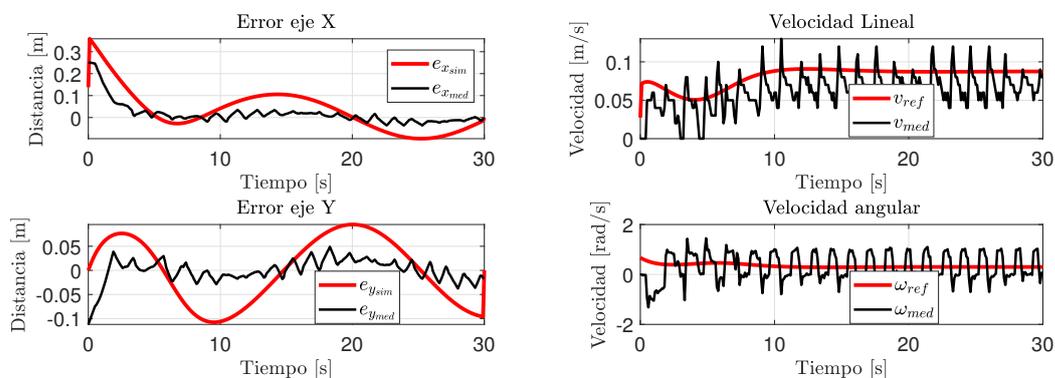
5.2.5. Control de Seguimiento de Trayectoria

Para comprobar experimentalmente el algoritmo de control se consideró el robot con condiciones iguales a cero y la trayectoria deseada (círculo de radio 0.5 m) se representa por una curva parametrizada en el tiempo por:

$$x_d(t) = 0.25 \cos(0.3t)$$

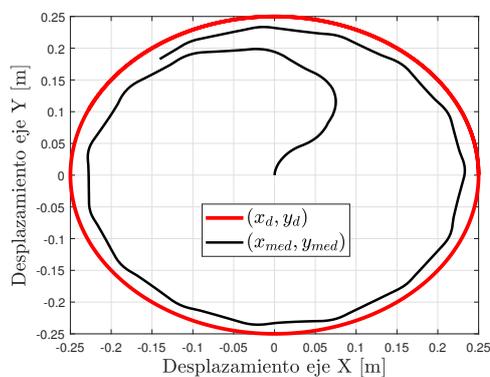
$$y_d(t) = 0.25 \sin(0.3t)$$

Los resultados son los siguientes.



(a) Errores de posición.

(b) Velocidades.



(c) Trayectoria.

Figura 5.12: Respuesta Experimental del Robot No. 1 al CST.

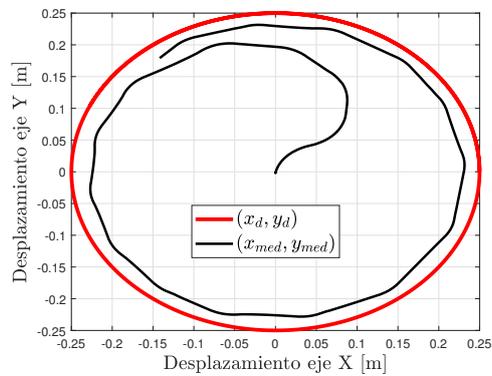
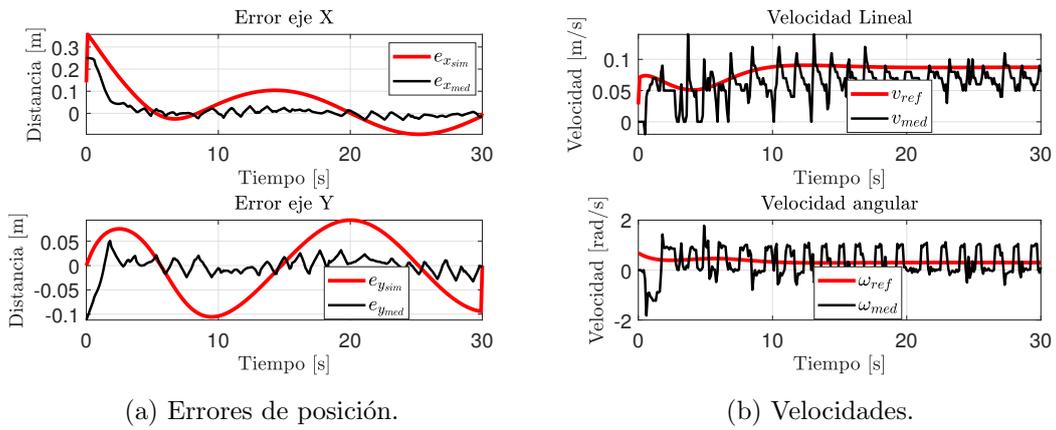


