

ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA

CONTROL DE POSICIÓN Y SEGUIMIENTO DE TRAYECTORIAS DE UN MANIPULADOR MÓVIL DE 3 GRADOS DE LIBERTAD

PROYECTO PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN ELECTRÓNICA Y CONTROL

GUAMÁN RIVERA ROBERT BERNARDO
robert.guaman@epn.edu.ec

LOZADA CALLE WILSON SEBASTIÁN
wilson.lozada@epn.edu.ec

DIRECTOR: Dr. PAULO LEICA
paulo.leica@epn.edu.ec

Quito, Julio 2016

DECLARACIÓN

Nosotros, Guamán Rivera Robert Bernardo y Lozada Calle Wilson Sebastián, declaramos bajo juramento que el trabajo aquí descrito es de nuestra autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y, que hemos consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedemos nuestros derechos de propiedad intelectual correspondientes a este trabajo, a la Escuela Politécnica Nacional, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normatividad institucional vigente.

Guamán Rivera Robert Bernardo

Lozada Calle Wilson Sebastián

CERTIFICACIÓN

Certifico que el presente trabajo fue desarrollado por Guamán Rivera Robert Bernardo y Lozada Calle Wilson Sebastián, bajo mi supervisión.

Dr. Paulo Leica
DIRECTOR DEL PROYECTO

AGRADECIMIENTO

Agradezco a Dios, por haber protegido y brindado de salud en este tiempo de estudio.

A mis padres por haber apoyado y alentado en todo momento de mi vida universitaria.

A José, mi hermano con quien he compartido gratos momentos.

A Santiago y Mireya, mis hermanos quienes me brindaron su apoyo incondicional.

A mis tías Marta y María Guamán, por el apoyo brindado de manera desinteresada en el transcurso de mis estudios.

A mi tía Carmita quien me acogió, me brindó su apoyo y amor. A todos mis tíos y primos, quienes me brindaron su ayuda para poder cumplir con mi meta.

A mi compañero Sebastián por la dedicación, esfuerzo y la buena voluntad que puso para sacar adelante éste proyecto.

Al Dr. Paulo Leica, por haber guiado de manera desinteresada el desarrollo de éste proyecto y por motivarnos a salir adelante, así como por los conocimientos que nos brindó y por su paciencia para el desarrollo del mismo.

Al Personal del Laboratorio de Investigación Robótica y Sistemas Inteligentes de la Escuela Politécnica Nacional, por haber facilitado sus instalaciones. Al Dr. Danilo Chávez y al Dr. Oscar Camacho por la ayuda brindada para realizar éste proyecto.

A mis amigos los Juniors, con quienes he compartido los mejores momentos de mi vida universitaria.

Robert

DEDICATORIA

A mis padres Bernardino Guamán y Piedad Rivera, por el apoyo y la confianza que me brindaron. Con todo mi cariño y amor.

Robert

AGRADECIMIENTO

Primeramente gracias a Dios, por darme la salud y permitirme alcanzar esta meta.

A mi familia, la bendición más grande que Dios me ha dado. A mis padres Wilson y Martha por su amor incondicional, por su apoyo en cada momento de mi vida y por aguantarme todos estos años.

A mis hermanos Silvia y Byron por estar siempre a mi lado desde que nacieron y brindarme cada día su cariño.

A mi tía Hilda por brindarme su apoyo cuando lo necesité.

A Andrés, Carina, Andrea, Javier, Vanessa, Daniel, por su amistad sincera, y a todos mis amigos de la 'Poli' que talvez he olvidado nombrar, gracias totales.

A mi compañero y amigo Robert por el trabajo y esfuerzo dedicados al presente proyecto y que ayudó a superar todos los obstáculos encontrados durante la realización del mismo.

Al Dr. Paulo Leica, por su invaluable apoyo, ya que, gracias a su ayuda, a los conocimientos impartidos y a su acertada guía se pudo llevar a cabo el presente trabajo.

Al personal que labora en el Laboratorio de Investigación Robótica y Sistemas Inteligentes de la Escuela Politécnica Nacional por las facilidades y la ayuda brindadas para la realización del presente proyecto, en especial al Dr. Danilo Chávez, al Dr. Oscar Camacho y a Juan Carlos Suarez.

Sebastián

DEDICATORIA

A mis padres Martha de Jesús y Wilson Alberto.

Sebastián

CONTENIDO

RESUMEN	i
PRESENTACIÓN.....	ii
CAPÍTULO 1.....	1
FUNDAMENTO TEÓRICO.....	1
1.1 INTRODUCCIÓN A LA ROBÓTICA.....	1
1.1.1 DEFINICIÓN DE ROBOT	1
1.2 CLASIFICACIÓN DE LOS ROBOTS.....	2
1.3 ROBOT MÓVIL.....	3
1.3.1 ROBOT MÓVIL TIPO UNICICLO NO HOLONÓMICO	3
1.4 ROBOT MANIPULADOR	5
1.4.1 ESTRUCTURA DE LOS MANIPULADORES	5
1.5 MANIPULADOR MÓVIL.....	7
1.5.1 REDUNDANCIA CINEMÁTICA	8
1.6 REPRESENTACIÓN MATEMÁTICA DEL MOVIMIENTO DE ROBOTS	8
1.6.1 CINEMÁTICA.....	9
1.6.1.1 Cinemática Directa.....	10
1.6.1.2 Cinemática Inversa	10
1.6.1.3 Matriz Jacobiana	11
1.6.2 DINÁMICA	11
1.7 PRELIMINARES MATEMÁTICOS.....	12
1.7.1 ESCALARES Y VECTORES [17]	12
1.7.2 NORMA DE UN VECTOR [17]	12
1.7.3 MATRICES [18]	13

1.7.4	MATRIZ IDENTIDAD [18].....	13
1.7.5	MATRIZ TRASPUESTA [18].....	13
1.7.6	MATRIZ CUADRADA [18]	13
1.7.7	MATRIZ INVERSA [18].....	13
1.7.8	MATRIZ PSEUDOINVERSA POR LA IZQUIERDA [19].....	14
1.7.9	MATRIZ PSEUDOINVERSA POR LA DERECHA [19].....	14
1.7.10	FORMAS CUADRÁTICAS [19].....	14
1.7.11	PROPIEDADES DE LINEALIDAD [20].....	15
1.7.11.1	Propiedad de proporcionalidad.....	15
1.7.11.2	Propiedad de aditividad	15
1.7.12	SISTEMA DE ECUACIONES LINEALES [21]	15
1.7.13	APROXIMACIÓN POR MÍNIMOS CUADRADOS [20].....	16
1.7.14	APROXIMACIÓN DE EULER [22].....	16
1.8	CONTROL DE ROBOTS	17
1.8.1	CONTROL DE SISTEMAS NO LINEALES.....	18
1.8.2	TÉCNICAS DE CONTROL UTILIZADAS	19
1.8.2.1	Control basado en Mínima Norma	19
1.8.2.2	Control basado en Álgebra Lineal [28].....	19
1.8.2.3	Control por Modos Deslizantes (CMD).....	20
1.8.3	OBJETIVOS DE CONTROL	23
1.8.3.1	Control de posición	23
1.8.3.2	Seguimiento de trayectoria.....	24
1.8.3.3	Seguimiento de Caminos	25
1.9	RENDIMIENTO DE LOS SISTEMAS DE CONTROL [34].....	25
1.9.1	CRITERIO INTEGRAL DEL ERROR ABSOLUTO (IAE)	25

1.10 ESTABILIDAD EN SISTEMAS ROBÓTICOS [9]	26
1.10.1 TEORÍA DE ESTABILIDAD DE LYAPUNOV.....	26
1.10.1.1 Función definida positiva.....	27
1.10.1.2 Función semidefinida positiva.....	27
1.10.1.3 Función definida negativa.....	28
1.10.1.4 Función semidefinida negativa.....	29
1.10.1.5 Función candidata de Lyapunov.....	30
1.10.1.6 Método Directo de Lyapunov.....	31
1.10.1.7 Estabilidad en el sentido de Lyapunov.....	31
1.10.1.8 Teorema de estabilidad.....	32
1.11 SIMULACIÓN DE ROBOTS EN COMPUTADOR	32
1.11.1 MATLAB [37].....	33
1.11.2 SOLIDWORKS [39].....	34
CAPÍTULO 2	36
2 ESTUDIO Y MODELADO DEL MANIPULADOR MÓVIL	36
2.1 MODELO CINEMÁTICO	36
2.1.1 CINEMÁTICA DEL ROBOT MÓVIL TIPO UNICICLO.....	36
2.1.2 CINEMÁTICA DEL BRAZO MANIPULADOR.....	37
2.1.3 CINEMÁTICA DEL MANIPULADOR MÓVIL DE TRES GRADOS DE LIBERTAD.....	40
2.1.4 CINEMÁTICA INVERSA.....	46
2.2 MODELO DINÁMICO	47
2.3 SIMULACIÓN DEL MODELO	49
2.3.1 SIMULACIÓN DEL MODELO CINEMÁTICO.....	49
2.3.2 SIMULACION DEL MODELO DINÁMICO.....	54

CAPÍTULO 3	58
3 DISEÑO E IMPLEMENTACIÓN DE LOS ALGORITMOS DE CONTROL	58
3.1 DISEÑO DE LOS CONTROLADORES	58
3.1.1 CONTROLADOR BASADO EN MÍNIMA NORMA.....	58
3.1.1.1 Análisis de Estabilidad	61
3.1.2 CONTROLADOR BASADO EN APROXIMACIONES DE EULER.....	62
3.1.3 CONTROLADOR BASADO EN MODOS DESLIZANTES	64
3.1.3.1 Controlador CMD con una superficie proporcional.....	65
3.1.3.2 Controlador CMD con una superficie proporcional integral	67
3.2 IMPLEMENTACIÓN DE LOS ALGORITMOS DE CONTROL	69
3.2.1 BLOQUE GENERADOR DE TRAYECTORIAS	69
3.2.2 BLOQUE SISTEMA DE CONTROL.....	70
3.2.3 BLOQUE MANIPULADOR MÓVIL	71
3.2.4 BLOQUE TIEMPO DE SIMULACIÓN.....	71
3.2.5 DESARROLLO DEL ENTORNO VIRTUAL	72
CAPÍTULO 4	74
4 PRUEBAS DE SIMULACIÓN	74
PRUEBA 1: REALIDAD VIRTUAL VRML	74
4.1 PRUEBA 2: CONTROL DE POSICIÓN APLICADO AL MODELO CINEMÁTICO	78
4.1.1 RENDIMIENTO DE LOS CONTROLADORES CON EL MODELO CINEMÁTICO	82
4.2 PRUEBA 3: CONTROL DE POSICIÓN APLICADO AL MODELO DINÁMICO	83

4.2.1	RENDIMIENTO DE LOS CONTROLADORES CON EL MODELO DINÁMICO.....	85
4.3	PRUEBA 4: SEGUIMIENTO DE TRAYECTORIA APLICADO AL MODELO CINEMÁTICO	86
4.3.1	SEGUIMIENTO DE TRAYECTORIA SENO	87
4.3.2	SEGUIMIENTO DE TRAYECTORIA TIPO OCHO	90
4.3.3	SEGUIMIENTO DE TRAYECTORIA CUADRADA	92
4.3.4	SEGUIMIENTO DE TRAYECTORIA CIRCULAR.....	95
4.3.5	RENDIMIENTO DE LOS CONTROLADORES CON EL MODELO CINEMÁTICO	97
4.4	PRUEBA 5: SEGUIMIENTO DE TRAYECTORIAS APLICADO AL MODELO DINÁMICO	99
4.4.1	SEGUIMIENTO DE TRAYECTORIA SENO	100
4.4.2	SEGUIMIENTO DE TRAYECTORIA TIPO OCHO	103
4.4.3	SEGUIMIENTO DE TRAYECTORIA CUADRADA	106
4.4.4	SEGUIMIENTO DE TRAYECTORIA CIRCULAR.....	108
4.4.5	RENDIMIENTO DE LOS CONTROLADORES CON EL MODELO DINÁMICO.....	111
CAPÍTULO 5	114
5	CONCLUSIONES Y RECOMENDACIONES	114
5.1	CONCLUSIONES	114
5.2	RECOMENDACIONES	115
6	REFERENCIAS BIBLIOGRÁFICAS	116
7	ANEXOS.....	120

RESUMEN

El presente trabajo está enfocado en resolver los problemas de control de posición y seguimiento de trayectorias para un manipulador móvil de tres grados de libertad, el cual es un robot formado por un brazo manipulador montado sobre una plataforma móvil. El estudio se realiza en base a datos de características de dos robots comerciales: el robot móvil “Pionner 3D-X” y el brazo manipulador “Mover 4”.

Con la finalidad de alcanzar los objetivos de control, se proponen cuatro controladores basados en tres técnicas de control diferentes: i) un controlador PID basado en mínima norma, ii) un controlador utilizando álgebra lineal y iii) dos controladores basados en modos deslizantes. Estos controladores están diseñados en base al modelo cinemático, por lo que se presenta el desarrollo de dicho modelo, partiendo de las relaciones geométricas del manipulador móvil. Para probar la robustez de los controladores, se implementa en el modelo dinámico aproximado del manipulador móvil, se hace esta aproximación debido a que la masa del manipulador es despreciable respecto a la masa de la plataforma móvil. Para cada controlador se presenta el análisis de estabilidad, utilizando algunos conceptos de la teoría de estabilidad de Lyapunov.

El control del manipulador móvil se realiza en simulación utilizando el software Matlab y su paquete de programación gráfica Simulink, se analiza el rendimiento de cada controlador mediante el índice integral del error absoluto IAE y se realiza una comparación entre controladores.

Para tener una mejor visualización del robot y observar su comportamiento, se realiza la representación gráfica del mismo utilizando Solidwoks, un programa de diseño asistido por computador (CAD), mediante el cual se diseña el manipulador móvil para luego convertir el archivo del dibujo a formato de realidad virtual VRML. Este dibujo se inserta luego en un entorno de realidad virtual creado mediante las herramientas disponibles en Matlab, en el cual se puede observar el seguimiento de trayectorias del manipulador móvil.

PRESENTACIÓN

El presente trabajo tiene como objetivo implementar mediante un software de simulación, el control de posición y seguimiento de trayectorias de un manipulador móvil de tres grados de libertad.

En el Capítulo 1 se presentan los conceptos teóricos utilizados en el desarrollo del proyecto, se hace una breve introducción a la robótica, se presentan también definiciones de los robots que componen el manipulador móvil, además se estudian los conceptos de representación matemática de robots, así como definiciones matemáticas necesarias para estudiar las técnicas de control utilizadas, finalmente se presenta un resumen de la teoría de estabilidad de Lyapunov y una introducción a los programas computacionales utilizados para la simulación.

En el Capítulo 2 se estudian los modelos matemáticos del manipulador móvil, para la obtención del modelo cinemático se analizan los modelos del robot móvil y el brazo manipulador por separado y se unen en un solo modelo. También se presenta el modelo dinámico del robot móvil, validado en De la Cruz et al. (2006), la simulación de estos modelos se realiza mediante simulación en Matlab.

En el Capítulo 3 se presenta el diseño de los algoritmos de control, i) un controlador PID basado en mínima norma, ii) un controlador utilizando álgebra lineal y iii) dos controladores basados en modos deslizantes. Se realiza el análisis de estabilidad para cada controlador y se presenta la implementación de los mismos en el paquete Simulink de Matlab.

En el Capítulo 4 se presentan resultados de la simulación de los controladores implementados para control de posición y seguimiento de trayectorias. También se presenta el rendimiento mediante el índice integral de error absoluto IAE y la comparación entre controladores.

En el capítulo 5 se presentan conclusiones y recomendaciones.

CAPÍTULO 1

FUNDAMENTO TEÓRICO

En este capítulo se presenta una introducción a la robótica, una breve reseña histórica y una clasificación de los robots para poder enmarcar al manipulador móvil en el presente proyecto. De igual manera se hace una introducción al modelamiento de robots y las técnicas de diseño de controladores utilizadas. Este capítulo también abarca los conceptos de estabilidad de sistemas, haciendo énfasis en la teoría de estabilidad de Lyapunov, así como las herramientas matemáticas que en los capítulos posteriores serán utilizadas para el modelamiento del robot, diseño de los sistemas de control y pruebas de estabilidad.

1.1 INTRODUCCIÓN A LA ROBÓTICA

Desde hace siglos la humanidad ha tenido la necesidad de crear algún mecanismo que se asemeje e imite los movimientos de los seres vivos y que realice tareas forzosas o que resultan riesgosas para el ser humano. El estudio de la robótica ha estado ligado a esta necesidad [1]. Actualmente la Robótica se define como el conocimiento y el arte requeridos para diseñar y construir dispositivos o sistemas de interacción hombre-máquina con la finalidad de facilitar distintas tareas del devenir humano [2].

Es en el año de 1921, que el escritor checo Karel Capek, en su obra Rossum's Universal Robot (R.U.R) introduce por primera vez el término "robot" [1], que en su idioma significa fuerza de trabajo o servidumbre [3].

1.1.1 DEFINICIÓN DE ROBOT

En la actualidad, la definición de robot es revisada y ampliada frecuentemente debido al desarrollo tecnológico vertiginoso y al amplio campo de dispositivos que hoy en día son reconocidos como tal, por lo que resulta complicado dar una definición exacta. A continuación se presentan dos definiciones consideradas por diccionarios y enciclopedias [1]:

Enciclopedia Británica: *“Máquina operada automáticamente que sustituye el esfuerzo de los humanos, aunque no tiene por qué tener apariencia humana o desarrollar sus actividades a la manera de los humanos”.*

Diccionario de la Real Academia Española: *“Máquina o ingenio electrónico programable, capaz de manipular objetos y realizar operaciones antes observadas sólo a las personas”.*

En un esfuerzo conjunto de la Federación Internacional de Robótica (IFR) y la Comisión Económica de Naciones Unidas para Europa (UNUCE) por homogenizar términos, en 1995 se obtuvo la siguiente definición: *“Un robot es un mecanismo accionado programable en dos o más ejes con cierto grado de autonomía, moviéndose dentro de su entorno para ejecutar tareas previstas”.*

En este contexto autonomía significa la capacidad para realizar las tareas en base a estado actual y detección, sin intervención humana [4].

1.2 CLASIFICACIÓN DE LOS ROBOTS

Existe una gran variedad de robots con diversas estructuras, funcionalidad y aplicación, pero de manera general se pueden clasificar en [1] [4]:

- robots móviles
- robots humanoides
- robots de servicio
- robots industriales

Un Robot de servicio es aquel que realiza tareas útiles para el ser humano y equipos, con excepción de aplicaciones de automatización industrial [4], por tanto, un robot industrial puede ser considerado como robot de servicio cuando no realice tareas de tipo industrial o manufacturera [1].

El manipulador móvil analizado en el presente proyecto de titulación forma parte de los robots de servicio.

1.3 ROBOT MÓVIL

El robot móvil, según la definición de la Organización Internacional para Estandarización es un robot que contiene los sistemas de potencia, control y navegación necesarios para su movimiento [1]. Un ejemplo de este tipo de robot se presenta en la Figura 1.1, tiene la capacidad de desplazarse con autonomía en entornos conocidos o desconocidos, es decir que para su desplazamiento utiliza herramientas de percepción, modelación y planificación que forman parte su misma estructura [5].



Figura 1.1. Robot Móvil

1.3.1 ROBOT MÓVIL TIPO UNICICLO NO HOLONÓMICO

Una fotografía de un robot móvil tipo unicycle se presenta en la Figura 1.2.



Figura 1.2. Robot Móvil tipo unicycle

Este tipo de robot utiliza como sistema de locomoción un par de ruedas montadas sobre un mismo eje, controladas independientemente, además de una rueda de castor que le proporciona estabilidad para desplazarse en un ambiente de trabajo.

La expresión “robot móvil tipo unicycle” se utiliza para identificar a robots que tienen la capacidad de girar sobre su propio eje [6].

El término “no holonómico” hace referencia a una restricción de movimiento, pues este tipo de robot solo puede moverse en forma perpendicular al eje que une sus dos ruedas [6], no puede desplazarse en direcciones arbitrarias, debido a su configuración espacial [7].

La restricción no holonómica se puede entender mejor al analizar la Figura 1.3, donde se observa que el robot únicamente puede realizar movimientos hacia adelante o hacia atrás y no hacia los lados.

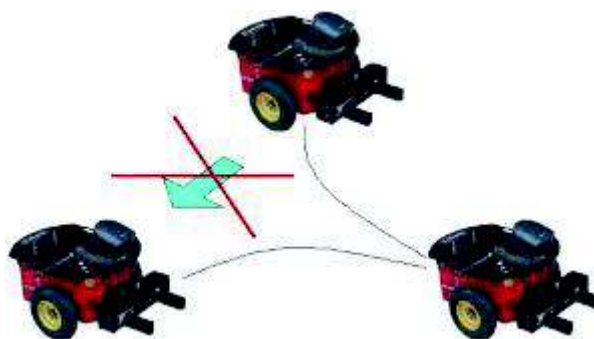


Figura 1.3. Restricción no holonómica [8]

Un ejemplo de este tipo de robot es el Pioneer 3-DX, mostrado en la Figura 1.4, un robot de tracción diferencial de tres ruedas y dos motores.



Figura 1.4. Robot Pioneer 3-DX [8]

El Pioneer 3-DX es utilizado para usos de investigación y educativos, cuenta con sensores de proximidad, sensores de velocidad en las ruedas, un microcontrolador con entradas digitales y análogas, además de un software para programación. Más detalles de este robot se presentan en el Anexo A.

1.4 ROBOT MANIPULADOR

Los robots industriales, también conocidos como robots manipuladores, brazos robots o brazos mecánicos, haciendo analogía con el brazo humano, se componen de una base que puede rotar sobre su propio eje y de articulaciones que simulan el movimiento de codo y brazo, en el extremo de la última articulación se encuentra la herramienta final que define la tarea a realizar. La ISO define a este tipo de robot como un manipulador multipropósitos, reprogramable y controlado automáticamente en tres o más ejes [9].