Figura 5.13: Respuesta Experimental del Robot No. 2 al CST

5.3. Pruebas Experimentales de la Formación de Dos Robots

5.3.1. Trayectoria de un Círculo

El experimento que se realizó fue con condiciones iniciales iguales a $(x_{0_1}, y_{0_1}, \varphi_{0_1}) = (0, 0, 90^\circ)$ y $(x_{0_2}, y_{0_2}, \varphi_{0_2}) = (0.25, 0, 90^\circ)$, donde la tarea de control es el de seguimiento de trayectoria parametrizada por las ecuaciones:

$$x_d(t) = 0.25 \cos(0.3t)$$

$$y_d(t) = 0.25 \sin(0.3t)$$

Se desea que la formación tenga las variables $\Phi_{C_d} = 0^\circ$, $D_d = 0.25$, con matrices de ganancia $\mathbf{K} = \text{diag}(1.3, 1.3, 1.3, 1.3)$, $\mathbf{K}_1 = \text{diag}(1, 1)$, $\mathbf{K}_2 = \mathbf{K}_1$. Los resultados obtenidos se muestran a continuación:

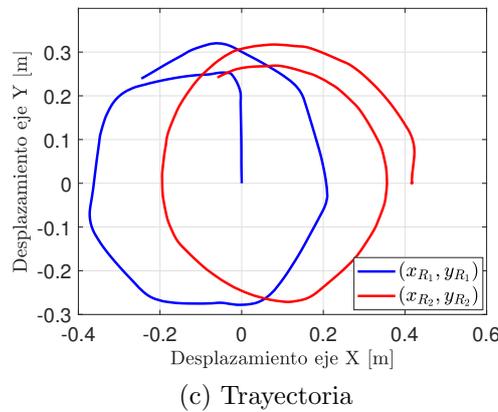
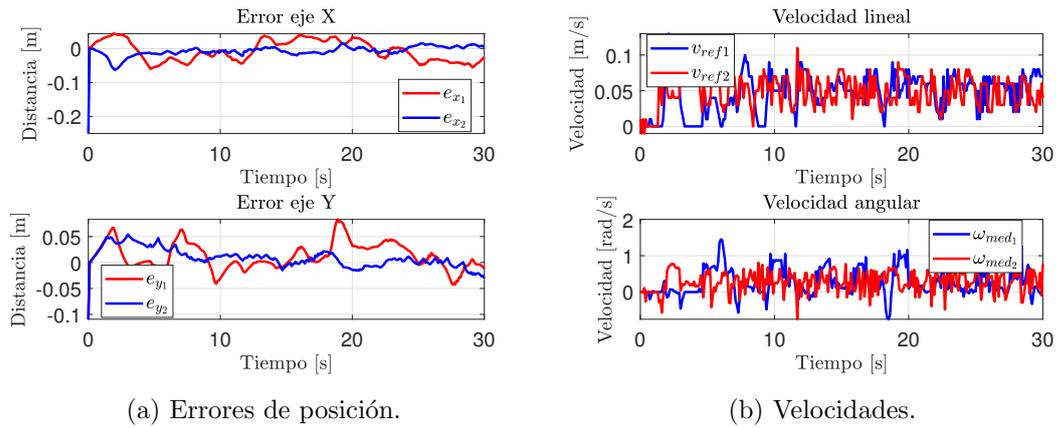


Figura 5.14: Respuestas Experimentales al CSTF2R al Seguimiento de un Círculo.

5.3.2. Trayectoria de una Función Seno

El experimento que se realizó fue con condiciones iniciales iguales a $(x_{0_1}, y_{0_1}, \varphi_{0_1}) = (0, 0, 90^\circ)$ y $(x_{0_2}, y_{0_2}, \varphi_{0_2}) = (0.5, 0, 90^\circ)$, donde la tarea de control es el de seguimiento de trayectoria parametrizada por las ecuaciones:

$$\begin{aligned}x_d(t) &= x_C(0) + v_d \cdot t \cos(\theta_d) - A \operatorname{sen}(\theta_d) \cdot \operatorname{sen}(\gamma_d t) \\y_d(t) &= y_C(0) + v_d \cdot t \operatorname{sen}(\theta_d) - A \cos(\theta_d) \cdot \operatorname{sen}(\gamma_d t)\end{aligned}$$

con $v_d = 0.08$ (velocidad lineal máxima), $\gamma_d = 0.25$ (velocidad angular máxima) $\theta_d = 0$ (ángulo de inclinación) y $A = 0.25$ (amplitud de la trayectoria sinusoidal). Se desea que la formación tenga las variables $\Phi_{C_d} = 90^\circ$, $D_d = 0.5$, con matrices de ganancia $\mathbf{K} = \operatorname{diag}(1.5, 1.5, 1.5, 1.5)$, $\mathbf{K}_1 = \operatorname{diag}(0.8, 0.8)$, $\mathbf{K}_2 = \mathbf{K}_1$. Los resultados obtenidos se muestran a continuación:

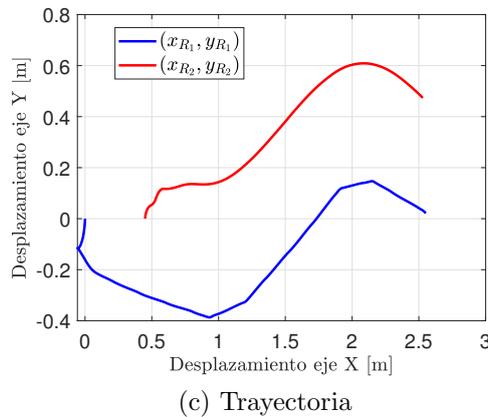
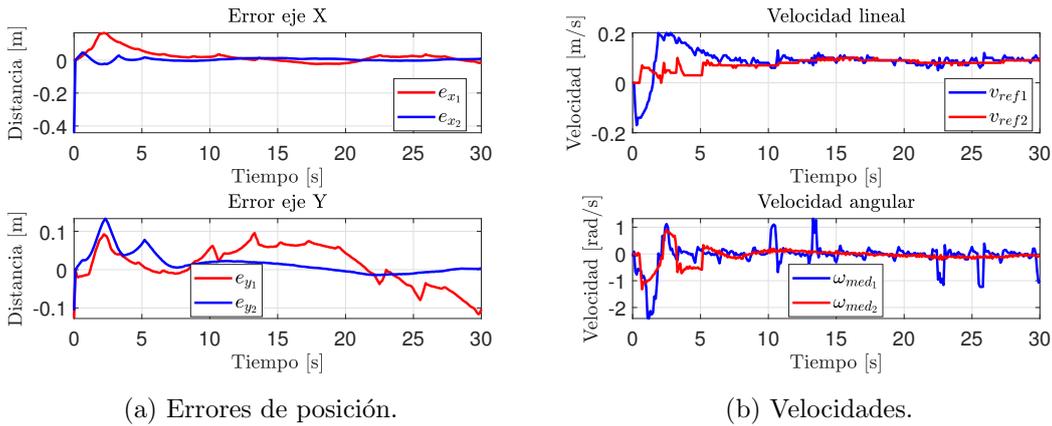


Figura 5.15: Respuestas Experimentales al CSTF2R al Seguimiento de una Función Seno.