Figura 1.5. Robot Manipulador

1.4.1 ESTRUCTURA DE LOS MANIPULADORES

Los robots manipuladores son esencialmente brazos articulados, de manera más precisa un manipulador es una cadena cinemática abierta, formada por un conjunto de eslabones que se relacionan entre sí mediante articulaciones, como se observa en la Figura 1.5.

Las articulaciones del manipulador pueden ser de distintos tipos y permiten al eslabón rotar o desplazarse sobre los ejes de la articulación, a esta capacidad de rotación o traslación se conoce como grado de libertad. Los manipuladores pueden adoptar distintas configuraciones dependiendo de la disposición de sus eslabones y tipo de articulaciones, la configuración del manipulador estudiado en este proyecto es la configuración angular. Como se muestra en la Figura 1.6, la configuración angular tiene tres articulaciones de tipo rotacional, las que le proporcionan tres grados de

libertad, su estructura permite un mejor acceso a espacios cerrados y es fácil desde el punto de vista constructivo [3].



Figura 1.6. Manipulador en configuración angular

La configuración angular es la más utilizada en actividades de educación, investigación y desarrollo [3].

Un efector final, como el mostrado en la Figura 1.7, es el elemento que se ubica en el extremo del último enlace del manipulador, permite la colocación de una herramienta, como por ejemplo una pinza si la tarea deseada es manipulación u otra herramienta dependiendo de la tarea a realizar (pintura, soldadura, etc.).



Figura 1.7. Efector Final tipo gripper

Un ejemplo de manipulador de tres grados de libertad es el Mover4 [10], mostrado en la Figura 1.8, diseñado por la empresa alemana Commonplace Robotics, es un robot de cuatro ejes para uso en entornos educativos, de entretenimiento y de investigación.

Las articulaciones del brazo se mueven por servomotores, el extremo frontal del robot permite montar una pinza u otra herramienta.

El Mover4 es el manipulador que, montado sobre la plataforma móvil PIONEER 3-DX forman el manipulador móvil, objeto de estudio del presente proyecto.



Figura 1.8. Robot Mover 4 [10]

1.5 MANIPULADOR MÓVIL

Es un robot que se origina de la unión de dos robots diferentes, consta de un robot manipulador montado sobre una base que a su vez es un robot móvil. Un ejemplo de manipulador móvil es el G-Ball de la empresa Robotnik, mostrado en la Figura 1.9.



Figura 1.9. Manipulador Móvil G Ball [11]

La razón de combinar dos robots en uno solo es aumentar las ventajas que cada uno tiene cuando actúa por separado, como en este caso, aumentando el espacio de trabajo del manipulador estacionario y agregando al robot móvil la capacidad de manipulación [12].

Los sistemas manipuladores móviles se consideran cada vez más para aplicaciones en entornos peligrosos u hostiles, donde el acceso humano es limitado. También están atrayendo un interés significativo en los ámbitos de fabricación, militares y de servicio público [13]

1.5.1 REDUNDANCIA CINEMÁTICA

Cuando un manipulador posee más grados de libertad que el mínimo número requerido para realizar una tarea dada ocurre lo que se conoce como redundancia cinemática [14]. En el manipulador móvil, la capacidad de desplazamiento que la base móvil aporta al brazo manipulador, incrementa los grados de libertad del brazo, por lo que se considera que este sistema tiene un alto grado de redundancia cinemática [13], esto determina que existan varias maneras en que el manipulador móvil pueda realizar una determinada tarea, o dicho de otra manera, que el sistema tenga varias soluciones.

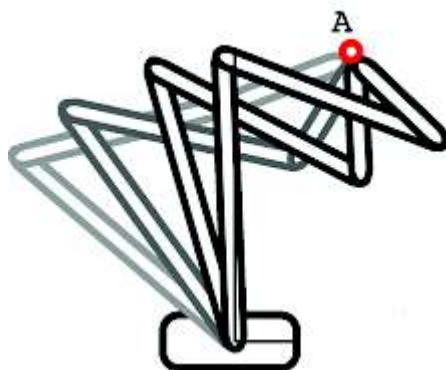


Figura 1.10. Redundancia cinemática

En la Figura 1.10 se puede ver el efecto de la redundancia cinemática, el manipulador tiene múltiples opciones para realizar la tarea, que en este caso es posicionar su efector final en el punto A.

1.6 REPRESENTACIÓN MATEMÁTICA DEL MOVIMIENTO DE ROBOTS

En robótica se necesita describir las posiciones y orientaciones de objetos en el espacio, de modo que estas puedan ser reconocidas por el robot, para que pueda realizar la tarea encomendada, de aquí la necesidad de contar con herramientas matemáticas que permitan identificar estas posiciones y orientaciones. Para representar la posición del robot, el sistema de referencia utilizado en este trabajo consta de ejes perpendiculares entre sí con un origen definido, y es conocido como

sistema cartesiano. Para definir un robot en el espacio es necesario conocer su posición y su ángulo de orientación [1].

1.6.1 CINEMÁTICA

La cinemática es el estudio del movimiento, sin tomar en cuenta las fuerzas que lo provocan, por tanto abarca el estudio de las propiedades geométricas y temporales del movimiento, en este caso de los robots. Desde este punto de vista el problema se vuelve solamente geométrico para el posicionamiento y de velocidades que describen el movimiento [3].

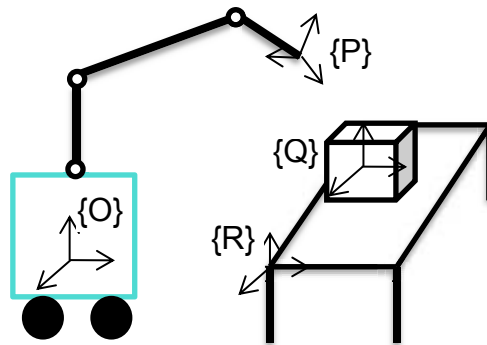


Figura 1.11. Diferencia entre espacio de trabajo y espacio articular

En la Figura 1.11 se puede observar la diferencia entre espacio de trabajo y espacio articular, la diferencia radica en que el estudio en el espacio de trabajo toma en cuenta el movimiento del efector final sin considerar los movimientos que hagan las articulaciones para lograrlo; los sistemas $\{Q\}$ y $\{R\}$ son sistemas de referencia en el espacio de trabajo. En el caso del estudio en el espacio de las articulaciones, el sistema de referencia se ubica sobre las articulaciones, y se analiza el movimiento que cada articulación tiene que hacer para posicionar al efector final. Los sistemas $\{O\}$ y $\{P\}$ son sistemas en el espacio articular [1].

En los modelos cinemáticos se estudia principalmente las relaciones que existen entre el espacio de las variables articulares y el espacio de trabajo que generalmente es un espacio cartesiano [3]. Dentro del estudio de la cinemática del manipulador móvil se analizan dos problemas, el de la cinemática directa y el de la cinemática inversa, los cuales se explican a continuación.

1.6.1.1 Cinemática Directa

Tiene como objetivo describir la geometría y ubicación en el espacio de cada eslabón que constituye el robot [2], se utiliza para determinar la posición y orientación del efector final del robot con respecto a un sistema de referencia [1], en función de las variables de las articulaciones y parámetros geométricos de la estructura mecánica, el origen del sistema de referencia generalmente es la base del robot [15]. En la Figura 1.12 se presenta un diagrama de la cinemática directa de un manipulador.

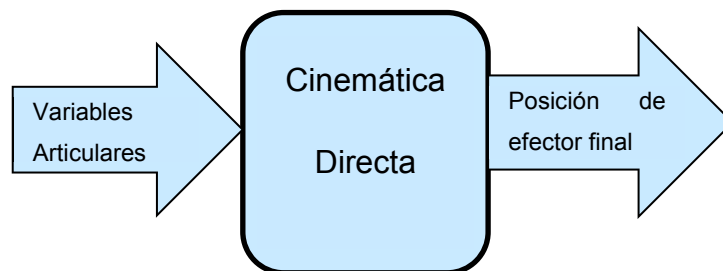


Figura 1.12. Cinemática directa del manipulador

1.6.1.2 Cinemática Inversa

El problema cinemático inverso consiste en determinar las variables de articulaciones correspondientes a una orientación y posición específicas del efector final, es decir resuelve la configuración que debe adoptar el robot para una posición y orientación del extremo conocidas [1]. En la Figura 1.13 se presenta un diagrama de la cinemática inversa de un manipulador.

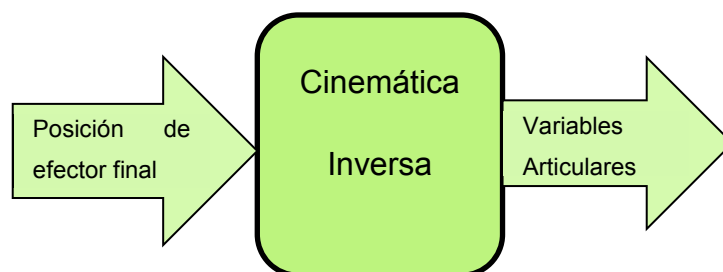


Figura 1.13. Cinemática inversa del manipulador

Al contrario que el problema cinemático directo, el cálculo de la cinemática inversa no es sencillo, pues implica la resolución de una serie de ecuaciones fuertemente dependiente de la configuración del robot [16]. En la Figura 1.14 se muestra un diagrama de la relación entre la cinemática directa para un manipulador.

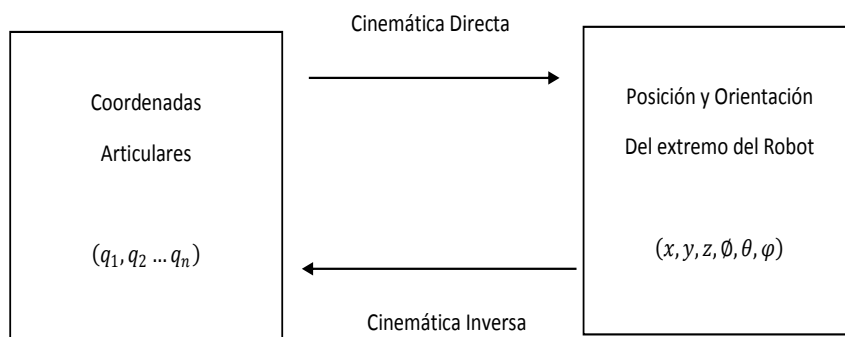


Figura 1.14. Diagrama de relación entre cinemática directa e inversa del manipulador [1]

1.6.1.3 Matriz Jacobiana

El modelado cinemático de un robot busca relacionar variables articulares con la posición y orientación del extremo del robot, sin tomar en cuenta las fuerzas o pares que originan el movimiento. Sin embargo, se necesita conocer la relación entre la velocidad de las articulaciones y la velocidad del efector final, esta relación se concreta en la denominada matriz Jacobiana [1].

La Figura 1.15 muestra un diagrama de la matriz Jacobiana de un manipulador.

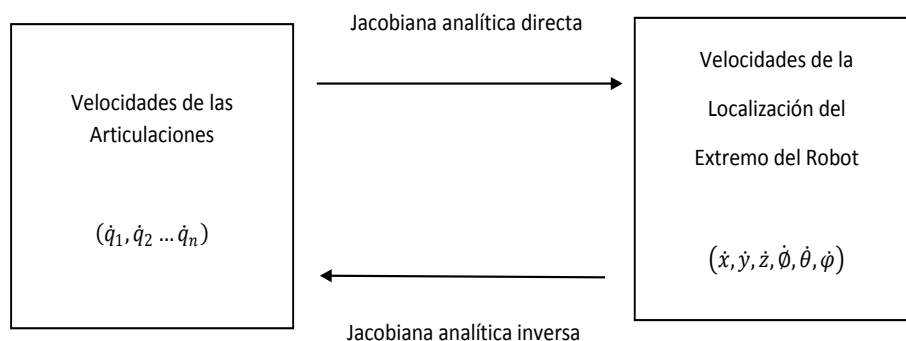


Figura 1.15. Diagrama de la matriz Jacobiana de un manipulador [1]

1.6.2 DINÁMICA

En la sección 1.6.1 se mencionó que la cinemática estudia el movimiento de los cuerpos sin analizar las fuerzas que lo provocan, el análisis de dichas fuerzas es el objetivo del estudio de la dinámica. Las velocidades lineales y angulares de un robot tienen su origen en fuerzas y pares aplicados a la estructura mecánica del mismo, los

cuales dependen de la magnitud de la masa del robot y de su distribución, las relaciones entre todos estos parámetros constituyen el modelo dinámico. La identificación del modelo dinámico de un robot es, en general difícil y además complica el desarrollo del sistema de control, es por esto que muchos controladores son diseñados tomando en consideración principalmente el modelo cinemático, obteniendo resultados aceptables para bajas velocidades y cambios suaves de trayectoria, sin embargo, cuando se requiere realizar tareas que involucren cambios rápidos de trayectoria, además de variaciones importantes de velocidad y masa, se vuelve imprescindible la consideración del modelo dinámico en el diseño del sistema de control [3].

1.7 PRELIMINARES MATEMÁTICOS

A continuación se presentan conceptos básicos para entender de mejor manera la explicación acerca de las técnicas de control aplicadas al manipulador móvil del presente proyecto.

1.7.1 ESCALARES Y VECTORES [17]

En la mayoría de aplicaciones físicas aparecen cantidades como la distancia y la rapidez que solo poseen magnitud y pueden ser representadas por números que se denominan escalares, otras cantidades como la velocidad como la velocidad que poseen magnitud y dirección, estas pueden ser representadas por flechas que parten de un origen y se denominan vectores. En el presente proyecto se asume que los escalares pertenecen al conjunto de los números reales \mathbb{R} , y que los vectores pertenecen al conjunto de n – plas de números reales \mathbb{R}^n . Un ejemplo de escalar es la distancia $d = 5 [m]$ y un ejemplo de vector es la velocidad $\vec{v} = (5\vec{i} + 3\vec{j})m/s$.

1.7.2 NORMA DE UN VECTOR [17]

La definición de norma se origina en el teorema de Pitágoras, e indica la longitud de un vector. Sea $u = (u_1, u_2, \dots, u_n)$ un vector en \mathbb{R}^n , su norma $\|u\|$ se define como:

$$\|u\| = \sqrt{u_1^2 + u_2^2 + \dots + u_n^2}$$

1.7.3 MATRICES [18]

Una matriz de m filas y n columnas es un arreglo de $n \times m$ números a_{ij} , donde a_{ij} está en la fila i –ésima y columna j –ésima, al conjunto de matrices de m filas y n columnas se denota por $M_{m \times n}$. Cuando una matriz tiene igual número de filas que de columnas se denomina matriz simétrica.

1.7.4 MATRIZ IDENTIDAD [18]

La matriz identidad denotada por I , es una matriz que cumple la propiedad de ser elemento neutro del producto de matrices, es decir que al multiplicarse por una matriz A de m filas y n columnas, el resultado es la misma matriz A .

$$I_m A = A I_n = A, \text{ con } A \in M_{m \times n}$$

1.7.5 MATRIZ TRASPUESTA [18]

La matriz traspuesta A^T de otra matriz A reemplaza las filas de la matriz original por sus columnas y viceversa. De esta manera, la matriz A con dimensiones $n \times m$, tiene una matriz traspuesta con dimensiones $m \times n$.

1.7.6 MATRIZ CUADRADA [18]

Es una matriz de m filas y n columnas, en la cual el número de filas coincide con el número de columnas, es decir $m = n$.

1.7.7 MATRIZ INVERSA [18]

Una matriz cuadrada A de m filas y n columnas es invertible si existe una matriz llamada matriz inversa y denotada por A^{-1} , tal que al multiplicarse entre ellas, el resultado sea la matriz identidad I , es decir que cumpla con lo siguiente:

$$A A^{-1} = I$$

Para matrices que no son cuadradas, como en el caso del Jacobiano del modelo cinemático del manipulador móvil, su matriz inversa se calcula utilizando las definiciones de matriz pseudoinversa, enunciadas a continuación.

1.7.8 MATRIZ PSEUDOINVERSA POR LA IZQUIERDA [19]

Se utiliza para matrices de mayor número de filas que de columnas, sea la matriz A , su pseudoinversa A^+ por la izquierda está definida por:

$$A^+ = (A^T \cdot A)^{-1} A^T$$

1.7.9 MATRIZ PSEUDOINVERSA POR LA DERECHA [19]

Se utiliza para matrices de menor número de filas que de columnas, sea la matriz A , su pseudoinversa A^+ por la derecha está definida por:

$$A^+ = A^T (A \cdot A^T)^{-1}$$

1.7.10 FORMAS CUADRÁTICAS [19]

Para realizar el análisis de estabilidad de los controladores implementados en el presente proyecto, se requiere conocer el concepto de forma cuadrática y su matriz asociada para el planteamiento de las funciones de Lyapunov.

Una forma cuadrática es una aplicación $q: \mathbb{R} \rightarrow \mathbb{R}^n$ que a cada vector $\vec{x} \in \mathbb{R}^n$ le hace corresponder un número real que forma un polinomio homogéneo de segundo grado en las variables del vector \vec{x} . Una forma cuadrática puede expresarse como una expresión matricial de la forma:

$$q(\vec{x}) = X^T A X$$

Donde A es la matriz asociada de la forma cuadrática.

Clasificación de las formas cuadráticas:

Sea $q: \mathbb{R} \rightarrow \mathbb{R}^n$ una forma cuadrática y $\vec{x} = (x_1, x_2, \dots, x_n) \in \mathbb{R}^n$, se dice que:

- $q(\vec{x})$ es definida positiva si $q(\vec{x}) > 0, \forall \vec{x} \in \mathbb{R}^n, \vec{x} \neq 0$.
- $q(\vec{x})$ es definida negativa si $q(\vec{x}) < 0, \forall \vec{x} \in \mathbb{R}^n, \vec{x} \neq 0$.
- $q(\vec{x})$ es semidefinida positiva si $q(\vec{x}) \geq 0, \forall \vec{x} \in \mathbb{R}^n$.
- $q(\vec{x})$ es semidefinida negativa si $q(\vec{x}) \leq 0, \forall \vec{x} \in \mathbb{R}^n$.

1.7.11 PROPIEDADES DE LINEALIDAD [20]

Son condiciones que debe cumplir determinado sistema para ser definido como un sistema lineal, su estudio es necesario en el campo de los sistemas de control, pues en base a estas condiciones se puede distinguir un sistema lineal de uno no lineal e implementar las técnicas de control más adecuadas para cada caso.

Un sistema lineal es aquel que verifica el principio de superposición, el cual engloba las propiedades de proporcionalidad y aditividad, las cuales se detallan a continuación.

1.7.11.1 Propiedad de proporcionalidad

Un sistema H cumple la propiedad de proporcionalidad cuando al multiplicar la entrada $f(t)$ del sistema por una constante de valor k , la salida del sistema queda multiplicada por la misma constante.

$$H(kf(t)) = kH(f(t))$$

1.7.11.2 Propiedad de aditividad

Esta propiedad establece que la respuesta de un sistema H , a la aplicación de una suma de entradas $f_n(t)$ es igual a la suma de respuestas del sistema, a cada entrada aplicada por separado.

$$H(f_1(t) + f_2(t)) = H(f_1(t)) + H(f_2(t))$$

Estas dos propiedades permiten también identificar sistemas no lineales, como se verá en la sección 1.8.1

1.7.12 SISTEMA DE ECUACIONES LINEALES [21]

Una ecuación lineal es una expresión de la forma $a_1x_1 + \dots + a_nx_n = b$; $a, b \in \mathbb{R}$ donde los términos $a_1 \dots a_n$ se denominan coeficientes y b es el término independiente.

Un sistema de ecuaciones lineales es un conjunto de m ecuaciones con las mismas incógnitas.

$$a_{11}x_1 + \dots + a_{1n}x_n = b_1$$

$$a_{21}x_1 + \dots + a_{2n}x_n = b_2$$

$$\vdots$$

$$a_{m1}x_1 + \dots + a_{mn}x_n = b_m$$

Un sistema de ecuaciones puede tener solución única, varias soluciones o no tener solución, cuando no existe solución el sistema se denomina sistema incompatible.

A menudo los sistemas incompatibles aparecen en la vida real por lo que se ha desarrollado una técnica para determinar una solución aproximada tan cercana como sea posible, este método se denomina aproximación por mínimos cuadrados.

1.7.13 APROXIMACIÓN POR MÍNIMOS CUADRADOS [20]

Entre los controladores diseñados en el presente proyecto se encuentra un controlador tipo PID basado en mínima norma, cuyo funcionamiento se explica mediante la aproximación por mínimos cuadrados para resolución de sistemas de ecuaciones, la cual se detalla a continuación.

Sea el sistema incompatible: $Ax = b$, con $A \in M_{n \times m}$; $x \in M_{m \times 1}$; $b \in M_{n \times 1}$, se llama solución aproximada por mínimos cuadrados a todo vector $\hat{x} \in M_{m \times 1}$; que haga mínima la distancia entre Ax y b , es decir: $\|A\hat{x} - b\| \leq \|Ax - b\|$ para cualquier otro vector $x \in M_{m \times 1}$.

La solución por mínimos cuadrados \hat{x} es igual a la proyección de b sobre el espacio generado por las columnas de A .

1.7.14 APROXIMACIÓN DE EULER [22]

Es un método que se utiliza para expresar un sistema continuo en tiempo discreto, consiste en encontrar el valor del siguiente estado de una variable, aplicando para esto una aproximación lineal.