Conclusiones y Trabajo Futuro

Conclusiones

En primer lugar, se diseñaron controladores de bajo orden tipo PI para el control de velocidad de cada uno de los motores de los robots móviles. Basándose en métricas de rendimiento en las respuestas como lo es el tiempo de asentamiento t_s y el máximo sobre impulso M_P se concluyó que el controlador denominado $PI\sigma$ es el que cumple con las necesidades, mayor velocidad y no presentar sobre impulso.

Posteriormente, se diseñaron controladores para la localización del robot móvil sin importar la orientación final, el controlador que presenta mejor desempeño es el CPOGA el cual tiene una ganancia adaptativa dependiente de la distancia euclidiana entre el punto inicial y el punto final del robot, por lo que los esfuerzos de control disminuyen en gran medida su magnitud y la velocidad de convergencia a cero de los errores es más rápida.

Para el control de formación de dos y tres robots respectivamente se realizaron las simulaciones correspondientes para poder validar las leyes de control propuestas, teniendo resultados aceptables ya que los errores oscilan entre ± 6 cm en los dos ejes del plano \mathcal{G} . También por medio de la simulación se verifica que no existe colisión entre los robots móviles.

Posteriormente se validó experimentalmente el control de formación y el seguimiento de trayectorias teniendo resultados aceptables, presentando errores que oscilan entre los ± 8 cm en los dos ejes del plano y en los experimentos tampoco se presentan colisiones entre los agentes individuales.

Por último, se realizó el análisis teórico y validación numérica de una ley de control que toma en cuenta el retardo inducido por el cálculo de la localización de los robots por medio de un sistema de visión monocular, teniendo resultados

aceptables.

Trabajo Futuro

Uno de los primeros cambios que se proponen es mejorar la plataforma experimental construida, al agregarle sensores de corriente para el cálculo de pares en los motores y con esto poder implementar leyes de control basadas en un modelo dinámico (como Euler-Lagrange o Newton-Euler). Por otro lado, cambiar el dispositivo de comunicación inalámbrica para la PC-Robot ya que la comunicación serial al momento de formar un clúster de comunicación presenta errores y retardos en la recepción de información no deseados.

El siguiente cambio es mejorar los algoritmos de tracking de los robots para el sistema de visión monocular, ya que en este proyecto se utilizaron librerías ya publicadas, por lo que se propone el trabajar los algoritmos de manera propia, el uso de filtros para la imagen y con ello mejorar el tracking al momento de realizar los experimentos. También poder controlar el ambiente del lugar de los experimentos (brillo) para poder obtener mejores resultados.

Por último continuar la línea del estudio de los retardos presentes en el experimento, ya que solo se realizó el análisis teórico quedando pendiente la experimentación del mismo. También realizar el análisis de estabilidad por medio del método Lyapunov-Krasovski.

Productos

- Artículo *Fastest with no-overshoot velocity control design of a two-differential robotic formation* sometido al congreso «Latin American Congress on Automation and Robotics 2021».



Bibliografía

- [1] DS Pavlov, AO Kasumyan, et al. Patterns and mechanisms of schooling behavior in fish: a review. *Journal of Ichthyology*, 40(2):S163, 2000.
- [2] Steven J Portugal, Tatjana Y Hubel, Johannes Fritz, Stefanie Heese, Daniela Trobe, Bernhard Voelkl, Stephen Hailes, Alan M Wilson, y James R Usherwood. Upwash exploitation and downwash avoidance by flap phasing in ibis formation flight. *Nature*, 505(7483):399–402, 2014.
- [3] John Buck y Elisabeth Buck. Toward a functional interpretation of synchronous flashing by fireflies. *The American Naturalist*, 112(985):471–492, 1978.
- [4] Masaru K Hojo, Naomi E Pierce, y Kazuki Tsuji. Lycaenid caterpillar secretions manipulate attendant ant behavior. *Current Biology*, 25(17):2260–2264, 2015.
- [5] Steven H. Strogatz. *Sync*. Hyperion, 2003.
- [6] J Alexander Fax y Richard M Murray. Graph Laplacians and stabilization of vehicle formations. *IFAC Proceedings Volumes*, 35(1):55–60, 2002.
- [7] J. Alexander Fax y Richard M. Murray. Information flow and cooperative control of vehicle formations. *IEEE Transactions on Automatic Control*, 49(9):1465–1476, 2004.
- [8] Reza Olfati-Saber, J. Alex Fax, y Richard M. Murray. Consensus and cooperation in networked multi-agent systems. *IEEE Trans. Robot. Autom.*, 20:443–455, 2004.
- [9] T. Chen, X. Liu, y W. Lu. Pinning complex networks by a single controller. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 54(6):1317–1326, 2007.
- [10] Makan Fardad, Fu Lin, y Mihailo R Jovanović. Design of optimal sparse interconnection graphs for synchronization of oscillator networks. *IEEE Transactions on Automatic Control*, 59(9):2457–2462, 2014.

- [11] Reza Olfati-Saber y Richard M Murray. Distributed structural stabilization and tracking for formations of dynamic multi-agents. In *Proceedings of the 41st IEEE Conference on Decision and Control, 2002.*, volume 1, pages 209–215. IEEE, 2002.
- [12] Aris Dimeas y Nikos Hatziargyriou. A multi-agent system for microgrids. In *Hellenic Conference on Artificial Intelligence*, pages 447–455. Springer, 2004.
- [13] Yinliang Xu y Zhicheng Li. Distributed optimal resource management based on the consensus algorithm in a microgrid. *IEEE transactions on industrial electronics*, 62(4):2584–2592, 2014.
- [14] Randal W. Beard, Jonathan Lawton, y Fred Y. Hadaegh. A coordination architecture for spacecraft formation control. *IEEE Transactions on Control Systems Technology*, 9(6):777–790, 2001.
- [15] H. Hamann. *Swarm Robotics: A Formal Approach*. Springer International Publishing, 2018.
- [16] J. Lin, A. S. Morse, y B. D. O. Anderson. The multi-agent rendezvous problem. In *42nd IEEE International Conference on Decision and Control (IEEE Cat. No.03CH37475)*, volume 2, pages 1508–1513 Vol.2, 2003.
- [17] R. Olfati-Saber. Flocking for multi-agent dynamic systems: algorithms and theory. *IEEE Transactions on Automatic Control*, 51(3):401–420, 2006.
- [18] Craig W Reynolds. Flocks, herds and schools: A distributed behavioral model. In *Proceedings of the 14th annual conference on Computer graphics and interactive techniques*, pages 25–34, 1987.
- [19] A. Ailon y I. Zohar. Control strategies for driving a group of nonholonomic kinematic mobile robots in formation along a time-parameterized path. *IEEE/ASME Transactions on Mechatronics*, 17(2):326–336, 2012.
- [20] Luca Consolini, Fabio Morbidi, Domenico Prattichizzo, y Mario Tosques. Leader–follower formation control of nonholonomic mobile robots with input constraints. *Automatica*, 44(5):1343–1349, 2008.