En la Figura 1.16, el valor del siguiente estado de la variable $x(t)$ se calcula integrando su derivada $\dot{x}(t)$, en el intervalo $t_m \leq t \leq t_{m+1}$. Esta integral se puede interpretar como una diferencia finita hacia adelante, se la expresa de la siguiente forma:

$$\Delta x(t) = \int_{t_m}^{t_{m+1}} \dot{x}(t) dx \approx \dot{x}(t_m)(t_{m+1} - t_m) \quad (1.1)$$

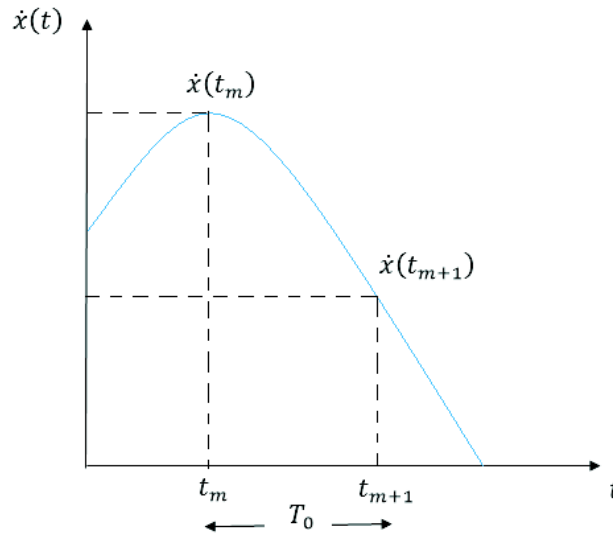


Figura 1.16. Método de aproximación de Euler [23]

De esta manera, la relación entre el estado actual x_m y el estado siguiente x_{m+1} queda definida por:

$$x_{m+1} = x_m + T_0 \dot{x}_m \quad (1.2)$$

1.8 CONTROL DE ROBOTS

El diseño de controladores para robots sigue siendo un área de intenso estudio por parte de los constructores de robots y diseñadores de sistemas de control, a pesar de la existencia de gran variedad de robots comerciales.

En la actualidad se podría pensar que la mayoría de robots industriales y comerciales son capaces de realizar sus tareas de manera adecuada, por lo que, aparentemente no es necesario el desarrollo de nuevos sistemas de control, sin embargo, el estudio de esto último no solo representa un gran reto teórico sino que es indispensable en aplicaciones específicas que no pueden ser llevadas a cabo por los robots comerciales [24], como se muestra en la Figura 1.17.



Figura 1.17. Robots de Nissan ensamblando automóvil en fábrica de Inglaterra, [25]

1.8.1 CONTROL DE SISTEMAS NO LINEALES

La representación matemática del manipulador móvil del presente proyecto, constituye un sistema no lineal.

Como se mencionó en la sección 1.7.11, un sistema lineal es aquel que satisface el principio de superposición, es decir que cumple las propiedades de proporcionalidad y aditividad. Un sistema no lineal es aquel que no cumple con alguna o ninguna de estas propiedades.

La teoría de sistemas de control estudia el análisis y diseño de los componentes que interactúan en un sistema, cuya configuración se basa en el principio de realimentación, el cual consiste en medir las variables de interés del sistema, para con esa información controlar su comportamiento.

El análisis del sistema de control parte de la representación matemática del mismo, como un conjunto de ecuaciones diferenciales o un operador entre entradas y salidas del sistema. La mayoría de los modelos utilizados tradicionalmente en control son de tipo lineal, debido a su simplicidad y a que representan el comportamiento de sistemas reales en muchos casos útiles. Sin embargo, el desarrollo de la tecnología ha generado una variedad de nuevos problemas y aplicaciones que son no lineales en esencia, como es el caso de los robots manipuladores, de ahí la importancia del análisis y desarrollo de herramientas no lineales de control [26].

El control de robots móviles es un área de la robótica que se ha venido desarrollando para brindar mayor autonomía a los mismos.

Existe una variedad de técnicas de control para sistemas no lineales, en el presente proyecto se utilizan tres, las cuales se detallan a continuación.

1.8.2 TÉCNICAS DE CONTROL UTILIZADAS

En el presente proyecto se estudian tres técnicas de control para el manipulador móvil, control basado en mínima norma, control por álgebra lineal y control por modos deslizantes. Estas técnicas se explican a continuación.

1.8.2.1 Control basado en Mínima Norma

El concepto de mínima norma se explica a partir de la aproximación por mínimos cuadrados, explicado en la sección 1.7.13. Un sistema $Ax = b$ puede no tener solución, sin embargo, es factible encontrar la mejor solución posible, es decir el valor de x para el cual la distancia entre A y b sea mínima y esto requiere que la norma $\|Ax - b\|$ sea minimizada [20].

Como se mencionó en la sección 1.5.1, el manipulador móvil tiene un alto grado de redundancia cinemática, esta redundancia es muy deseable para funciones de manipulación y transporte, ya que permite al sistema ser posicionado de manera óptima y configurado para un máximo rendimiento [13]. Un controlador basado en la solución por mínima norma, supone que en cualquier momento el sistema alcanzará su objetivo con el menor número de movimientos posible [27].

1.8.2.2 Control basado en Álgebra Lineal [28]

Muchos sistemas de la vida real pueden ser representados mediante ecuaciones diferenciales, cuya solución no es exacta, estos sistemas de ecuaciones diferenciales de orden n pueden ser convertidos en sistemas de ecuaciones de primer orden, los cuales son mucho más fáciles de resolver numéricamente, es decir mediante un software computacional. Esta es la ventaja de diseñar controladores mediante álgebra lineal. A continuación se presenta un procedimiento general para el diseño.

Se parte de la siguiente ecuación diferencial.

$$\dot{y} = f(y, u, t) \text{ con } y(0) = y_0 \quad (1.3)$$

La salida del sistema del controlador es y , la acción de control es expresada por u y t es el tiempo. En tiempo discreto las variables son y_k y T_0 . Donde k es una constante.

$k \in \{0,1,2,3, \dots\}$ y T_0 es el tiempo de muestreo.

$$t = kT_0$$

Para obtener el valor siguiente y_{k+1} conociendo el valor actual y_k , se integra la ecuación (1.1) en el intervalo de tiempo $kT_0 \leq t \leq (k+1)T_0$, es decir se aplica una aproximación de Euler.

$$y_{k+1} = y_k + \int_{kT_0}^{(k+1)T_0} f(y, u, t) dt \quad (1.4)$$

Realizando la integración de la ecuación anterior se obtiene una aproximación cuyo resultado se expresa.

$$y_{k+1} \cong y_k + T_0 f(y_k, u_k, t_k) \quad (1.5)$$

De esta manera se calcula el valor aproximado de y_{k+1} , conociendo el valor de k . En y_{k+1} se puede remplazar una trayectoria deseada para posteriormente calcular las acciones de control que permitan que la salida del sistema vaya de su valor actual a su valor deseado.

1.8.2.3 Control por Modos Deslizantes (CMD)

Es una técnica que se deriva de control por estructura variable, las técnicas de control por modos deslizantes (CMD) se utilizan debido a la robustez que presentan al aplicar el control en sistemas no lineales y variantes en el tiempo. [29]

En el presente proyecto se desarrollan dos controladores basados en esta técnica de control, la cual se explica a continuación.

La idea del control CMD es definir una superficie a lo largo de la cual el proceso pueda deslizarse hacia su valor final deseado [29], esta técnica de control implica la solución de dos problemas, en primer lugar establecer una función del error $\tilde{h}(t)$ conocida como superficie de deslizamiento, la misma que rige el comportamiento del sistema, dicha

superficie debe ser estable y puede ser lineal. En la Figura 1.18 se presenta una interpretación gráfica del CMD, como se puede observar la superficie deslizante $s(t)$ es una función del error $\tilde{h}(t)$, el estado X_0 alcanza la superficie $s(t)$ y luego permanece sobre ella.

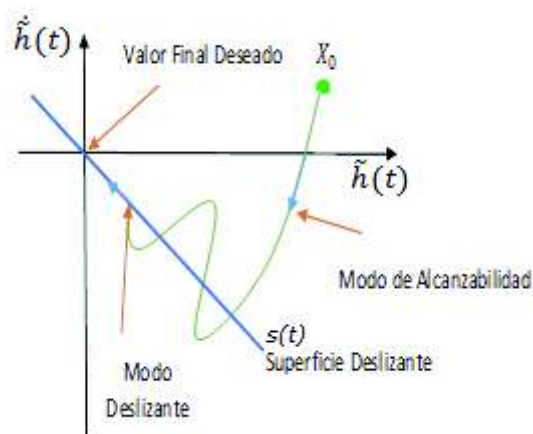


Figura 1.18. Interpretación Gráfica de un CMD

Después de definir la superficie de deslizamiento, se procede a diseñar una ley de control que garantice que las trayectorias cercanas hacia la superficie de deslizamiento conduzcan hacia la superficie [30]. Este control implica una ley de control por conmutación a frecuencia infinita, de esta manera las trayectorias antes que deslizar presentan una oscilación a lo largo de la superficie de deslizamiento $s(t)$, como se muestra en la Figura 1.19. [31]

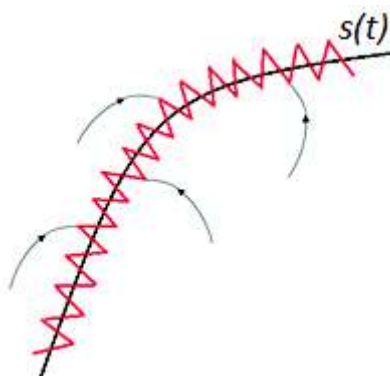


Figura 1.19. Modo deslizante real

La superficie de deslizamiento se define como una función del error y puede ser de tipo proporcional, integral o diferencial dependiendo del tipo del sistema [29]. Una forma de definir la superficie de deslizamiento se muestra en (1.6):

$$s(t) = \left(\frac{d}{dt} + \lambda \right)^n \int \tilde{h}(t) dt \quad (1.6)$$

Donde $s(t)$ es la superficie deslizante, n es el orden del sistema a controlar y λ es una matriz de ganancia definida positiva.

Se puede definir superficies deslizantes tipo proporcional integral (PI), proporcional diferencial (PD) y proporcional integral diferencial (PID), dependiendo del orden del sistema. Otra forma de definir una superficie se muestra en (1.7), la cual se utiliza para superficies tipo proporcional (P).

$$s(t) = \left(\frac{d}{dt} + \lambda \right)^{n-1} \tilde{h}(t) \quad (1.7)$$

El objetivo de control es asegurar que la variable controlada sea igual al valor de referencia, es decir que el error y sus derivadas en el tiempo sean cero [34].

Una vez definida la superficie de deslizamiento, la atención se centra en desarrollar la ley de control $U(t)$, la cual se divide en dos partes aditivas, una parte continua $v_c(t)$ en función de la variable a controlar $x(t)$ y el valor de referencia $r(t)$, es decir $v_c(t) = f(x(t), r(t))$ y una parte discontinua $v_D(t)$ que incorpora elementos no lineales, que incluye el elemento de conmutación de la ley de control, estos elementos originan un fenómeno llamado chattering que son oscilaciones de alta frecuencia [34]., la ley de control mencionada está definida por:

$$U(t) = v_c(t) + v_D(t) \quad (1.8)$$

En la parte continua $v_c(t)$, para que la dinámica del sistema permanezca en la superficie de deslizamiento, la condición es que la derivada temporal de la superficie sea igual a cero, es decir $\dot{s}(t) = 0$, de esta forma se garantiza que la trayectoria sea tangente a la superficie de deslizamiento [32].

La parte discontinua está conmutando sobre la superficie de deslizamiento a gran velocidad, su acción transitoria se representa generalmente por:

$$u_D(t) = K_D \text{sign}(s(t)) \quad (1.9)$$

El parámetro de ajuste responsable del alcance es $K_D \text{sign}(s(t))$, es decir una función no lineal expresada en función de $s(t)$. Esta función debe cumplir con la condición de alcanzabilidad, tiene que ser muy rápida con el fin de llegar a la superficie de deslizamiento y cumplir la condición $s(t)\dot{s}(t) = 0$ [23].

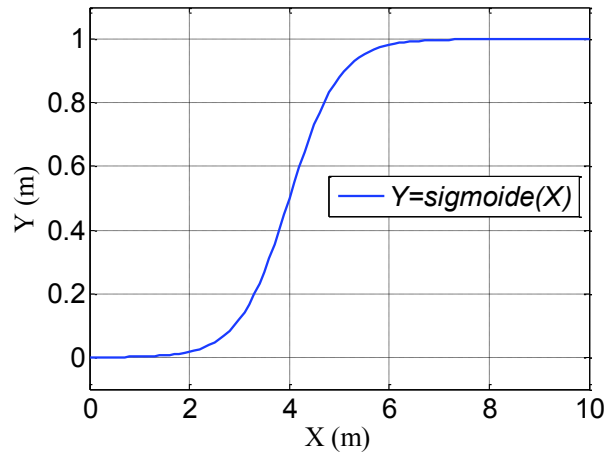


Figura 1.20. Función sigmoide

Para eliminar el efecto de chattering que se presenta por efecto de las oscilaciones, se aplica una función sigmoide, como la mostrada en la Figura 1.20, la cual permite el suavizado de la curva:

$$v_D(t) = K_D \frac{s(t)}{|s(t)| + \delta} \quad (1.10)$$

Donde, δ es un parámetro de ajuste que permite eliminar las oscilaciones de alta frecuencia, y K_D es un parámetro de ajuste del alcance [29].

1.8.3 OBJETIVOS DE CONTROL

El objetivo del presente proyecto es resolver los problemas de control de posición y seguimiento de trayectorias, a continuación, se detalla cada uno de estos conceptos.

1.8.3.1 Control de posición

Los problemas principales en el control de movimiento de robots móviles, según [33], se pueden clasificar en tres grupos: control de posición, seguimiento de caminos y seguimiento de trayectorias.

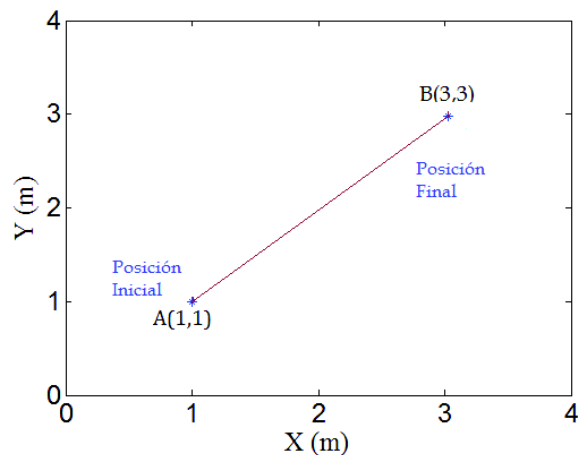


Figura 1.21. Control de posición

El control de posición consiste en ubicar al robot en un punto de referencia dado, con una orientación deseada [6].

La Figura 1.21 muestra un ejemplo de control de posición, donde el objetivo de control consiste ubicar el robot desde la posición inicial en el punto A, hasta la posición deseada en el punto B.

1.8.3.2 Seguimiento de trayectoria

El control de seguimiento de trayectoria en robots móviles consiste en conseguir que el robot siga como referencia a una curva parametrizada en el tiempo. El objetivo de este tipo de controlador, es conseguir mediante una ley de control, que el robot alcance y siga con error cero estados deseados variantes con el tiempo, dichos estados describen la trayectoria deseada a seguir [6].

Las trayectorias, objetivo de seguimiento de los controladores diseñados en el presente proyecto se forman a partir de un conjunto de coordenadas en el plano cartesiano (x, y, z) variantes en el tiempo. Este conjunto de coordenadas se obtienen de funciones de la librería de la herramienta Simulink de Matlab.

1.8.3.3 Seguimiento de Caminos

Un camino es una sucesión de líneas que unen puntos ubicados en un plano, o en el espacio [6].

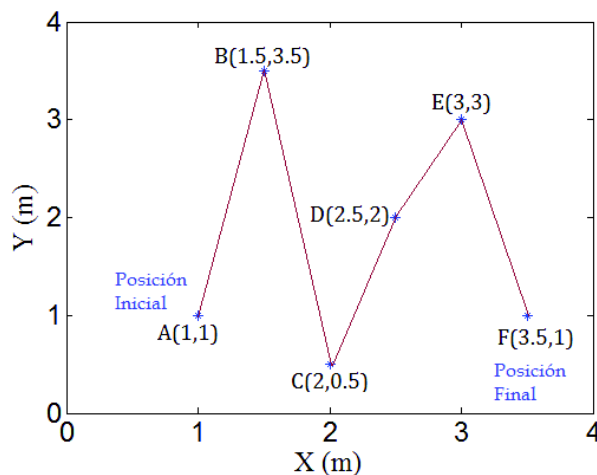


Figura 1.22. Seguimiento de Caminos

Un camino también puede provenir de la discretización de una curva continua que interpola ciertos puntos de la misma, como se muestra en la Figura 1.22.

El control de seguimiento de caminos implica que el robot siga el camino sin ninguna especificación temporal, a diferencia del seguimiento de trayectorias, donde los puntos describen una función del tiempo. [6]

1.9 RENDIMIENTO DE LOS SISTEMAS DE CONTROL [34]

Para evaluar un sistema de control, hay que analizar su evolución en el tiempo, para ello existen criterios de desempeño que permiten evaluar el comportamiento de los sistemas y son la base para optimizarlos respecto del comportamiento de su error en el transitorio, uno de los criterios utilizados es el criterio integral del error absoluto (IAE).

1.9.1 CRITERIO INTEGRAL DEL ERROR ABSOLUTO (IAE)

Viene dado por la siguiente expresión:

$$IAE = \int |e(t)| dt \quad (1.11)$$

Una de las características de este criterio es su fácil aplicación, este indicador mide el área bajo la curva del error, por lo que mientras más bajo sea su valor, reflejará un controlador más estable y de mejor rendimiento.

1.10 ESTABILIDAD EN SISTEMAS ROBÓTICOS [9]

El concepto de estabilidad puede tener varias interpretaciones dependiendo del área de la ciencia donde se aplique. En ingeniería la estabilidad de un sistema implica que el error, después de un intervalo de tiempo, se mantenga dentro de un rango o se reduzca a cero. En el estudio de la estabilidad de un sistema existen varios criterios de estabilidad como el criterio de estabilidad de Bode, de Nyquist o de Routh Hurwitz, la limitación de estos criterios radica en que su aplicación se limita a sistemas lineales.

Para el estudio de estabilidad de sistemas no lineales, la teoría que mayor ventaja ofrece es la teoría de estabilidad de Lyapunov, dado que proporciona varias propiedades matemáticas que facilitan el análisis y diseño de esquemas de control.

1.10.1 TEORÍA DE ESTABILIDAD DE LYAPUNOV

Introducida en 1892 por el ruso Aleksandr Mikhailovich Lyapunov (1857-1918), la teoría de estabilidad de Lyapunov detonó el desarrollo de control en robótica a partir de la década de 1950, cuando se reconoció su legado científico, el cual sentó las bases del control moderno.

Hoy en día es una herramienta indispensable para el análisis de estabilidad de sistemas, especialmente sistemas no lineales, puesto que, sin importar el orden del sistema, permite obtener información sobre su estabilidad sin resolver la ecuación diferencial que lo caracteriza, resolver esta ecuación en el caso de muchos sistemas no lineales resulta una difícil, debido a la complejidad del sistema [9].

En la teoría de estabilidad de Lyapunov se trabaja con funciones conocidas como funciones de Lyapunov, que se interpretan como representaciones de la energía del sistema, a continuación, se presentan algunas definiciones para entender este tipo de funciones.

1.10.1.1 Función definida positiva

Este tipo de funciones son ampliamente utilizadas en el diseño de algoritmos de control de robots, se interpretan como la inyección de energía aplicada al robot para moverlo desde su posición inicial hasta un punto deseado [9].

En la Figura 1.24 se presenta un ejemplo del gráfico de función definida positiva.

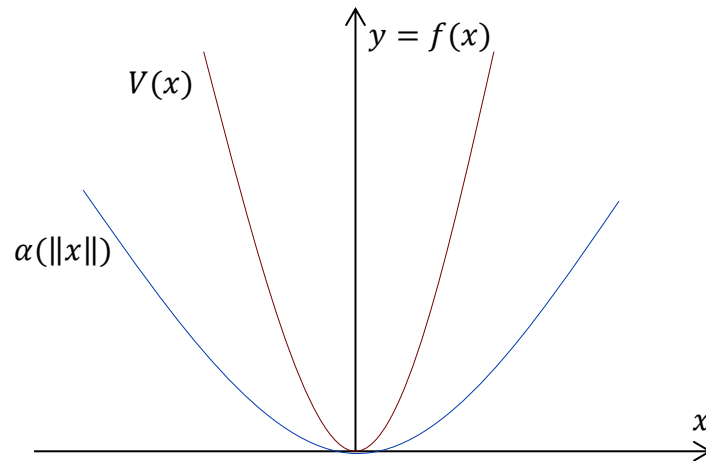


Figura 1.24 Gráfica de una función definida positiva

Una función $V(x)$ en el conjunto de los números reales \mathbb{R}^n , es definida positiva si cumple con las siguientes condiciones:

- Es una función continua en su argumento x , $x \in \mathbb{R}^n$
- Solo puede ser igual a cero cuando su argumento es cero, $V(x) = 0 \Leftrightarrow x = 0 \in \mathbb{R}^n$
- Para cualquier valor de su argumento diferente de cero, siempre será positiva, es decir $V(x) > 0 \forall x \neq 0$

1.10.1.2 Función semidefinida positiva

La Figura 1.25 muestra un ejemplo de gráfica de una función semidefinida positiva.