- [21] Gian Luca Mariottini, Fabio Morbidi, Domenico Prattichizzo, Nicholas Vander Valk, Nathan Michael, George Pappas, y Kostas Daniilidis. Vision-based localization for leader-follower formation control. *IEEE Transactions on Robotics*, 25(6):1431–1438, 2009.
- [22] Magnus Egerstedt, Xiaoming Hu, y Alexander Stotsky. Control of mobile platforms using a virtual vehicle approach. *IEEE transactions on automatic control*, 46(11):1777–1782, 2001.
- [23] T. Dierks y S. Jagannathan. Control of nonholonomic mobile robot formations: Backstepping kinematics into dynamics. In *2007 IEEE International Conference on Control Applications*, pages 94–99, 2007.
- [24] Ignacio Mas, Ognjen Petrovic, y C. Kitts. Cluster space specification and control of a 3 – robot mobile system. pages 3763 – 3768, 06 2008.
- [25] Alejandro Rodriguez-Angeles y Henk Nijmeijer. Mutual synchronization of robots via estimated state feedback: a cooperative approach. *IEEE Transactions on control systems technology*, 12(4):542–554, 2004.
- [26] Dong Sun. *Synchronization and Control of Multiagent Systems*. CRC Press, Inc., USA, 1st edition, 2010.
- [27] Dongjun Lee y Mark W Spong. Stable flocking of multiple inertial agents on balanced graphs. *IEEE transactions on automatic control*, 52(8):1469–1475, 2007.
- [28] Sara Fleury, Philippe Soueres, J-P Laumond, y Raja Chatila. Primitives for smoothing mobile robot trajectories. *IEEE transactions on robotics and automation*, 11(3):441–448, 1995.
- [29] Jean-Paul Laumond. Finding collision-free smooth trajectories for a non-holonomic mobile robot. In *IJCAI*, volume 87, pages 1120–1123, 1987.
- [30] V Munoz, Aníbal Ollero, M Prado, y A Simon. Mobile robot trajectory planning with dynamic and kinematic constraints. In *Proceedings of the 1994 IEEE International Conference on Robotics and Automation*, pages 2802–2807. IEEE, 1994.
- [31] Carl D Meyer. *Matrix analysis and applied linear algebra*, volume 71. Siam, 2000.

- [32] IV Sergienko, EF Galba, y VS Deineka. Existence and uniqueness of weighted pseudoinverse matrices and weighted normal pseudosolutions with singular weights. *Ukrainian Mathematical Journal*, 63(1):98–124, 2011.
- [33] James E Scroggs y Patrick L Odell. An alternate definition of a pseudoinverse of a matrix. *SIAM Journal on Applied Mathematics*, 14(4):796–810, 1966.
- [34] Anibal Ollero Baturone. *ROBÓTICA: Manipuladores y robots móviles*. Boixareu Editores, 2001.
- [35] Carlos Balaguer & Rafael Aracil Antonio Barrientos, Luis Felipe. *Fundamentos De Robótica*. McGraw Hill, 1997.
- [36] K.S. Fu, R.C. González, y C.S.G. Lee. *Robótica: control, detección, visión e inteligencia*. McGraw-Hill, 1988.
- [37] Soon-Jo Chung y Jean-Jacques E. Slotine. Cooperative Robot Control and Concurrent Synchronization of Lagrangian Systems. *IEEE Transactions on Robotics*, 25(3):686–700, 2009.
- [38] Teun Koetsier y I Grattan-Guinness. Lakatos’ philosophy of mathematics. an historical approach. *Annals of Science*, 51(2):191–191, 1994.
- [39] Krzysztof Kozłowski y Dariusz Pazderski. Modeling and control of a 4-wheel skid-steering mobile robot. *International journal of applied mathematics and computer science*, 14:477–496, 2004.
- [40] Spyros G Tzafestas. *Introduction to mobile robot control*. Elsevier, 2013.
- [41] Hassan K Khalil y Jessy W Grizzle. *Nonlinear systems*, volume 3. Prentice hall Upper Saddle River, NJ, 2002.
- [42] Shankar Sastry. *Nonlinear systems: analysis, stability, and control*, volume 10. Springer Science & Business Media, 2013.
- [43] PG Balaji y D Srinivasan. An introduction to multi-agent systems. In *Innovations in multi-agent systems and applications-1*, pages 1–27. Springer, 2010.
- [44] Karl Johan Åström y Tore Hägglund. *Control PID avanzado*. Pearson, Madrid, 2009.