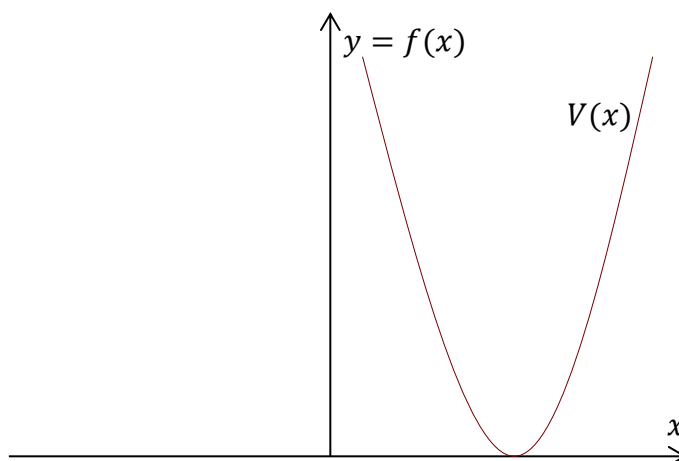


Figura 1.25 Gráfica de una función semidefinida positiva

Una función $V(x)$ en el conjunto de los números reales \mathbb{R}^n , es semidefinida positiva si cumple con las siguientes condiciones:

- Es una función continua en su argumento x , $x \in \mathbb{R}^n$
- Existe un valor no nulo de su argumento, para el cual la función es igual a cero, $V(x) = 0 \Leftrightarrow x \neq 0 \in \mathbb{R}^n$
- Para cualquier valor de su argumento, siempre será mayor o igual que cero, $V(x) \geq 0 \forall x$

1.10.1.3 Función definida negativa

Una función $V(x)$ en el conjunto de los números reales \mathbb{R}^n , es definida negativa si cumple con las siguientes condiciones:

- Es una función continua en su argumento x , $x \in \mathbb{R}^n$
- Solo puede ser igual a cero cuando su argumento es cero, $V(x) = 0 \Leftrightarrow x = 0 \in \mathbb{R}^n$
- Para cualquier valor de su argumento diferente de cero, siempre será negativa, es decir $V(x) < 0 \forall x \neq 0$

La Figura 1.26 muestra un ejemplo de gráfica de función definida negativa

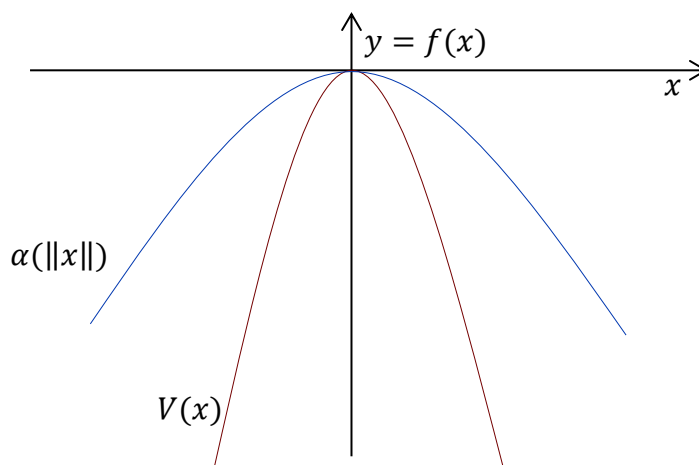


Figura 1.26 Gráfica de una función definida negativa

1.10.1.4 Función semidefinida negativa

Una función $V(x)$ en el conjunto de los números reales \mathbb{R}^n , es semidefinida negativa si cumple con las siguientes condiciones:

- Es una función continua en su argumento x , $x \in \mathbb{R}^n$
- Existe un valor no nulo de su argumento, para el cual la función es igual a cero, $V(x) = 0 \Leftrightarrow x \neq 0 \in \mathbb{R}^n$
- Para cualquier valor de su argumento, siempre será menor o igual que cero, $V(x) \leq 0$

La Figura 1.27 muestra la gráfica de un ejemplo de función semidefinida negativa.

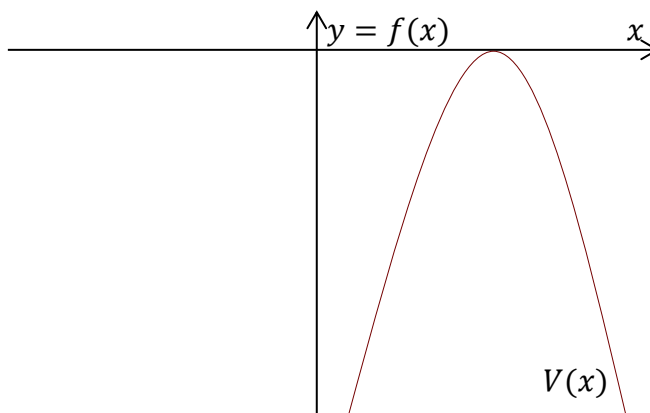


Figura 1.27 Gráfica de una función semidefinida negativa

La figura 1.28 muestra la gráfica de un ejemplo de función semidefinida negativa.

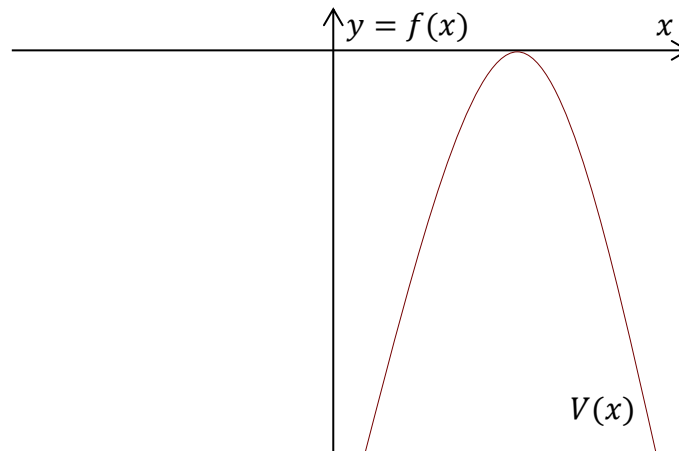


Figura 1.28 Gráfica de una función semidefinida negativa

1.10.1.5 Función candidata de Lyapunov

Es una función cuya estructura matemática permite demostrar la estabilidad de un modelo. Una función $V(x)$ es una función candidata de Lyapunov al cumplir con lo siguiente:

- Es una función definida positiva.
- Su derivada parcial con respecto a x , $\frac{\partial V(x)}{\partial x}$ es una función continua de x .
- Su derivada temporal, $\frac{dV(x)}{dt}$ existe, es una función continua respecto a x , y además es una función definida negativa, es decir, $\dot{V}(x) < 0 ; \forall t \geq 0 ; \forall x \in \mathbb{R}^n$

Es importante mencionar que la selección de una función candidata de Lyapunov, tal que satisfaga las condiciones descritas, no es trivial, depende en gran medida de la intuición y experiencia del diseñador [9].

Con estos preliminares se puede entender de mejor manera la teoría de Lyapunov, la cual se refiere principalmente a los dos métodos propuestos por Lyapunov: el método indirecto y el directo. El método utilizado en este proyecto es el método directo, presentado a continuación.

1.10.1.6 Método Directo de Lyapunov

También conocido como segundo método de Lyapunov, en principio, el método directo enfoca la estabilidad desde una generalización de los conceptos de energía asociados con sistemas mecánicos. La interpretación física de este método se resume en que un sistema es estable si su energía total se reduce continuamente hasta alcanzar su estado de equilibrio [35].

$$\dot{E} < 0$$

La idea de este método es construir una función candidata de Lyapunov, que represente la energía del sistema, con la finalidad de observar si ésta decrece [36].

1.10.1.7 Estabilidad en el sentido de Lyapunov

Los criterios de estabilidad de Lyapunov están referidos a las trayectorias de un sistema cuando el estado inicial está próximo al punto de equilibrio [35].

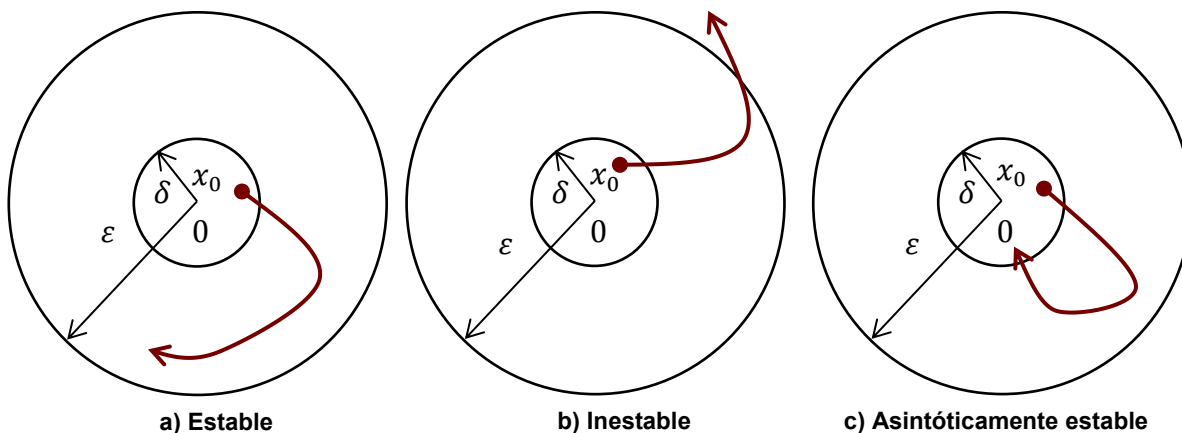


Figura 1.29 Características de estabilidad de sistemas no lineales

En la Figura 1.29 se presentan los casos de estabilidad en el sentido de Lyapunov. Un punto de equilibrio de un sistema dinámico puede ser:

a) Estable en el sentido de Lyapunov si todas las soluciones que nacen en las cercanías del punto x_0 (región delimitada por δ), permanecen en dichas cercanías (región delimitada por ϵ).

b) Inestable si las soluciones no permanecen en las cercanías del punto x_0 .

c) Asintóticamente estable si las soluciones además de permanecer en las cercanías del punto x_0 , tienden hacia el punto de equilibrio 0 a medida que transcurre el tiempo.

1.10.1.8 Teorema de estabilidad

Sea $V(x)$ una función candidata de Lyapunov, y $\dot{V}(x)$ su derivada temporal

- Si $\dot{V}(x) < 0$ la energía disminuye y llega a un valor pequeño, el punto de equilibrio es estable y si llega a cero es asintóticamente estable.
- Si $\dot{V}(x) > 0$ la energía se incrementa indefinidamente, por lo que el sistema es inestable [35].

Por tanto, el objetivo de Lyapunov es establecer una función de energía definida positiva V y analizar su derivada. Cuando $\dot{V} < 0$, la energía disminuye y llega a un valor pequeño, el punto de equilibrio es estable, y si llega a cero es asintóticamente estable [36].

La limitación del método directo recae en que por lo general, es difícil encontrar una función candidata de Lyapunov para el sistema analizado. Los resultados o teoremas de estabilidad de Lyapunov proveen de condiciones suficientes para establecer la estabilidad de los puntos de equilibrio [9].

1.11 SIMULACIÓN DE ROBOTS EN COMPUTADOR

La simulación mediante programas computacionales se ha convertido en una herramienta fundamental a la hora de diseñar y analizar todo tipo de procesos, por ejemplo, procesos que involucran sistemas mecánicos en movimiento, como es el caso de los robots [37]. Algunos programas de simulación incluyen entornos de realidad virtual, una tecnología considerada una extensión de los gráficos 3D por computadora, para diferentes aplicaciones en ingeniería [38], que proveen una experiencia realista al usuario.

El software de simulación Matlab facilita en gran medida el análisis de sistemas mecánicos en movimiento [37], gracias a su estructura que permite el desarrollo de algoritmos y funciones a partir de sus funciones básicas. Además, Matlab provee un entorno de simulación dinámica con prestaciones gráficas conocido como Simulink que permite la programación utilizando bloques funcionales y conexiones gráficas [2]. Por otro lado, permite trabajar con un entorno tridimensional de realidad virtual [37]. En el presente proyecto se utiliza Simulink para la simulación de los modelos del robot y de los sistemas de control, adicionalmente y para aportar a la visualización de los resultados se presenta una simulación en realidad en Matlab utilizando un modelo gráfico del robot creado en el software Solidworks, programa de diseño asistido por computadora (CAD). El desarrollo de los modelos matemáticos, los controladores y las pruebas de simulación se presentan en los siguientes capítulos.

A continuación se presenta una breve explicación de los programas computacionales utilizados para la simulación del manipulador móvil.

1.11.1 MATLAB [37]

El “Matrix Laboratory” (Laboratorio Matricial), es una herramienta de software matemático, cuyo elemento básico es una matriz que no necesita ser dimensionada, una de las ventajas de trabajar con matrices es que se realizan cálculos en un menor rango de tiempo, se lo usa generalmente para análisis numérico, cálculo matricial, programación, procesamiento de señales y gráficos.

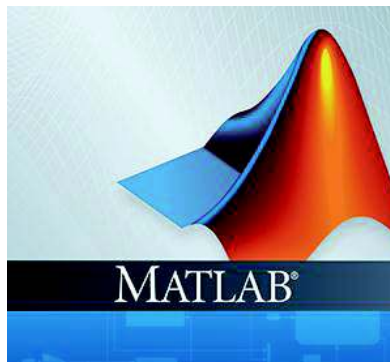


Figura 1.28. Logotipo de MATLAB [37]

Cada año, ofrece más y mejores aplicaciones, las mismas que son adaptadas a las necesidades que se presentan actualmente, en el área educativa es una herramienta de enseñanza para cursos de álgebra lineal aplicada y materias avanzadas en el área de matemática, en la parte industrial se emplea para realizar investigación de procesos y resolver problemas de ingeniería.

El paquete computacional está compuesto por funciones específicas llamadas TOOLBOXES, extendiendo el entorno de trabajo de este software en la solución de problemas específicos como: procesamiento de señales, diseño y simulación de sistemas de control, etc.

Simulink es una herramienta de MATLAB que permite modelar y simular sistemas dinámicos, se basa en diagrama de bloques para simular sistemas y diseñar modelos matemáticos, lo cual permite un diseño y simulación a nivel de sistema y la generación automática de código.

Los diagramas de bloque permiten comprender el sistema mediante una representación gráfica, estos bloques son representaciones individuales de cada componente del sistema, los cuales se interconectan mediante un flujo de datos.

Debido a que se integra con MATLAB permite utilizar algoritmos en este lenguaje dentro de modelos de Simulink, así como exportar datos al espacio de trabajo de MATLAB para su posterior análisis. Permitiendo simular y modelar procesos lineales y no lineales dependientes del tiempo.

En los capítulos posteriores se describe a detalle los algoritmos implementados en este software.

1.11.2 SOLIDWORKS [39]

Es un software de diseño asistido por computadora (CAD), creada para el diseño y modelado mecánico en 3D, cada modelo está compuesto por elementos individuales a los que se denomina operaciones. Estas operaciones son de tipo geométrica y de fácil entendimiento.



Figura 1.29. Logotipo de SolidWorks [39]

El fin de este software es que el usuario pueda diseñar cualquier pieza mecánica de manera sencilla, por lo cual presenta un interfaz muy amigable, con la que puede realizar diferentes tipos de operaciones para llegar a obtener la pieza mecánica deseada. El software permite aplicar operaciones de corte, taladrado, redondeo, chaflán, entre otras, dependiendo de la necesidad de la pieza a construir.

El fin de este software es que el usuario pueda diseñar cualquier pieza mecánica de manera sencilla, por lo cual presenta un interfaz muy amigable, con la que puede realizar diferentes tipos de operaciones para llegar a obtener la pieza mecánica deseada. El software permite aplicar operaciones de corte, taladrado, redondeo, chaflán, entre otras, dependiendo de la necesidad de la pieza a construir.

Estas operaciones parten de un modelo 2D, al cual se aplica una operación de extrusión, rotación o barrido para obtener el sólido.

En el Anexo B se detalla el diseño y desarrollo de un manipulador móvil en este software.

CAPÍTULO 2

ESTUDIO Y MODELADO DEL MANIPULADOR MÓVIL

En este capítulo se describe el modelado del manipulador móvil analizado, para la obtención del modelo cinemático y dinámico, se estudia en primer lugar la cinemática de un robot móvil tipo unicycle, luego se obtiene el modelo cinemático del manipulador de tres grados de libertad, después ambos modelos se unen para obtener las ecuaciones que describen la cinemática del manipulador móvil.

2.1 MODELO CINEMÁTICO

Existe gran variedad de modelos matemáticos para representar al robot móvil tipo unicycle, cada uno de ellos refleja diferentes características del robot, de igual manera cada modelo será de utilidad dependiendo del parámetro o de la característica que se desee observar [6].

En el presente trabajo se considera que en el modelo cinemático se desea conocer la velocidad lineal y angular con la que se desplaza el robot, así como su posición y orientación.

Así como en el caso del robot móvil, también existen varios métodos para obtener el modelo cinemático del manipulador, por ejemplo, se puede aplicar definiciones geométricas al mecanismo del robot para obtener sus ecuaciones cinemáticas de manera sencilla, sin embargo, cuando aumenta el número de grados de libertad este método se vuelve largo de desarrollar.

En el presente proyecto se obtiene el modelo cinemático del manipulador por geometría.

2.1.1 CINEMÁTICA DEL ROBOT MÓVIL TIPO UNICYCLO

Se considera el robot móvil tipo unicycle de la Figura 2.1, donde $\{R\}$ es el sistema de coordenadas globales y la posición de robot está definida por el punto G, que se encuentra ubicado a una distancia a del centro del eje que une las ruedas [6].

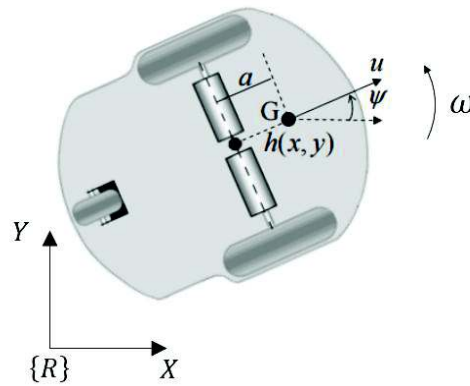


Figura 2.1. Robot móvil tipo unicycle, [6]

El punto G representa el centro de masa, $h(x, y)$ es la posición del punto G respecto al sistema de coordenadas global y ψ es la orientación de G respecto a X . La restricción no holonómica de velocidad lineal del robot explicada en el capítulo anterior está dada por:

$$\dot{x} \sin(\psi) - \dot{y} \cos(\psi) + a\dot{\psi} = 0 \quad (2.1)$$

Donde, \dot{x} e \dot{y} representan las variaciones temporales de la posición en los ejes X e Y del sistema de coordenadas globales $\{R\}$ respectivamente, además $\dot{\psi}$ es igual a la velocidad angular ω del robot, y u es la velocidad lineal del robot.

Al tomar en cuenta la restricción descrita en (2.1), el modelo cinemático del robot móvil tipo unicycle queda definido por:

$$\begin{cases} \dot{x} = \mu \cos(\psi) - a\omega \sin(\psi) \\ \dot{y} = \mu \sin(\psi) + a\omega \cos(\psi) \\ \dot{\psi} = \omega \end{cases} \quad (2.2)$$

Donde $[x, y, \psi]$ son los estados que representan el movimiento del robot, mientras que $[u, \omega]$ son las entradas [6].

2.1.2 CINEMÁTICA DEL BRAZO MANIPULADOR

El modelo cinemático se obtendrá de un manipulador de tres grados de libertad, el que se representa en la Figura 2.2.

Donde, l_1, l_2 y l_3 son las longitudes de los eslabones primero, segundo y tercero respectivamente, θ_1, θ_2 y θ_3 , son los ángulos de rotación de cada articulación, el origen del manipulador se encuentra en el punto x_0, y_0, z_0 .

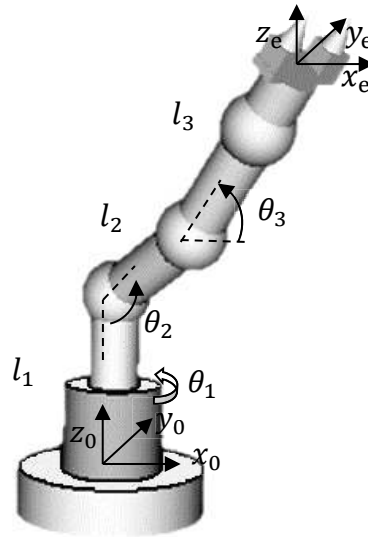


Figura 2.2. Manipulador de tres grados de libertad

La posición del efector final es x_e, y_e, z_e . El objetivo del modelo cinemático, en este caso es obtener la posición del efector final (x_e, y_e, z_e) con respecto a la base. Para visualizar de mejor manera las propiedades geométricas aplicadas, se presenta a continuación las proyecciones en los diferentes planos.

En la Figura 2.3 se presenta el manipulador visto desde el plano XZ para poder realizar el análisis de sus propiedades geométricas.