- [45] Elena N Gryazina. The d-decomposition theory. *Automation and Remote Control*, 65(12):1872–1884, 2004.
- [46] Shankar P Bhattacharyya y Lee H Keel. Robust control: the parametric approach. In *Advances in control education 1994*, pages 49–52. Elsevier, 1995.
- [47] H Chappellat, M Mansour, y SP Bhattacharyya. Elementary proofs of some classical stability criteria. *IEEE Transactions on Education*, 33(3):232–239, 1990.
- [48] Adrian-Josue Guel-Cortez, César-Fernando Méndez-Barrios, Eunjin Kim, y Mihir Sen. Fractional-order controllers for irrational systems. *IET Control Theory & Applications*, 15(7):965–977, 2021.
- [49] D. Torres-García, C. F. Méndez-Barrios, y A. Ramírez. Maximum exponential decay rate for first-order time-delay systems with pi controllers. *Submitted to Latin American Congress on Automation and Robotics 21*, 2021.
- [50] Edwin Pruna, Edison R Sasig, y Santiago Mullo. Pi and pid controller tuning tool based on the lambda method. In *2017 CHILEAN Conference on Electrical, Electronics Engineering, Information and Communication Technologies (CHILECON)*, pages 1–6. IEEE, 2017.
- [51] Massimiliano Veronesi y Antonio Visioli. On the selection of lambda in lambda tuning for pi(d) controllers. *IFAC-PapersOnLine*, 53(2):4599–4604, 2020. 21th IFAC World Congress.
- [52] Aidan O’Dwyer. A summary of pi and pid controller tuning rules for processes with time delay. part 1: Pi controller tuning rules. *IFAC Proceedings Volumes*, 33(4):159–164, 2000. IFAC Workshop on Digital Control: Past, Present and Future of PID Control, Terrassa, Spain, 5-7 April 2000.
- [53] A Haalman. Adjusting controllers for a deadtime process. *Control Engineering*, pages 71–73, 1965.
- [54] Jorge Angeles. *Fundamentals of Robotic Mechanical Systems*. Springer International Publishing, 2014.
- [55] C. Canudas, B. Siliciano, y G. Bastin. *Theory of Robot Control*. Springer, 1996.

- [56] Iman Anvari. *Non-holonomic Differential Drive Mobile Robot Control & Design: Critical Dynamics and Coupling Constraints*. Tesis de Doctorado, Arizona State University, 2013.
- [57] Ramón Ortigoza, María Vilchis, Victor Guzmán, Magdalena Melchor, y Edgar Flores. Modelado y control de un robot móvil tipo newt en la tarea de seguimiento de trayectoria. *Télématique: Revista Electrónica de Estudios Telemáticos*, ISSN 8156-4194, Vol. 7, N.º. 2, 2008, pags. 129-145, 9, 01 2008.
- [58] Abelardo del Pozo-Quintero. Modelado cinemático dinámico del mini robot RICIMAF. *Revista e Ingeniería Electrónica, Automática y Comunicaciones*, 33:49–62, 2012.
- [59] Azaza Awatef y A Kinematics Model. Dynamic Modeling And Inverse Dynamic Control Of Mobile Robot. *2017 International Conference on Green Energy Conversion Systems (GECS)*, pages 1–5, 2017.
- [60] Silviu-Iulian Niculescu. *Delay effects on stability: a robust control approach*, volume 269. Springer Science & Business Media, 2001.
- [61] M. Aicardi, G. Casalino, A. Bicchi, y A. Balestrino. Closed loop steering of unicycle like vehicles via lyapunov techniques. *IEEE Robotics Automation Magazine*, 2(1):27–35, 1995.