En el plano XZ se pueden aplicar propiedades trigonométricas para obtener la proyección de l_2 sobre el eje X .

$$\cos(\theta_2) = \frac{l_{2x}}{l_2} \rightarrow l_{2x} = l_2 \cos(\theta_2) \quad (2.3)$$

Para la proyección de l_3 sobre el eje X se observa que esta no depende solo del ángulo θ_3 , sino también del ángulo θ_2 , por lo que este se incluye en (2.4).

$$\cos(\theta_2 + \theta_3) = \frac{l_{3x}}{l_3} \rightarrow l_{3x} = l_3 \cos(\theta_2 + \theta_3) \quad (2.4)$$

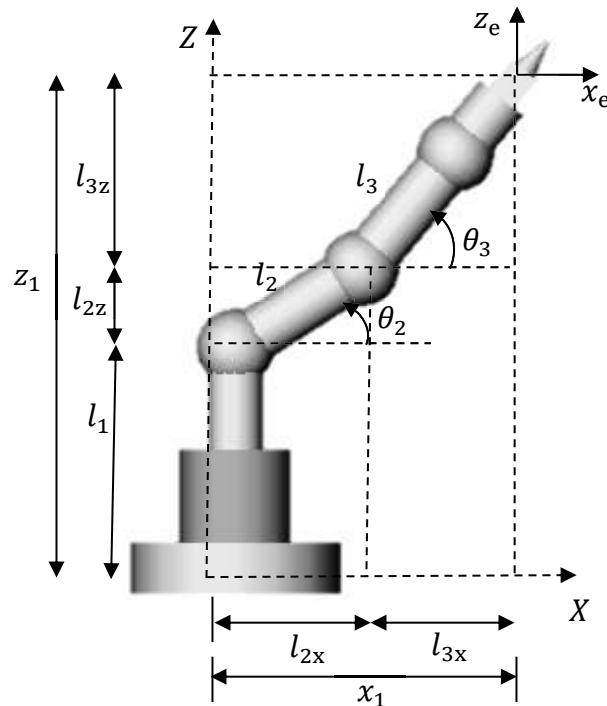


Figura 2.3. Proyección del manipulador en el plano XZ

La suma de estas proyecciones da como resultado la posición x_1 del efector final, que está formado por la suma de los eslabones l_2 y l_3 proyectados sobre el eje X .

$$x_1 = l_2 \cos(\theta_2) + l_3 \cos(\theta_2 + \theta_3) \quad (2.5)$$

Para la proyección sobre el eje z , se procede de la misma manera, aplicando propiedades trigonométricas y tomando en cuenta que el eslabón l_1 se proyecta completamente en el eje z , finalmente se obtiene la posición z_1 del efector final.

$$z_1 = l_1 + l_2 \sin(\theta_2) + l_3 \sin(\theta_2 + \theta_3) \quad (2.6)$$

Para incluir la acción del ángulo θ_1 sobre la posición del efector final, se obtienen las proyecciones en el plano XY , como se muestra en Figura 2.4.

Aplicando trigonometría, la expresión para la posición del efector final x_e queda definida por:

$$x_e = \cos(\theta_1)[x_1] \quad (2.7)$$

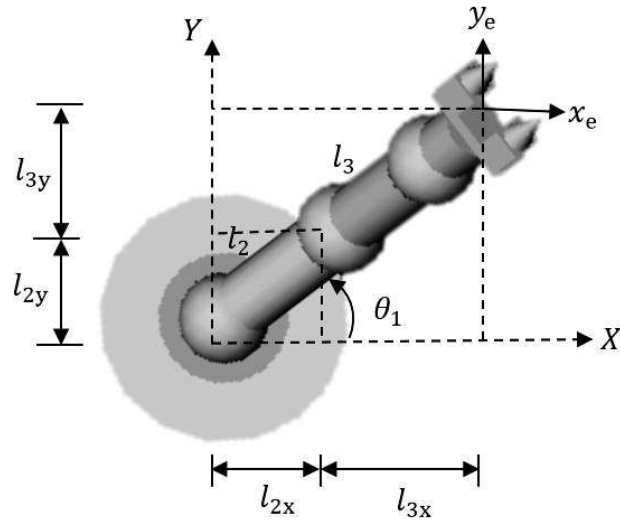


Figura 2.4. Proyección del manipulador en el plano XY

Reemplazando (2.5) en (2.7)

$$x_e = \cos(\theta_1)[l_2 \cos(\theta_2) + l_3 \cos(\theta_2 + \theta_3)] \quad (2.8)$$

Como se observa, la acción del ángulo θ_1 influye en las posiciones en x_e e y_e , por lo que se repite el procedimiento y se obtiene la posición y_e .

$$y_e = \sin(\theta_1)[l_2 \cos(\theta_2) + l_3 \cos(\theta_2 + \theta_3)] \quad (2.9)$$

Como se observa el ángulo θ_1 no influye en la posición en z , con lo que la cinemática del manipulador queda definida por:

$$\begin{cases} x_e = \cos(\theta_1)[l_2 \cos(\theta_2) + l_3 \cos(\theta_2 + \theta_3)] \\ y_e = \sin(\theta_1)[l_2 \cos(\theta_2) + l_3 \cos(\theta_2 + \theta_3)] \\ z_e = l_1 + l_2 \sin(\theta_2) + l_3 \sin(\theta_2 + \theta_3) \end{cases} \quad (2.10)$$

2.1.3 CINEMÁTICA DEL MANIPULADOR MÓVIL DE TRES GRADOS DE LIBERTAD

Para obtener el modelo cinemático del manipulador móvil de tres grados de libertad, se utiliza una metodología similar a la utilizada en la obtención del modelo cinemático del brazo manipulador, es decir deducir el modelo cinemático a partir de las relaciones geométricas del manipulador móvil de tres grados de libertad de la Figura 2.5.

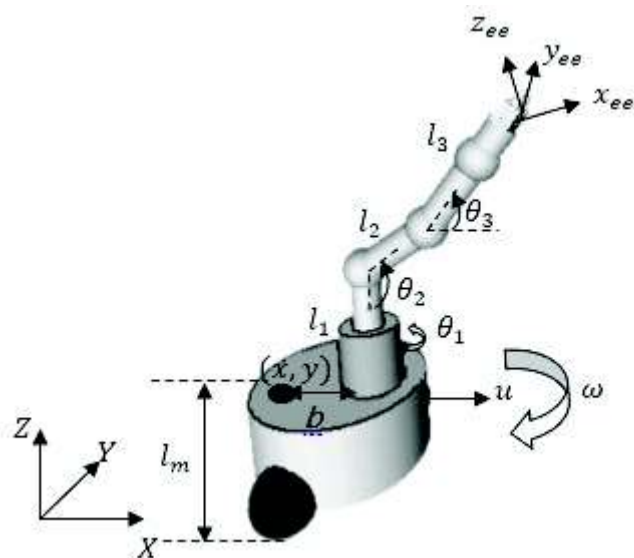


Figura 2.5. Manipulador Móvil de tres grados de libertad

Para el análisis se considera un punto de coordenadas (x, y) , con respecto al sistema de referencia (X, Y, Z) , este punto coincide con el centro del eje de las ruedas.

El brazo manipulador se encuentra ubicado a una distancia b del punto (x, y) . Finalmente la altura de la base móvil es l_m . El objetivo del modelo cinemático es obtener la posición del efector final (x_{ee}, y_{ee}, z_{ee}) con respecto al origen de coordenadas (X, Y, Z) .

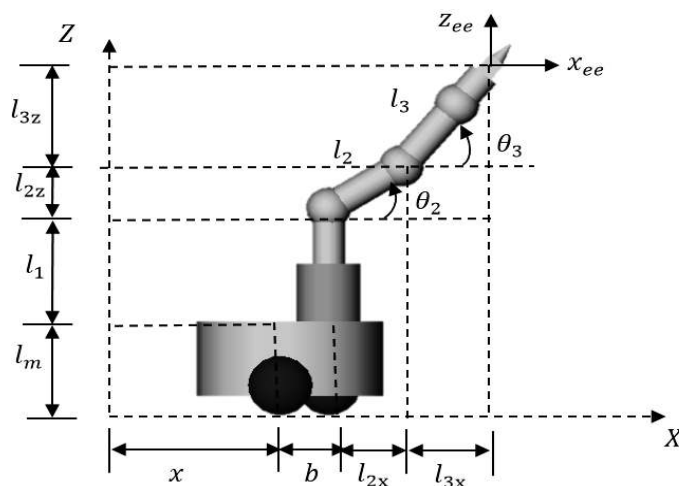


Figura 2.6. Proyección del manipulador móvil en XZ

Para la posición del efector final se toma en cuenta la posición x del robot y la proyección de la distancia b , como se indica en la Figura 2.6, mientras que para el eje

Z se toma en consideración la altura de la base móvil. El análisis es similar para la proyección en el plano XY, como se muestra en la Figura 2.7, aquí se puede notar la influencia del ángulo de rotación de la base móvil ψ .

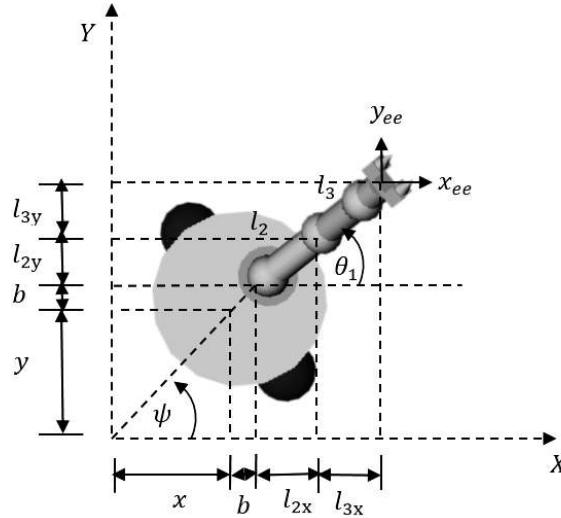


Figura 2.7. Proyección del manipulador móvil en XY

Con lo que las relaciones geométricas del manipulador móvil quedan dadas por:

$$x_{ee} = x + b \cos(\psi) + \cos(\theta_1 + \psi) [l_2 \cos(\theta_2) + l_3 \cos(\theta_2 + \theta_3)] \quad (2.11)$$

$$y_{ee} = y + b \sin(\psi) + \sin(\theta_1 + \psi) [l_2 \cos(\theta_2) + l_3 \cos(\theta_2 + \theta_3)] \quad (2.12)$$

$$z_{ee} = (l_1 + l_m) + l_2 \sin(\theta_2) + l_3 \sin(\theta_2 + \theta_3) \quad (2.13)$$

Para expresar el sistema en función de entradas y salidas, se recurre a la matriz Jacobiana J_{ee} explicada en la sección 1.6.1.3.

$$\dot{h}_{ee} = J_{ee} \dot{q}_{ee} \quad (2.14)$$

Donde, $h_{ee} = [x_{ee} \ y_{ee} \ z_{ee}]^T$ es el vector de posiciones del efector final, el vector de velocidades lineales del efector final es $\dot{h}_{ee} = [\dot{x}_{ee} \ \dot{y}_{ee} \ \dot{z}_{ee}]^T$, el cual es el vector de salidas del sistema, $q_{ee} = [x \ y \ \psi \ \theta_1 \ \theta_2 \ \theta_3]^T$ es el vector de posiciones y ángulos del manipulador móvil y $\dot{q}_{ee} = [\dot{x} \ \dot{y} \ \dot{\psi} \ \dot{\theta}_1 \ \dot{\theta}_2 \ \dot{\theta}_3]^T$ es el vector de variación temporal de posición y ángulo y representa las entradas del sistema.

El Jacobiano J_{ee} por definición, viene dado por:

$$J_{ee} = \frac{\partial h_{ee}}{\partial q_{ee}} = \begin{bmatrix} \frac{\partial x_{ee}}{\partial x} & \frac{\partial x_{ee}}{\partial y} & \frac{\partial x_{ee}}{\partial \psi} & \frac{\partial x_{ee}}{\partial \theta_1} & \frac{\partial x_{ee}}{\partial \theta_2} & \frac{\partial x_{ee}}{\partial \theta_3} \\ \frac{\partial y_{ee}}{\partial x} & \frac{\partial y_{ee}}{\partial y} & \frac{\partial y_{ee}}{\partial \psi} & \frac{\partial y_{ee}}{\partial \theta_1} & \frac{\partial y_{ee}}{\partial \theta_2} & \frac{\partial y_{ee}}{\partial \theta_3} \\ \frac{\partial z_{ee}}{\partial x} & \frac{\partial z_{ee}}{\partial y} & \frac{\partial z_{ee}}{\partial \psi} & \frac{\partial z_{ee}}{\partial \theta_1} & \frac{\partial z_{ee}}{\partial \theta_2} & \frac{\partial z_{ee}}{\partial \theta_3} \end{bmatrix} \quad (2.15)$$

Desarrollando las derivadas de (2.11) se tiene:

$$\frac{\partial x_{ee}}{\partial x} = 1 \quad (2.16)$$

$$\frac{\partial x_{ee}}{\partial y} = 0 \quad (2.17)$$

$$\frac{\partial x_{ee}}{\partial \psi} = -b \sin(\psi) - \sin(\theta_1 + \psi) [l_2 \cos(\theta_2) + l_3 \cos(\theta_2 + \theta_3)] \quad (2.18)$$

$$\frac{\partial x_{ee}}{\partial \theta_1} = -\sin(\theta_1 + \psi) [l_2 \cos(\theta_2) + l_3 \cos(\theta_2 + \theta_3)] \quad (2.19)$$

$$\frac{\partial x_{ee}}{\partial \theta_2} = -\cos(\theta_1 + \psi) [l_2 \sin(\theta_2) + l_3 \sin(\theta_2 + \theta_3)] \quad (2.20)$$

$$\frac{\partial x_{ee}}{\partial \theta_3} = -l_3 \cos(\theta_1 + \psi) \sin(\theta_2 + \theta_3) \quad (2.21)$$

Desarrollando las derivadas de (2.12) se tiene:

$$\frac{\partial y_{ee}}{\partial x} = 0 \quad (2.22)$$

$$\frac{\partial y_{ee}}{\partial y} = 1 \quad (2.23)$$

$$\frac{\partial y_{ee}}{\partial \psi} = b \cos(\psi) + \cos(\theta_1 + \psi) [l_2 \cos(\theta_2) + l_3 \cos(\theta_2 + \theta_3)] \quad (2.24)$$

$$\frac{\partial y_{ee}}{\partial \theta_1} = \cos(\theta_1 + \psi) [l_2 \cos(\theta_2) + l_3 \cos(\theta_2 + \theta_3)] \quad (2.25)$$

$$\frac{\partial y_{ee}}{\partial \theta_2} = -\sin(\theta_1 + \psi) [l_2 \sin(\theta_2) + l_3 \sin(\theta_2 + \theta_3)] \quad (2.26)$$

$$\frac{\partial y_{ee}}{\partial \theta_3} = -l_3 \sin(\theta_1 + \psi) \sin(\theta_2 + \theta_3) \quad (2.27)$$

Desarrollando las derivadas de (2.13) se tiene:

$$\frac{\partial z_{ee}}{\partial x} = 0 \quad (2.28)$$

$$\frac{\partial z_{ee}}{\partial y} = 0 \quad (2.29)$$

$$\frac{\partial z_{ee}}{\partial \psi} = 0 \quad (2.30)$$

$$\frac{\partial z_{ee}}{\partial \theta_1} = 0 \quad (2.31)$$

$$\frac{\partial z_{ee}}{\partial \theta_2} = l_2 \cos(\theta_2) + l_3 \cos(\theta_2 + \theta_3) \quad (2.32)$$

$$\frac{\partial z_{ee}}{\partial \theta_3} = l_3 \cos(\theta_2 + \theta_3) \quad (2.33)$$

Reescribiendo 2.14 se tiene:

$$\begin{bmatrix} \dot{x}_{ee} \\ \dot{y}_{ee} \\ \dot{z}_{ee} \end{bmatrix} = J_{ee} \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\psi} \\ \dot{\theta}_1 \\ \dot{\theta}_2 \\ \dot{\theta}_3 \end{bmatrix} \quad (2.34)$$

Reemplazando (2.15) en (2.34) y multiplicando se encuentra \dot{x}_{ee} :

$$\begin{aligned} \dot{x}_{ee} = & \dot{x} + \{-b \sin(\psi) - \sin(\theta_1 + \psi)[l_2 \cos(\theta_2) + l_3 \cos(\theta_2 + \theta_3)]\} \dot{\psi} + \{-\sin(\theta_1 + \\ & \psi) [l_2 \cos(\theta_2) + l_3 \cos(\theta_2 + \theta_3)]\} \dot{\theta}_1 + \{-\cos(\theta_1 + \psi)[l_2 \sin(\theta_2) + l_3 \sin(\theta_2 + \theta_3)]\} \dot{\theta}_2 + \\ & \{-l_3 \cos(\theta_1 + \psi) \sin(\theta_2 + \theta_3)\} \dot{\theta}_3 \end{aligned} \quad (2.35)$$

Reemplazando (2.2) en (2.35)

$$\dot{x}_{ee} = u \cos(\psi) - a \omega \sin(\psi) + \{-b \sin(\psi) - \sin(\theta_1 + \psi)[l_2 \cos(\theta_2) + l_3 \cos(\theta_2 + \theta_3)]\}\omega + \{-\sin(\theta_1 + \psi)[l_2 \cos(\theta_2) + l_3 \cos(\theta_2 + \theta_3)]\}\dot{\theta}_1 + \{-\cos(\theta_1 + \psi)[l_2 \sin(\theta_2) + l_3 \sin(\theta_2 + \theta_3)]\}\dot{\theta}_2 + \{-l_3 \cos(\theta_1 + \psi) \sin(\theta_2 + \theta_3)\}\dot{\theta}_3 \quad (2.36)$$

$$\dot{x}_{ee} = u \cos(\psi) + \{-a \sin \psi - b \sin(\psi) - \sin(\theta_1 + \psi)[l_2 \cos(\theta_2) + l_3 \cos(\theta_2 + \theta_3)]\}\omega + \{-\sin(\theta_1 + \psi)[l_2 \cos(\theta_2) + l_3 \cos(\theta_2 + \theta_3)]\}\dot{\theta}_1 + \{-\cos(\theta_1 + \psi)[l_2 \sin(\theta_2) + l_3 \sin(\theta_2 + \theta_3)]\}\dot{\theta}_2 + \{-l_3 \cos(\theta_1 + \psi) \sin(\theta_2 + \theta_3)\}\dot{\theta}_3 \quad (2.37)$$

Se realiza de la misma forma para \dot{y}_{ee} , se encuentra de (2.34)

$$\dot{y}_{ee} = \dot{y} + \{b \cos(\psi) + \cos(\theta_1 + \psi)[l_2 \cos(\theta_2) + l_3 \cos(\theta_2 + \theta_3)]\}\dot{\psi} + \{\cos(\theta_1 + \psi)[l_2 \cos(\theta_2) + l_3 \cos(\theta_2 + \theta_3)]\}\dot{\theta}_1 + \{-\sin(\theta_1 + \psi)[l_2 \sin(\theta_2) + l_3 \sin(\theta_2 + \theta_3)]\}\dot{\theta}_2 + \{-l_3 \sin(\theta_1 + \psi) \sin(\theta_2 + \theta_3)\}\dot{\theta}_3 \quad (2.38)$$

Reemplazando (2.2) en (2.38)

$$\dot{y}_{ee} = u \sin(\psi) + a \omega \cos(\psi) + \{b \cos(\psi) + \cos(\theta_1 + \psi)[l_2 \cos(\theta_2) + l_3 \cos(\theta_2 + \theta_3)]\}\omega + \{\cos(\theta_1 + \psi)[l_2 \cos(\theta_2) + l_3 \cos(\theta_2 + \theta_3)]\}\dot{\theta}_1 + \{-\sin(\theta_1 + \psi)[l_2 \sin(\theta_2) + l_3 \sin(\theta_2 + \theta_3)]\}\dot{\theta}_2 + \{-l_3 \sin(\theta_1 + \psi) \sin(\theta_2 + \theta_3)\}\dot{\theta}_3 \quad (2.39)$$

$$\dot{y}_{ee} = u \sin(\psi) + \{a \cos(\psi) + b \cos(\psi) + \cos(\theta_1 + \psi)[l_2 \cos(\theta_2) + l_3 \cos(\theta_2 + \theta_3)]\}\omega + \{\cos(\theta_1 + \psi)[l_2 \cos(\theta_2) + l_3 \cos(\theta_2 + \theta_3)]\}\dot{\theta}_1 + \{-\sin(\theta_1 + \psi)[l_2 \sin(\theta_2) + l_3 \sin(\theta_2 + \theta_3)]\}\dot{\theta}_2 + \{-l_3 \sin(\theta_1 + \psi) \sin(\theta_2 + \theta_3)\}\dot{\theta}_3 \quad (2.40)$$

De igual forma para \dot{z}_{ee} , se encuentra de (2.34)

$$\dot{z}_{ee} = \{l_2 \cos(\theta_2) + l_3 \cos(\theta_2 + \theta_3)\}\dot{\theta}_2 + \{l_3 \cos(\theta_2 + \theta_3)\}\dot{\theta}_3 \quad (2.41)$$

Para simplificar las expresiones obtenidas se utilizan la siguiente nomenclatura:

$$C_\psi = \cos(\psi), S_\psi = \sin(\psi), \quad C_{\theta_2} = \cos(\theta_2), \quad S_{\theta_2} = \sin(\theta_2), \quad S_{\theta_1\psi} = \sin(\theta_1 + \psi), \\ C_{\theta_2\theta_3} = \cos(\theta_2 + \theta_3), C_{\theta_1\psi} = \cos(\theta_1 + \psi), S_{\theta_2\theta_3} = \sin(\theta_2 + \theta_3)$$

Reescribiendo (2.37), (2.40) y (2.41) se tiene:

$$\dot{x}_{ee} = u C_\psi + \{-a S_\psi - b S_\psi - S_{\theta_1\psi} [l_2 C_{\theta_2} + l_3 C_{\theta_2\theta_3}]\} \omega + \{-S_{\theta_1\psi} [l_2 C_{\theta_2} + l_3 C_{\theta_2\theta_3}]\} \dot{\theta}_1 + \{-C_{\theta_1\psi} [l_2 S_{\theta_2} + l_3 S_{\theta_2\theta_3}]\} \dot{\theta}_2 + \{-l_3 C_{\theta_1\psi} S_{\theta_2\theta_3}\} \dot{\theta}_3 \quad (2.42)$$

$$\dot{y}_{ee} = u S_\psi + \{a C_\psi + b C_\psi + C_{\theta_1\psi} [l_2 C_{\theta_2} + l_3 C_{\theta_2\theta_3}]\} \omega + \{C_{\theta_1\psi} [l_2 C_{\theta_2} + l_3 C_{\theta_2\theta_3}]\} \dot{\theta}_1 + \{-S_{\theta_1\psi} [l_2 S_{\theta_2} + l_3 S_{\theta_2\theta_3}]\} \dot{\theta}_2 + \{-l_3 S_{\theta_1\psi} S_{\theta_2\theta_3}\} \dot{\theta}_3 \quad (2.43)$$

$$\dot{z}_{ee} = (l_2 C_{\theta_2} + l_3 C_{\theta_2\theta_3}) \dot{\theta}_2 + (l_3 C_{\theta_2\theta_3}) \dot{\theta}_3 \quad (2.44)$$

La cinemática del sistema queda definida por:

$$\begin{bmatrix} \dot{x}_{ee} \\ \dot{y}_{ee} \\ \dot{z}_{ee} \end{bmatrix} = \begin{bmatrix} C_\psi & -a S_\psi - b S_\psi - S_{\theta_1\psi} [l_2 C_{\theta_2} + l_3 C_{\theta_2\theta_3}] & -S_{\theta_1\psi} [l_2 C_{\theta_2} + l_3 C_{\theta_2\theta_3}] & -C_{\theta_1\psi} [l_2 S_{\theta_2} + l_3 S_{\theta_2\theta_3}] & -l_3 C_{\theta_1\psi} S_{\theta_2\theta_3} \\ S_\psi & a C_\psi + b C_\psi + C_{\theta_1\psi} [l_2 C_{\theta_2} + l_3 C_{\theta_2\theta_3}] & C_{\theta_1\psi} [l_2 C_{\theta_2} + l_3 C_{\theta_2\theta_3}] & -S_{\theta_1\psi} [l_2 S_{\theta_2} + l_3 S_{\theta_2\theta_3}] & -l_3 S_{\theta_1\psi} S_{\theta_2\theta_3} \\ 0 & 0 & 0 & l_2 C_{\theta_2} + l_3 C_{\theta_2\theta_3} & l_3 C_{\theta_2\theta_3} \end{bmatrix} \begin{bmatrix} u \\ \omega \\ \dot{\theta}_1 \\ \dot{\theta}_2 \\ \dot{\theta}_3 \end{bmatrix} \quad (2.45)$$

Por facilidad de demostraciones se reescribe (2.45) como:

$$\begin{bmatrix} \dot{x}_{ee} \\ \dot{y}_{ee} \\ \dot{z}_{ee} \end{bmatrix} = J_{ee} \begin{bmatrix} u \\ \omega \\ \dot{\theta}_1 \\ \dot{\theta}_2 \\ \dot{\theta}_3 \end{bmatrix} \quad (2.46)$$

Dónde:

$$J_{ee} = \begin{bmatrix} C_\psi & -a S_\psi - b S_\psi - S_{\theta_1\psi} [l_2 C_{\theta_2} + l_3 C_{\theta_2\theta_3}] & -S_{\theta_1\psi} [l_2 C_{\theta_2} + l_3 C_{\theta_2\theta_3}] & -C_{\theta_1\psi} [l_2 S_{\theta_2} + l_3 S_{\theta_2\theta_3}] & -l_3 C_{\theta_1\psi} S_{\theta_2\theta_3} \\ S_\psi & a C_\psi + b C_\psi + C_{\theta_1\psi} [l_2 C_{\theta_2} + l_3 C_{\theta_2\theta_3}] & C_{\theta_1\psi} [l_2 C_{\theta_2} + l_3 C_{\theta_2\theta_3}] & -S_{\theta_1\psi} [l_2 S_{\theta_2} + l_3 S_{\theta_2\theta_3}] & -l_3 S_{\theta_1\psi} S_{\theta_2\theta_3} \\ 0 & 0 & 0 & l_2 C_{\theta_2} + l_3 C_{\theta_2\theta_3} & l_3 C_{\theta_2\theta_3} \end{bmatrix} \quad (2.47)$$

2.1.4 CINEMÁTICA INVERSA

Como se explicó en la sección 1.6.1.2, el objetivo de la cinemática inversa es determinar las variables de articulaciones correspondientes a una orientación y posiciones específicas del efector final, estas variables para el caso del manipulador móvil, se pueden obtener de (2.45), de la siguiente manera:

$$\begin{bmatrix} u \\ \omega \\ \dot{\theta}_1 \\ \dot{\theta}_2 \\ \dot{\theta}_3 \end{bmatrix} = J_{ee}^+ \begin{bmatrix} \dot{x}_{ee} \\ \dot{y}_{ee} \\ \dot{z}_{ee} \end{bmatrix} \quad (2.48)$$

Donde J_{ee}^+ es la matriz pseudoinversa, explicada en la sección 1.7.9. Reemplazando la definición de matriz pseudoinversa por derecha, la cinemática inversa del manipulador móvil queda definida por:

$$\begin{bmatrix} u \\ \omega \\ \dot{\theta}_1 \\ \dot{\theta}_2 \\ \dot{\theta}_3 \end{bmatrix} = J_{ee}^T (J_{ee} J_{ee}^T)^{-1} \begin{bmatrix} \dot{x}_{ee} \\ \dot{y}_{ee} \\ \dot{z}_{ee} \end{bmatrix} \quad (2.49)$$

2.2 MODELO DINÁMICO

Como se menciona en la sección 1.6.2, la dinámica estudia las fuerzas que originan el movimiento del robot, a diferencia de la cinemática que estudia el movimiento sin analizar las fuerzas que lo generan.

El objetivo de analizar el modelo dinámico en el presente proyecto es aplicar al modelo dinámico los controladores cinemáticos diseñados, y de esta manera probar la robustez de los controladores, ya que al no ser considerada la parte dinámica en el diseño del controlador, esta se convierte en una perturbación en la entrada del controlador cinemático.

Para el análisis dinámico, en vista de que la masa del manipulador es bastante menor que la masa del robot móvil, en una proporción de aproximadamente 1:7 [40] [10], el modelo dinámico del manipulador móvil analizado puede ser aproximado al modelo dinámico del robot móvil, el cual se presenta a continuación.

Para describir la dinámica del robot móvil tipo unicycle de la Figura 2.8 se considera el modelo dinámico validado experimentalmente con robots PIONEER 3DX en [41].

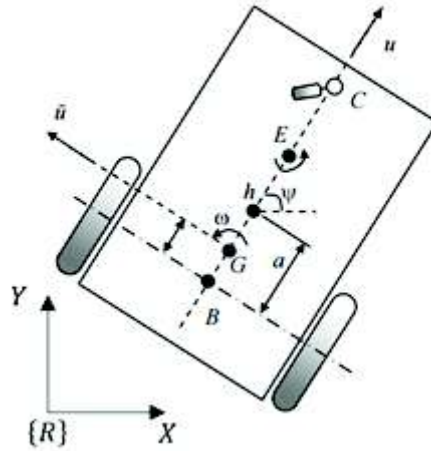


Figura 2.8. Modelo del Robot PIONEER [41]

Donde, u y \bar{u} son las velocidades longitudinal y lateral del centro de masa, C es el punto de ubicación de la rueda castor, E es el punto donde se ubica la herramienta de trabajo, ψ es el ángulo de orientación del robot, $h = [x, y]^T$ es el punto que define la posición del robot, el cual está ubicado a una distancia a delante del centro del eje que une las ruedas traseras del robot, G es el centro de gravedad, ω es la velocidad angular del robot, y B es el centro de la línea de base de las ruedas [28].

A partir del modelo de la Figura 2.8 se obtiene el siguiente modelo dinámico:

$$\begin{bmatrix} \dot{\mu} \\ \dot{\omega} \end{bmatrix} = \begin{bmatrix} \left(\frac{B_3}{B_1}\right)\omega^2 - \left(\frac{B_4}{B_1}\right)\mu \\ -\left(\frac{B_5}{B_2}\right)\mu\omega - \left(\frac{B_6}{B_2}\right)\omega \end{bmatrix} + \begin{bmatrix} \frac{1}{B_1} & 0 \\ 0 & \frac{1}{B_2} \end{bmatrix} \begin{bmatrix} \mu_c \\ \omega_c \end{bmatrix} \quad (2.50)$$

Donde, $[B] = [B_1 \ B_2 \ B_3 \ B_4 \ B_5 \ B_6]^T$ es el vector de parámetros identificados del robot.

De esta manera, tomando las consideraciones antes mencionadas, el modelo dinámico del manipulador móvil queda definido por:

$$\begin{bmatrix} \dot{x}_{ee} \\ \dot{y}_{ee} \\ \dot{z}_{ee} \\ \dot{\psi} \\ \dot{\mu} \\ \dot{\omega} \end{bmatrix} = \begin{bmatrix} \mu J_{11} + \omega J_{12} + \theta_{1c} J_{13} + \theta_{2c} J_{14} + \theta_{3c} J_{15}; \\ \mu J_{21} + \omega J_{22} + \theta_{1c} J_{23} + \theta_{2c} J_{24} + \theta_{3c} J_{25}; \\ \mu J_{31} + \omega J_{32} + \theta_{1c} J_{33} + \theta_{2c} J_{34} + \theta_{3c} J_{35}; \\ \omega; \\ \left(\frac{B_3}{B_1}\right) \omega^2 - \left(\frac{B_4}{B_1}\right) \mu + \left(\frac{1}{B_1}\right) \mu_C; \\ -\left(\frac{B_5}{B_2}\right) \mu \omega - \left(\frac{B_6}{B_2}\right) \omega + \left(\frac{1}{B_2}\right) \omega_C; \end{bmatrix} \quad (2.51)$$

En la tabla 2.1 se presentan los valores de los parámetros dinámicos identificados para robots Pioneer 3-DX en [41].

Tabla 2.1. Parámetros dinámicos identificados para el robot Pioneer 3-DX

Parámetro	Valor
B_1	0.2604
B_2	0.2509
B_3	-0.000499
B_4	0.9965
B_5	0.00263
B_6	1.0768

2.3 SIMULACIÓN DEL MODELO

El presente proyecto, abarca únicamente la simulación por lo que no se realiza la comparación con los robots físicos para la validación. Por esta razón en el presente proyecto se realiza una simulación de las ecuaciones que describen el comportamiento del manipulador móvil mediante la simulación en Matlab.

Para tener la certeza de que el modelo matemático es correcto, se procede a verificar el mismo mediante un conjunto de pruebas en lazo abierto. Estas pruebas consisten en variar los parámetros de entrada del sistema y observar si las salidas se comportan de manera esperada de acuerdo a las ecuaciones matemáticas del modelo obtenido.

2.3.1 SIMULACIÓN DEL MODELO CINEMÁTICO

Para verificar el modelo cinemático, las pruebas consisten en dar diferentes valores de velocidad en las entradas del sistema y observar si los valores en la salida se

comportan de acuerdo a las relaciones geométricas descritas en (2.45). Las pruebas de simulación se realizan en lazo abierto, como se muestra en la Figura 2.9.

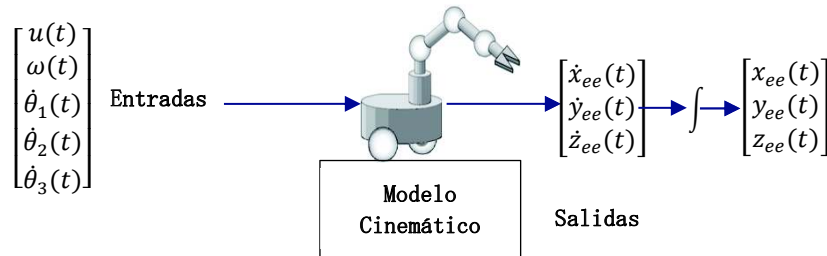


Figura 2.9. Esquema de pruebas de simulación del modelo cinemático con velocidades

En primer lugar, se da una velocidad lineal constante $\mu = 1m/s$ para un tiempo de simulación de 10 segundos, mientras las demás entradas del sistema tienen un valor de cero, es decir $[\mu \ \omega \ \dot{\theta}_1 \ \dot{\theta}_2 \ \dot{\theta}_3]^T = [1 \ 0 \ 0 \ 0 \ 0]^T$

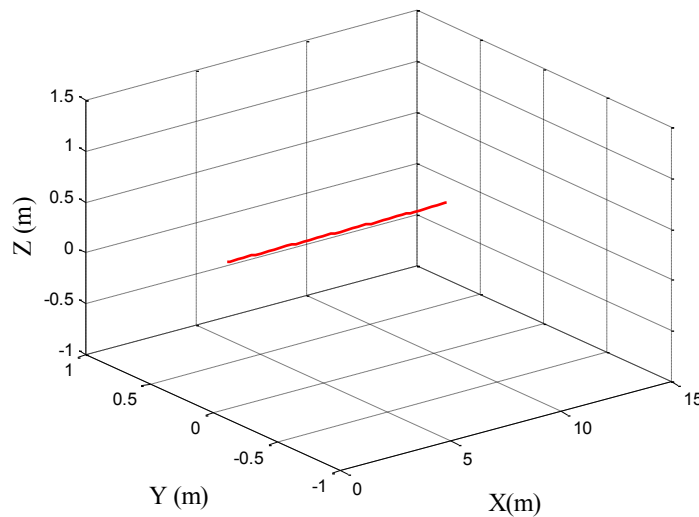


Figura 2.10. Simulación del modelo cinemático, posición del efector final $u = 1m/s$

En la Figura 2.10 se presenta el recorrido realizado por el efector final del manipulador móvil al dar una velocidad lineal de $u = 1m/s$ como entrada del sistema durante un tiempo de diez segundos. Por la definición de velocidad se espera que el robot recorra una distancia de diez metros, la posición en x del efector final observada es $x_{ee} = 10.61m$, que corresponde a la distancia recorrida por la base móvil sumado a la longitud de los eslabones del brazo manipulador y la distancia desde eje de las ruedas hasta la base del manipulador, es decir $x_{ee} = 10m + b + l_2 + l_3$.

Las demás posiciones del efector final son $y_{ee} = 0m$ y $z_{ee} = 0.443m$, es decir que como se esperaba, solo la posición en el eje x fue afectada, por lo que el resultado de esta primera prueba es positivo.

Se realiza una segunda prueba, similar a la primera, ingresando una velocidad angular constante $\omega = 1rad/s$ durante diez segundos mientras las otras entradas se mantienen en cero, es decir $[\mu \ \omega \ \dot{\theta}_1 \ \dot{\theta}_2 \ \dot{\theta}_3]^T = [0 \ 1 \ 0 \ 0 \ 0]^T$.

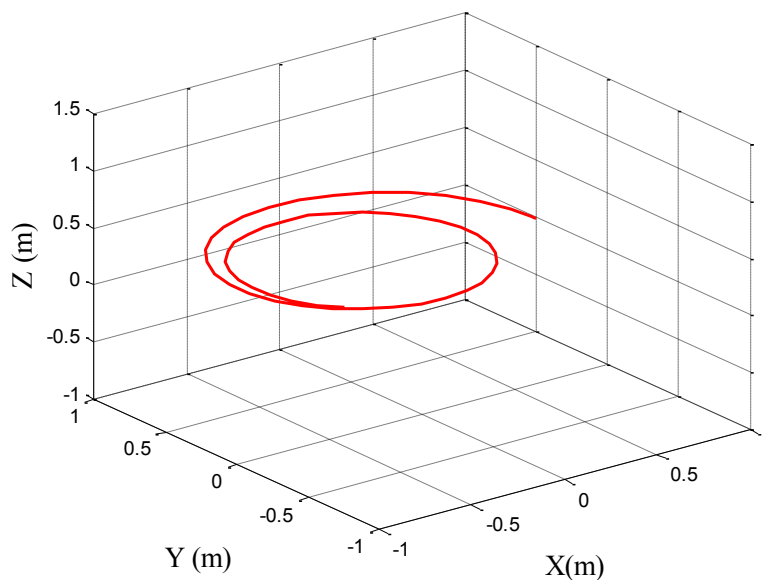


Figura 2.11. Simulación del modelo cinemático, posición del efector final $\omega = 1rad/s$

La respuesta del sistema se ilustra en la Figura 2.11, donde se observa la una forma circular, la posición del efector final en el eje z no varía, por lo que el resultado es satisfactorio para la segunda prueba.

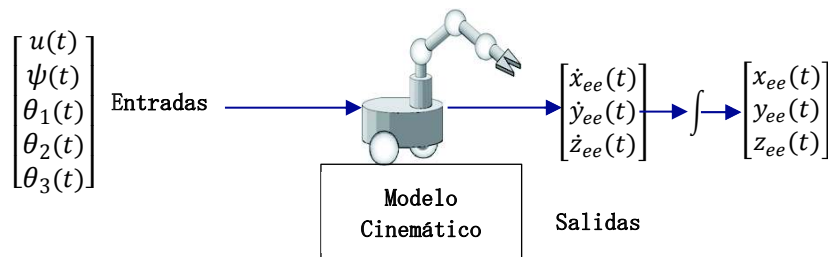
Después la prueba de simulación se repite, variando las demás velocidades de entrada del sistema.

En la tabla 2.2 se muestran los resultados de las pruebas realizadas para diferentes velocidades de entrada.

Tabla 2.2. Pruebas de simulación con velocidades de entrada, modelo cinemático

Velocidades de entrada					Posición del efector final			Tiempo
$\mu \left(\frac{m}{s}\right)$	$\omega \left(\frac{rad}{s}\right)$	$\dot{\theta}_1 \left(\frac{rad}{s}\right)$	$\dot{\theta}_2 \left(\frac{rad}{s}\right)$	$\dot{\theta}_3 \left(\frac{rad}{s}\right)$	$x_{ee}(m)$	$y_{ee}(m)$	$z_{ee}(m)$	(s)
1	0	0	0	0	10.61	0	0.44	10
0	1	0	0	0	-0.69	-0.35	0.44	10
0	0	1	0	0	0.21	-1.35	0.44	10
0	0	0	1	0	1.02	0	0.56	10
0	0	0	0	1	-1.38	0	0.79	10

Como se observa en los resultados de la Tabla 2.2, los valores de posición del efector final en el eje Z, varían cuando se dan velocidades a las articulaciones θ_2 y θ_3 , lo cual es un resultado esperado, mientras que al dar velocidad a la articulación θ_1 , se produce variación de posición del efector final en X e Y.

**Figura 2.12.** Esquema de pruebas de simulación del modelo cinemático con posiciones

Se procede a realizar una segunda prueba, dando valores a las condiciones iniciales del ángulo de rotación de la plataforma móvil, así como de los ángulos de las articulaciones θ_1, θ_2 y θ_3 .

Al mismo tiempo se da una velocidad lineal constante a la plataforma móvil, con la finalidad de observar la variación en la posición del efector final, como se muestra en el esquema de la Figura 2.12.

En la Figura 2.13 se puede observar la comparación del recorrido realizado por el efector final para dos ángulos de orientación ψ de la plataforma móvil.

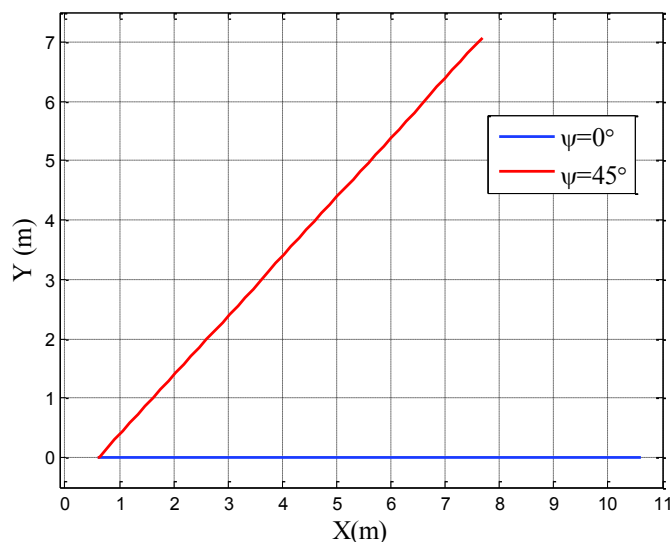


Figura 2.13. Simulación del modelo cinemático, variación del ángulo ψ

Los resultados para diferentes condiciones iniciales de ángulos se muestran en la tabla 2.3. En estos resultados se puede observar que el comportamiento de la posición del efector final coincide con el esperado de acuerdo a las expresiones matemáticas del modelo. Al comprobar que los resultados de la simulación del modelo cinemático coinciden con los valores esperados al resolver las ecuaciones que describen al manipulador móvil, y que el comportamiento de la salida del sistema, para diferentes entradas es el esperado, con lo cual se verifica que el modelo cinemático obtenido es correcto.

Tabla 2.3. Pruebas de simulación con condiciones iniciales, modelo cinemático

Condiciones iniciales					Posición del efector final			Tiempo
$\mu \left(\frac{m}{s}\right)$	$\psi(^{\circ})$	$\theta_1(^{\circ})$	$\theta_2(^{\circ})$	$\theta_3(^{\circ})$	$x_{ee}(m)$	$y_{ee}(m)$	$z_{ee}(m)$	(s)
1	0	0	0	0	10.61	0	0.44	10
1	30	0	0	0	9.28	5	0.44	10
1	0	45	0	0	10.40	1.86	0.44	10
1	0	0	60	0	10.61	0	0.44	10
1	0	0	0	90	10.61	0	0.44	10

2.3.2 SIMULACION DEL MODELO DINÁMICO

Como se mencionó en la sección 2.2, el modelo dinámico del robot móvil Pioneer 3-DX presentado en el presente proyecto ha sido validado en [41], mediante una parametrización lineal.

Se procede a realizar las pruebas de simulación de manera similar a la que se efectuó con el modelo cinemático, dando diferentes valores en las entradas del sistema y observando las salidas del mismo, como se muestra en la Figura 2.14.

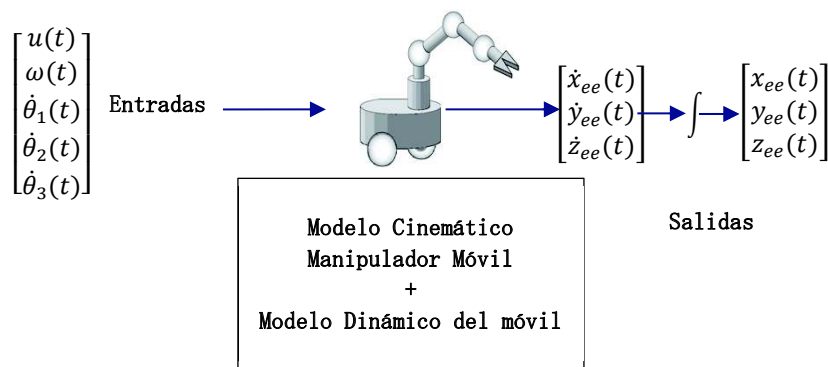


Figura 2.14. Esquema de pruebas de simulación del modelo dinámico con velocidades

A continuación se presentan resultados de simulación de la simulación de este modelo.

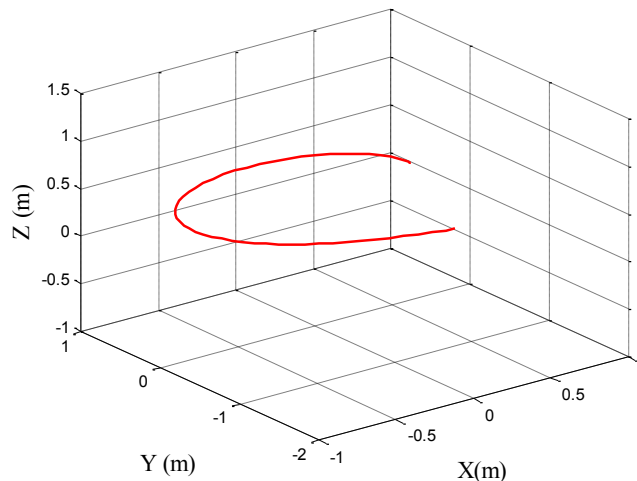


Figura 2.15. Simulación del modelo dinámico, posición del efector final $\dot{\theta}_1 = 1 \text{ rad/s}$

En la Figura 2.15 se presenta la trayectoria que describe el efector final cuando se aplica una velocidad angular $\dot{\theta}_1 = 1 \text{ rad/s}$ al primer eslabón del brazo manipulador. Se puede observar que la trayectoria descrita es la esperada.

En la tabla 2.4 se presentan los resultados de las pruebas al dar valores a las diferentes variables de entrada del sistema.

Tabla 2.4. Pruebas de simulación con velocidades de entrada, modelo dinámico

Velocidades de entrada					Posición del efector final			Tiempo
$\mu \left(\frac{m}{s}\right)$	$\omega \left(\frac{rad}{s}\right)$	$\dot{\theta}_1 \left(\frac{rad}{s}\right)$	$\dot{\theta}_2 \left(\frac{rad}{s}\right)$	$\dot{\theta}_3 \left(\frac{rad}{s}\right)$	$x_{ee}(m)$	$y_{ee}(m)$	$z_{ee}(m)$	(s)
1	0	0	0	0	10.41	0	0.44	10
0	1	0	0	0	-0.85	0.21	0.44	10
0	0	1	0	0	0.21	-1.35	0.44	10
0	0	0	1	0	1.02	0	0.56	10
0	0	0	0	1	-1.38	0	0.79	10

Comparando estos resultados con los obtenidos en la tabla 2.2, se puede observar que comportamiento de las salidas del sistema es similar al del modelo cinemático, con variaciones en los valores en los ejes X e Y , mientras que los valores en Z permanecen iguales.

Este comportamiento es esperado, puesto que el modelo dinámico presentado corresponde al del robot móvil, el cual solo puede desplazarse en el plano XY .

También se observa que la distancia recorrida en el eje X cuando se aplica una velocidad lineal $u = 1 \text{ m/s}$, es menor que la distancia recorrida para el modelo cinemático, lo que también es esperado debido a que ahora se considera la masa del sistema.

Se procede a realizar una segunda prueba, dando condiciones a los ángulos de las articulaciones y al ángulo de rotación de la plataforma móvil, como se muestra en la figura 2.16.

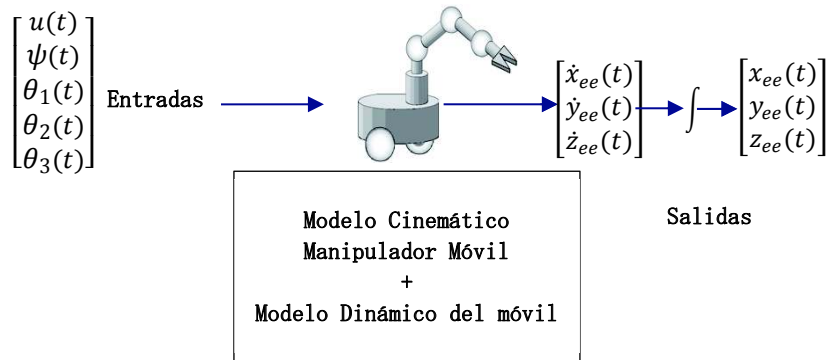


Figura 2.16. Esquema de pruebas de simulación del modelo dinámico con velocidades

En la figura 2.17 se observa el comportamiento del efector final cuando se varía el valor del ángulo ψ de orientación de la plataforma móvil.

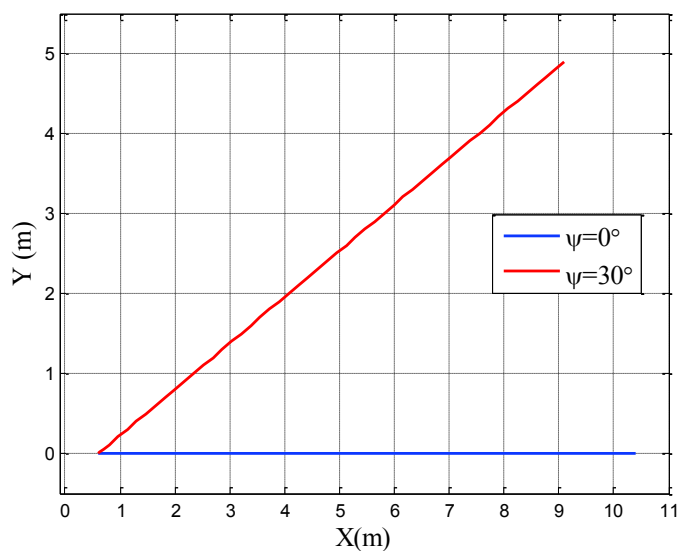


Figura 2.17. Simulación del modelo cinemático, variación del ángulo ψ

Como se observa, el comportamiento del sistema al variar el ángulo de orientación de la plataforma, es similar al observado en el modelo cinemático.

En la tabla 2.5 se presentan los resultados para diferentes condiciones iniciales.

Los resultados observados muestran un comportamiento esperado en las salidas del modelo manipulador móvil obtenido.

De esta manera se concluye la simulación del modelo matemático.

Tabla 2.5. Pruebas de simulación con condiciones iniciales, modelo dinámico

Condiciones iniciales					Posición del efector final			Tiempo
$\mu \left(\frac{m}{s}\right)$	$\psi(^{\circ})$	$\theta_1(^{\circ})$	$\theta_2(^{\circ})$	$\theta_3(^{\circ})$	$x_{ee}(m)$	$y_{ee}(m)$	$z_{ee}(m)$	(s)
1	0	0	0	0	10.41	0	0.44	10
1	30	0	0	0	9.10	4.90	0.44	10
1	0	45	0	0	10.21	1.80	0.44	10
1	0	0	60	0	10.41	0	0.44	10
1	0	0	0	90	10.41	0	0.44	10

Todas las pruebas realizadas para la simulación de los modelos cinemático y dinámico del manipulador móvil del presente proyecto, se realizaron a bajas velocidades para que el comportamiento del manipulador móvil se lo pueda analizar con relaciones geométricas.

CAPÍTULO 3

DISEÑO E IMPLEMENTACIÓN DE LOS ALGORITMOS DE CONTROL

En este capítulo se presenta el diseño de cada uno de los controladores para el seguimiento de trayectorias del manipulador móvil. Los sistemas de control propuestos utilizan tres técnicas de diseño explicadas en el Capítulo 1, control en tiempo continuo basado en mínima norma, control en tiempo discreto basado en aproximaciones de Euler y control en tiempo continuo basado en modos deslizantes. Los tres controladores presentados se realizan en base al modelo cinemático del manipulador móvil, explicado en la sección 2.1.3.

3.1 DISEÑO DE LOS CONTROLADORES

3.1.1 CONTROLADOR BASADO EN MÍNIMA NORMA

Este controlador se basa en el concepto de norma mínima explicado en la sección 1.8.2.1, el objetivo de control es que el manipulador móvil realice el seguimiento de trayectorias realizando el menor número posible de movimientos.

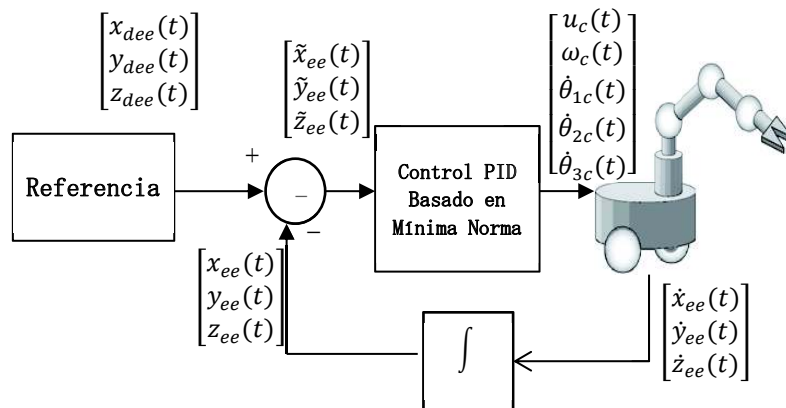


Figura 3.1. Diagrama de bloques del controlador PID

En la Figura 3.1 se presenta el diagrama de bloques del controlador PID diseñado, el mismo que se aplica al modelo del manipulador móvil.

Reescribiendo la ecuación (2.46) de la sección 2.1.3 obtenemos una expresión resumida del modelo cinemático del sistema:

$$\dot{h}(t) = J_{ee} U(t) \quad (3.1)$$

Donde $U(t) = [u(t) \ \omega(t) \ \dot{\theta}_1(t) \ \dot{\theta}_2(t) \ \dot{\theta}_3(t)]^T$ es el vector de velocidades de entrada del sistema, y $\dot{h}(t) = [\dot{x}_{ee}(t) \ \dot{y}_{ee}(t) \ \dot{z}_{ee}(t)]^T$ es el vector de velocidades de salida, a continuación de (3.1) se despeja $U(t)$.

$$U(t) = J_{ee}^+ \dot{h}(t) \quad (3.2)$$

Donde J_{ee}^+ es la matriz pseudoinversa del Jacobiano del manipulador móvil.

La ley de control propuesta es:

$$U_c(t) = J_{ee}^+ \left(\dot{h}_d(t) + k \tilde{h}(t) \right) \quad (3.3)$$

Donde, $\dot{h}_d(t) = [\dot{x}_{dee}(t) \ \dot{y}_{dee}(t) \ \dot{z}_{dee}(t)]^T$ es el vector de velocidades deseadas del efector final, $\tilde{h}(t)$ es el error de posición del efector final, $k = [k_x \ 0 \ 0; 0 \ k_y \ 0; 0 \ 0 \ k_z]^T$ es vector de constantes de ajuste de controlador, $\tilde{h}(t)$ es el error de posición del efector final, el cual se define como la diferencia entre el valor deseado de velocidad y el valor obtenido del robot, es decir:

$$\tilde{h}(t) = h_d(t) - h(t) = \begin{bmatrix} x_{dee}(t) - x_{ee}(t) \\ y_{dee}(t) - y_{ee}(t) \\ z_{dee}(t) - z_{ee}(t) \end{bmatrix} \quad (3.4)$$

Donde, $h(t) = [x_{ee}(t) \ y_{ee}(t) \ z_{ee}(t)]^T$ es el vector de posiciones del efector final del manipulador móvil y $h_d(t) = [x_{dee}(t) \ y_{dee}(t) \ z_{dee}(t)]^T$ es el vector de posiciones deseadas.

Se asume que el controlador provocará en el sistema que el error de velocidad se mantenga en cero, lo que se conoce como seguimiento perfecto de velocidad, esta condición se expresa de la siguiente manera:

$$U(t) \cong U_c(t) \quad (3.5)$$

Donde, $U_c(t) = [u_c(t) \omega_c(t) \dot{\theta}_{1c}(t) \dot{\theta}_{2c}(t) \dot{\theta}_{3c}(t)]^T$ son las señales de control que ingresan al manipulador móvil.

Reemplazando (3.3) en (3.1):

$$\dot{h}(t) = J_{ee} \left[J_{ee}^+ \left(\dot{h}_d(t) + k\tilde{h}(t) \right) \right] \quad (3.6)$$

Se utiliza la definición de matriz pseudoinversa por la derecha, mostrada en la sección 1.7.9:

$$\dot{h}(t) = J_{ee} J_{ee}^T (J_{ee} J_{ee}^T)^{-1} \left(\dot{h}_d(t) + k\tilde{h}(t) \right) \quad (3.7)$$

Aplicando la definición de matriz identidad de la sección 1.7.4:

$$I = J_{ee} J_{ee}^T (J_{ee} J_{ee}^T)^{-1} \quad (3.8)$$

Reemplazando (3.8) en (3.7):

$$\dot{h}(t) = I \left(\dot{h}_d(t) + k\tilde{h}(t) \right) \quad (3.9)$$

Por las propiedades de la matriz identidad, (3.9) se reescribe como:

$$\dot{h}(t) = \dot{h}_d(t) + k\tilde{h}(t) \quad (3.10)$$

Igualando (3.10) a cero:

$$\dot{h}_d(t) - \dot{h}(t) + k\tilde{h}(t) = 0 \quad (3.11)$$

Reemplazando (3.4) en (3.11):

$$\dot{h}_d(t) - \dot{h}(t) + k(h_d(t) - h(t)) = 0 \quad (3.12)$$

Agrupando términos en (3.12):

$$(\dot{h}_d(t) - \dot{h}(t)) + k(h_d(t) - h(t)) = 0 \quad (3.13)$$

Por la definición de error de (3.4), se puede reescribir (3.13) como:

$$\dot{\tilde{h}}(t) + k\tilde{h}(t) = 0 \quad (3.14)$$

3.1.1.1 Análisis de Estabilidad

De (3.14), se tiene:

$$\dot{\tilde{h}}(t) = -k\tilde{h}(t) \quad (3.15)$$

Para el análisis de estabilidad, se utiliza parte de la teoría de estabilidad de Lyapunov, explicada en la sección 1.10.1 y se propone la siguiente función V candidata de Lyapunov:

$$V = \frac{1}{2}\tilde{h}(t)^T\tilde{h}(t) \quad (3.16)$$

Para cumplir con las condiciones explicadas en la sección 1.10.1.5 se obtiene la derivada de la función candidata.

$$\dot{V} = \tilde{h}(t)^T\dot{\tilde{h}}(t) \quad (3.17)$$

Reemplazando (3.15) en (3.17) se tiene:

$$\begin{aligned} \dot{V} &= \tilde{h}^T(t) \left(-k\tilde{h}(t) \right) \\ \dot{V} &= -\tilde{h}^T(t) k\tilde{h}(t) \end{aligned} \quad (3.18)$$

Para que (3.18) se cumpla:

$$|k\tilde{h}^T(t)\tilde{h}(t)| > 0 \quad (3.19)$$

Para que (3.19) se cumpla:

$$k > 0 \quad (3.20)$$

Del análisis de estabilidad realizado, se comprueba que la derivada en el tiempo de la función de Lyapunov $\dot{V} < 0$, por lo que se verifica que el error de posición $\tilde{h}(t)$ tiende a cero $\tilde{h}(t) \rightarrow 0$ asintóticamente.

3.1.2 CONTROLADOR BASADO EN APROXIMACIONES DE EULER

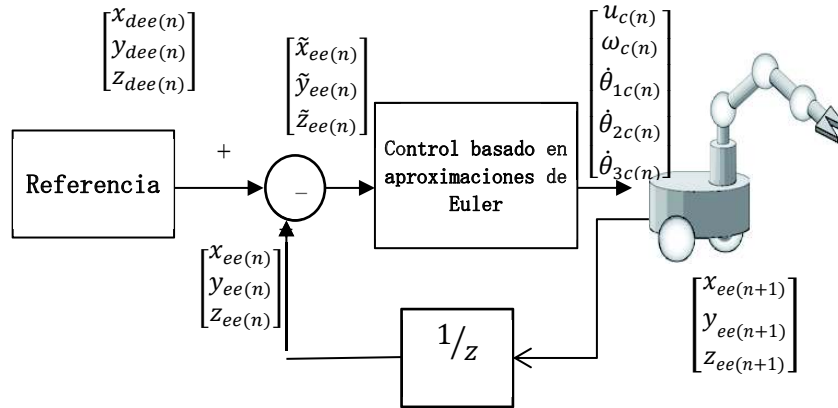


Figura 3.2. Diagrama de bloques del controlador Euler

En la Figura 3.2 se muestra un diagrama de bloques del controlador basado en aproximaciones de Euler.

En este método de diseño, el primer paso es obtener el modelo del sistema en tiempo discreto, aplicando aproximaciones de Euler, explicadas en la sección 1.7.14. Para esto se toma el modelo del sistema descrito en (2.45), el cual queda expresado como:

$$\frac{1}{T_0} \begin{bmatrix} x_{ee(n+1)} - x_{ee(n)} \\ y_{ee(n+1)} - y_{ee(n)} \\ z_{ee(n+1)} - z_{ee(n)} \end{bmatrix} = J_{ee(n)} \begin{bmatrix} u(n) \\ \omega(n) \\ U_1(n) \\ U_2(n) \\ U_3(n) \end{bmatrix} \quad (3.21)$$

Operando en (3.21):

$$\begin{bmatrix} x_{ee(n+1)} \\ y_{ee(n+1)} \\ z_{ee(n+1)} \end{bmatrix} = \begin{bmatrix} x_{ee(n)} \\ y_{ee(n)} \\ z_{ee(n)} \end{bmatrix} + T_0 J_{ee} \begin{bmatrix} u(n) \\ \omega(n) \\ U_1(n) \\ U_2(n) \\ U_3(n) \end{bmatrix} \quad (3.22)$$

En lazo cerrado, asumiendo seguimiento perfecto de velocidades $U \cong U_c$:

$$\begin{bmatrix} u(n) \\ \omega(n) \\ U_{1(n)} \\ U_{2(n)} \\ U_{3(n)} \end{bmatrix} = \begin{bmatrix} u_c(n) \\ \omega_c(n) \\ U_{1c(n)} \\ U_{2c(n)} \\ U_{3c(n)} \end{bmatrix} \quad (3.23)$$

La ley de control propuesta está definida por:

$$\begin{bmatrix} u_c(n) \\ \omega_c(n) \\ U_{1c(n)} \\ U_{2c(n)} \\ U_{3c(n)} \end{bmatrix} = \frac{J_{ee}^T (J_{ee} J_{ee}^T)^{-1}}{T_0} \begin{bmatrix} x_{dee(n+1)} - k_x (x_{dee(n)} - x_{ee(n)}) - x_{ee(n)} \\ y_{dee(n+1)} - k_y (y_{dee(n)} - y_{ee(n)}) - y_{ee(n)} \\ z_{dee(n+1)} - k_z (z_{dee(n)} - z_{ee(n)}) - z_{ee(n)} \end{bmatrix} \quad (3.24)$$

Reemplazando (3.24) en (3.22)

$$\begin{bmatrix} x_{ee(n+1)} \\ y_{ee(n+1)} \\ z_{ee(n+1)} \end{bmatrix} = \begin{bmatrix} x_{ee(n)} \\ y_{ee(n)} \\ z_{ee(n)} \end{bmatrix} + T_0 J_{ee} \frac{J_{ee}^T (J_{ee} J_{ee}^T)^{-1}}{T_0} \begin{bmatrix} x_{dee(n+1)} - k_x (x_{dee(n)} - x_{ee(n)}) - x_{ee(n)} \\ y_{dee(n+1)} - k_y (y_{dee(n)} - y_{ee(n)}) - y_{ee(n)} \\ z_{dee(n+1)} - k_z (z_{dee(n)} - z_{ee(n)}) - z_{ee(n)} \end{bmatrix} \quad (3.25)$$

Simplificando y aplicando propiedades de la matriz identidad de la sección 1.7.4:

$$\begin{bmatrix} x_{ee(n+1)} \\ y_{ee(n+1)} \\ z_{ee(n+1)} \end{bmatrix} = \begin{bmatrix} x_{ee(n)} \\ y_{ee(n)} \\ z_{ee(n)} \end{bmatrix} + \begin{bmatrix} x_{dee(n+1)} - k_x (x_{dee(n)} - x_{ee(n)}) - x_{ee(n)} \\ y_{dee(n+1)} - k_y (y_{dee(n)} - y_{ee(n)}) - y_{ee(n)} \\ z_{dee(n+1)} - k_z (z_{dee(n)} - z_{ee(n)}) - z_{ee(n)} \end{bmatrix} \quad (3.26)$$

Sumando términos semejantes en (3.26), la ley de control queda definida por:

$$\begin{bmatrix} x_{ee(n+1)} \\ y_{ee(n+1)} \\ z_{ee(n+1)} \end{bmatrix} = \begin{bmatrix} x_{dee(n+1)} - k_x (x_{dee(n)} - x_{ee(n)}) \\ y_{dee(n+1)} - k_y (y_{dee(n)} - y_{ee(n)}) \\ z_{dee(n+1)} - k_z (z_{dee(n)} - z_{ee(n)}) \end{bmatrix} \quad (3.27)$$

Para demostrar la estabilidad del controlador, se demuestra que tiene la forma de una función exponencial decreciente en tiempo discreto.

De (3.27)

$$x_{dee(n+1)} - x_{ee(n+1)} = k_x [x_{dee(n)} - x_{ee(n)}]$$

$$e_{1(n+1)} = k_x e_{1(n)}$$

$$y_{dee(n+1)} - y_{ee(n+1)} = k_y [y_{dee(n)} - y_{ee(n)}]$$

$$e_{2(n+1)} = k_y e_{2(n)}$$

$$z_{dee(n+1)} - z_{ee(n+1)} = k_z [z_{ee(n)} - z_{ee(n)}]$$

$$e_{3(n+1)} = k_z e_{3(n)}$$

Por lo tanto los errores quedan definidos por:

$$\begin{bmatrix} e_{1(n+1)} \\ e_{2(n+1)} \\ e_{3(n+1)} \end{bmatrix} = \begin{bmatrix} k_x e_{1(n)} \\ k_y e_{2(n)} \\ k_z e_{3(n)} \end{bmatrix}$$

Desarrollando los errores en el siguiente estado $e_{jee(n+1)}$ se tiene:

$$e_{jee(n+1)} = k_j e_{jee(n)} ;$$

$$e_{jee(n+2)} = k_j e_{jee(n+1)} = k_j^2 e_{jee(n)}$$

$$e_{jee(n+3)} = k_j e_{jee(n+2)} = k_j^3 e_{jee(n)}$$

$$e_{jee(n+4)} = k_j e_{jee(n+3)} = k_j^4 e_{jee(n)}$$

$$e_{jee(n+l)} = k_j e_{jee(n+l-1)} = k_j^l e_{jee(n)}$$

Se verifica que el error tiende asintóticamente a cero, cuando el valor de la constante k está en el rango: $0 < k \leq 1$ y l tiende a infinito $l \rightarrow \infty$.

3.1.3 CONTROLADOR BASADO EN MODOS DESLIZANTES

La teoría de esta técnica de control, junto a la de las otras técnicas utilizadas fue explicada en la sección 1.8.2.

Para los controladores basados en esta técnica, se proponen dos tipos de superficie de deslizamiento: i) una superficie de tipo proporcional, ii) una de tipo proporcional integral. En la Figura 3.3 se presenta el diagrama de bloques de los controladores diseñados mediante la técnica de modos deslizantes.

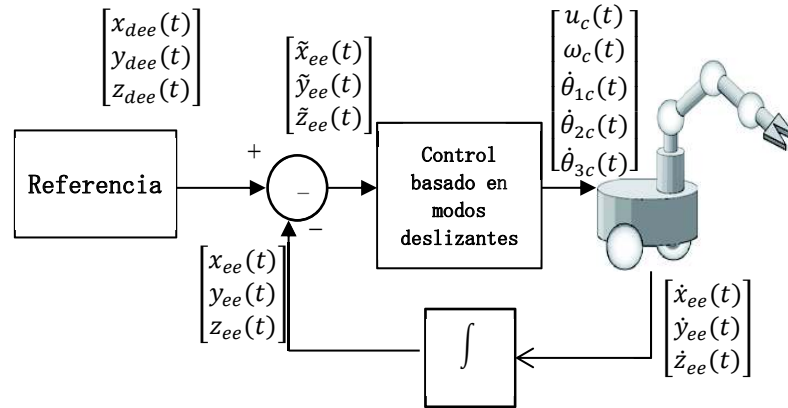


Figura 3.3. Diagrama de bloques del controlador por modos deslizantes

3.1.3.1 Controlador CMD con una superficie proporcional.

Para empezar con el diseño, se define la superficie deslizante de la forma propuesta en (1.5):

$$s(t) = \left(\frac{d}{dt} + \lambda \right)^{n-1} \tilde{h}(t)$$

El modelo cinemático presentado en la sección 2.1.3, es un sistema de orden $n = 1$, por lo que la superficie queda definida por:

$$s(t) = \tilde{h}(t) \quad (3.28)$$

Se deriva (3.28) respecto al tiempo:

$$\begin{aligned} \dot{s}(t) &= \frac{d\tilde{h}(t)}{dt} \\ \dot{s}(t) &= \dot{\tilde{h}}(t) \end{aligned} \quad (3.29)$$

El error $\tilde{h}(t)$, definido en (3.4), se deriva respecto al tiempo:

$$\dot{\tilde{h}}(t) = \dot{h}_a(t) - \dot{h}(t) \quad (3.30)$$

Se reemplaza (3.30) en (3.29):

$$\dot{s}(t) = \dot{h}_a(t) - \dot{h}(t) \quad (3.31)$$

Reemplazando el modelo del robot definido en (3.1), en (3.31), se tiene:

$$\dot{s}(t) = \dot{h}_d(t) - J_{ee} U(t) \quad (3.32)$$

La ley de control $U(t)$ está definida por $U(t) = v_c(t) + v_D(t)$, para encontrar la parte continua de la misma $v_c(t)$, que considera que se alcanzó la superficie deslizante y se desea que la dinámica del sistema se mantenga sobre esta, para lo cual es cero tanto la derivada de la superficie \dot{s} , como la parte discontinua $v_D(t)$.

$$\dot{s}(t) = 0 \quad (3.33)$$

Reemplazando (3.33) en (3.32).

$$0 = \dot{h}_d(t) - J_{ee} U(t) \quad (3.34)$$

Desarrollando (3.34) y considerando $U(t) = v_c(t)$ se tiene:

$$v_c(t) = J_{ee}^T (J_{ee} J_{ee}^T)^{-1} (\dot{h}_d(t)) \quad (3.35)$$

Con cual queda definido la parte continua del controlador.

Para la parte discontinua se considera la función de Lyapunov:

$$V = \frac{1}{2} s^T s \quad (3.36)$$

Derivando (3.36):

$$\dot{V} = s^T \dot{s} \quad (3.37)$$

Reemplazando (3.37) en (3.32):

$$\dot{V} = s^T [\dot{h}_d(t) - J_{ee} U(t)] \quad (3.38)$$

Sustituyendo en (3.38) la ley de control definida por $U(t) = v_c(t) + v_D(t)$:

$$\dot{V} = s^T [\dot{h}_d(t) - J_{ee} \{v_c(t) + v_D(t)\}] \quad (3.39)$$

Reemplazando (3.35) en (3.39):

$$\dot{V} = -s^T J_{ee} v_D(t) \quad (3.40)$$

Para cumplir la condición de estabilidad $\dot{V} < 0$, se propone la siguiente parte discontinua:

$$v_D(t) = J_{ee}^T (J_{ee} J_{ee}^T)^{-1} [k_1 \text{sign}(s(t))] \quad (3.41)$$

Donde k_1 , es una constante positiva de ajuste. La superficie de deslizamiento es alcanzada si $k_1 > 0$, esto garantiza la estabilidad del sistema, es decir $\tilde{h}(t) \rightarrow 0$.

Para reducir el problema de chattering, producido por las no linealidades alrededor del valor $s(t) = 0$ se usa la función sigmoide y se incluye un parámetro δ de sintonización.

$$\text{sign}[s(t)] = \frac{s(t)}{|s(t)| + \delta} \quad (3.42)$$

Se reemplaza (3.42) por la función signo de (3.41) con lo que la parte discontinua se reescribe:

$$v_D(t) = J_e^T (J_e J_e^T)^{-1} \left[k_1 \frac{s(t)}{|s(t)| + \delta} \right] \quad (3.43)$$

Donde $\delta > 0$, entonces la ley de control en modo deslizante queda definida de la siguiente manera:

$$U_c(t) = J_{ee}^T (J_{ee} J_{ee}^T)^{-1} \left[\dot{h}_d(t) + k_1 \frac{s(t)}{|s(t)| + \delta} \right] \quad (3.44)$$

3.1.3.2 Controlador CMD con una superficie proporcional integral

Para este controlador se considera la superficie definida por:

$$s(t) = \left(\frac{d}{dt} + \lambda \right)^n \int \tilde{h}(t) dt$$

Para el sistema de orden $n = 1$, la superficie queda definida por:

$$s(t) = \tilde{h}(t) + \lambda \int \tilde{h}(t) dt \quad (3.45)$$

Derivando (3.45) respecto al tiempo

$$\dot{s}(t) = \dot{\tilde{h}}(t) + \lambda \tilde{h}(t) \quad (3.46)$$

Se reemplaza (3.30) en (3.46):

$$\dot{s}(t) = (\dot{h}_d(t) - \dot{h}(t)) + \lambda \tilde{h}(t) \quad (3.47)$$

Reemplazando en la ecuación anterior, el modelo del robot definido en (3.1), se tiene:

$$\dot{s}(t) = (\dot{h}_d(t) - J_{ee} U(t)) + \lambda \tilde{h}(t) \quad (3.48)$$

La ley de control está definida por $U(t) = v_c(t) + v_D(t)$, para encontrar la parte continua de la misma $v_c(t)$ se considera la condición definida en (3.33), es decir $\dot{s}(t) = 0$ y $v_D(t) = 0$.

Reemplazando (3.33) en (3.48).

$$0 = (\dot{h}_d(t) - J_{ee} U(t)) + \lambda \tilde{h}(t) \quad (3.49)$$

Desarrollando (3.49):

$$U(t) = J_{ee}^+ (\dot{h}_d(t) + \lambda \tilde{h}(t))$$

$$U(t) = v_c(t)$$

$$v_c(t) = J_{ee}^T (J_{ee} J_{ee}^T)^{-1} (\dot{h}_d(t) + \lambda \tilde{h}(t)) \quad (3.50)$$

Para la parte discontinua se considera la función de Lyapunov (3.36) y su derivada (3.37), reemplazando (3.37) en (3.50):

$$\dot{V} = s^T [(\dot{h}_d(t) - J_{ee} U(t)) + \lambda \tilde{h}(t)] \quad (3.51)$$

Sustituyendo en (3.38) la ley de control definida por $U(t) = v_c(t) + v_D(t)$:

$$\dot{V} = s^T [\dot{h}_d(t) - J_{ee} \{u_c(t) + u_D(t)\} + \lambda \tilde{h}(t)] \quad (3.52)$$

Reemplazando (3.50) en (3.52):

$$\dot{V} = -s^T J_{ee} u_D(t) \quad (3.53)$$

Para cumplir la condición de estabilidad $\dot{V} < 0$, y reducir el efecto chattering, se propone la parte discontinua definida en (3.43). De esta manera la ley de control queda definida por:

$$U_c(t) = J_{ee}^T (J_{ee} J_{ee}^T)^{-1} \left[\dot{h}_d(t) + \lambda \tilde{h}(t) + k_1 \frac{s(t)}{|s(t)| + \delta} \right] \quad (3.54)$$

Los algoritmos descritos y desarrollados en las secciones anteriores se implementan en Matlab, esta implementación se detalla en la siguiente sección.

3.2 IMPLEMENTACIÓN DE LOS ALGORITMOS DE CONTROL

Como se explicó en la sección 1.10, las pruebas de simulación se realizaron en el software Matlab, utilizando la herramienta Simulink, debido a la facilidad que ofrece su programación gráfica con lenguaje de bloques.

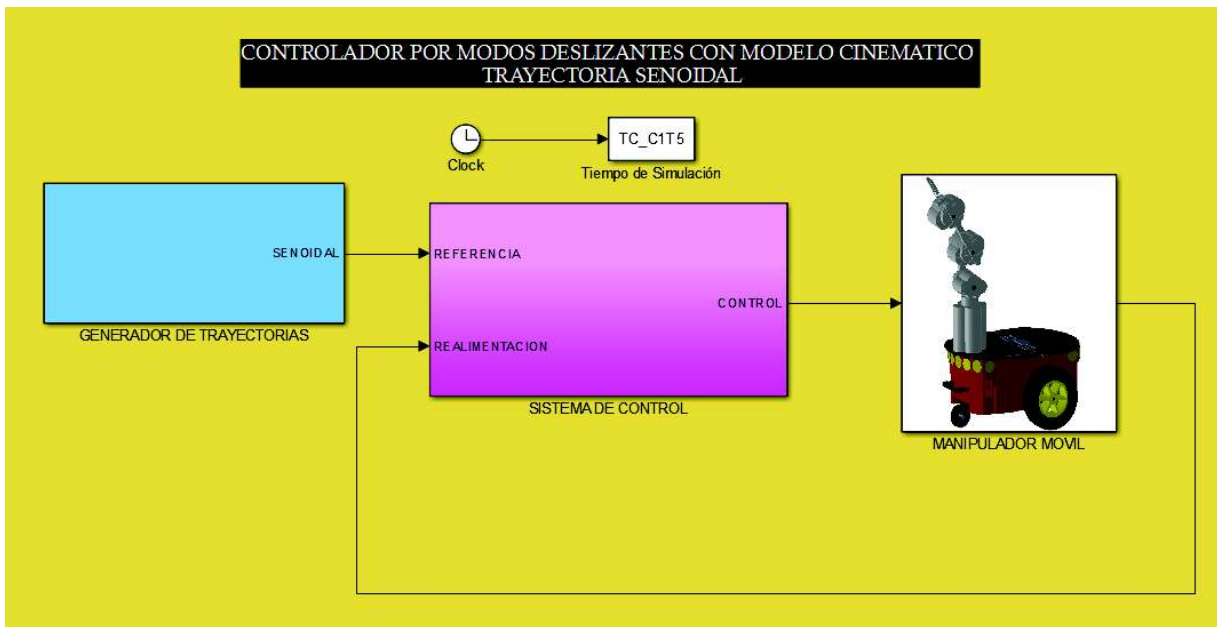


Figura 3.4. Diagrama de bloques del Sistema de control en lazo cerrado.

A continuación se detalla la función que cumple cada bloque del diagrama de la Figura 3.4.

3.2.1 BLOQUE GENERADOR DE TRAYECTORIAS

Este bloque se crea mediante la función "SubSystem" de MatLab y permite obtener la trayectoria deseada para el efector final del manipulador móvil. Las trayectorias se generan a partir de bloques de las funciones básicas de Simulink, como por ejemplo funciones rampa, seno y derivada en tiempo. Las salidas de estas funciones se

agrupan mediante una función MUX, para hacer más fácil la visualización en el diagrama de bloques general, como se muestra en la Figura 3.5.

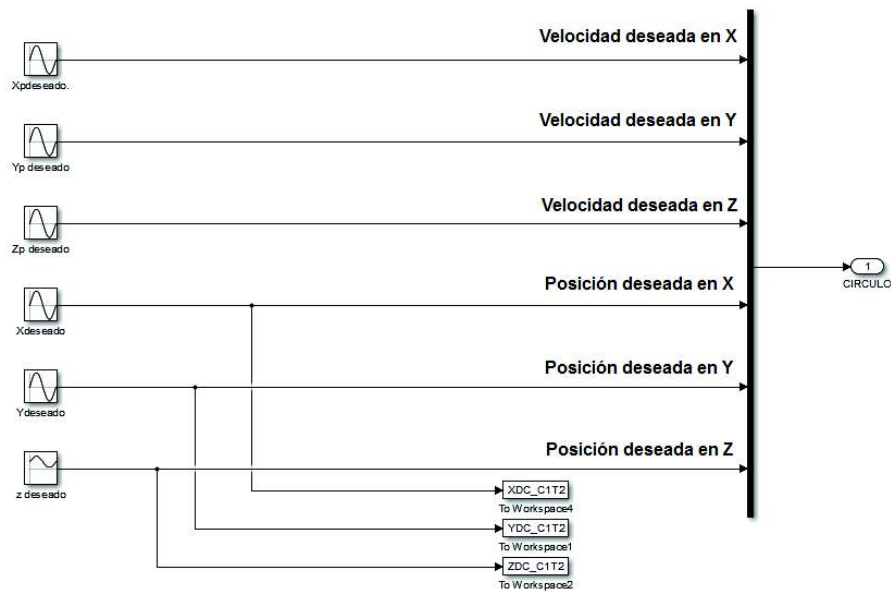


Figura 3.5. Bloque Generador de Trayectorias

3.2.2 BLOQUE SISTEMA DE CONTROL

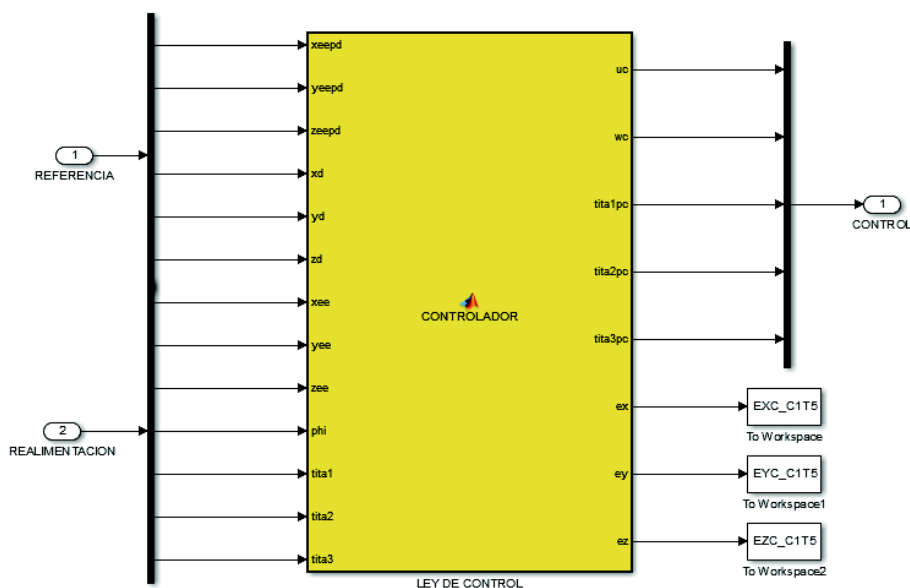


Figura 3.6. Bloque Sistema de Control

En este bloque se utiliza la herramienta “MATLAB Function” para implementar los algoritmos de control, como se observa en la Figura 3.6. Este bloque recibe señales

de la trayectoria deseada, así como las variables del robot. Sus salidas son las señales de control que van hacia el bloque del manipulador móvil y los errores de control para ser graficados en el espacio de trabajo de Matlab.

3.2.3 BLOQUE MANIPULADOR MÓVIL

En este bloque, mediante la herramienta “MATLAB Function” se implementa la cinemática directa, desarrollada en sección 2.1.3, así como la cinemática inversa, explicada en la sección 1.6.1. En la Figura 3.7 se muestra el bloque del manipulador móvil.

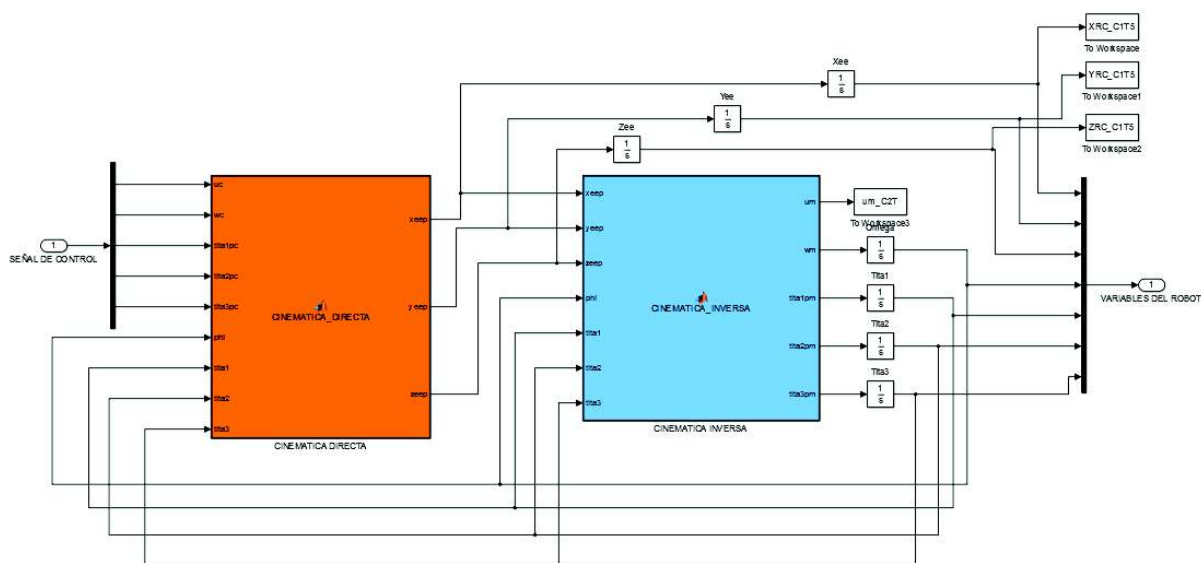


Figura 3.7. Bloque Manipulador Móvil

3.2.4 BLOQUE TIEMPO DE SIMULACIÓN

El bloque tiempo de simulación se muestra en la Figura 3.8.

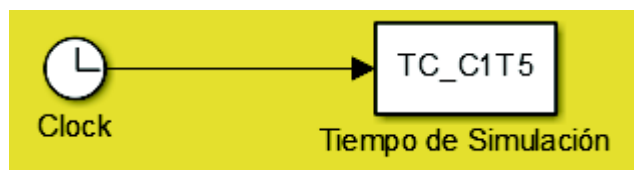


Figura 3.8. Bloque Tiempo de Simulación

En este bloque mediante las funciones “Clock” y “To workspace” se crea un vector en el espacio de trabajo de Matlab, que representa el tiempo que dura la simulación, este vector es necesario para realizar las gráficas de la evolución del error en el tiempo.

3.2.5 DESARROLLO DEL ENTORNO VIRTUAL

En el Anexo B, se muestra el desarrollo del manipulador en el software Solidworks, para que este archivo pueda ser reconocido por Matlab se lo guarda en formato VRML (Virtual Reality Markup Language), el lenguaje de realidad virtual utilizado por Matlab. En la Figura 3.9 se observar el manipulador móvil en el editor VRML de Matlab, en este editor se construye el entorno virtual donde se desplaza el manipulador móvil.

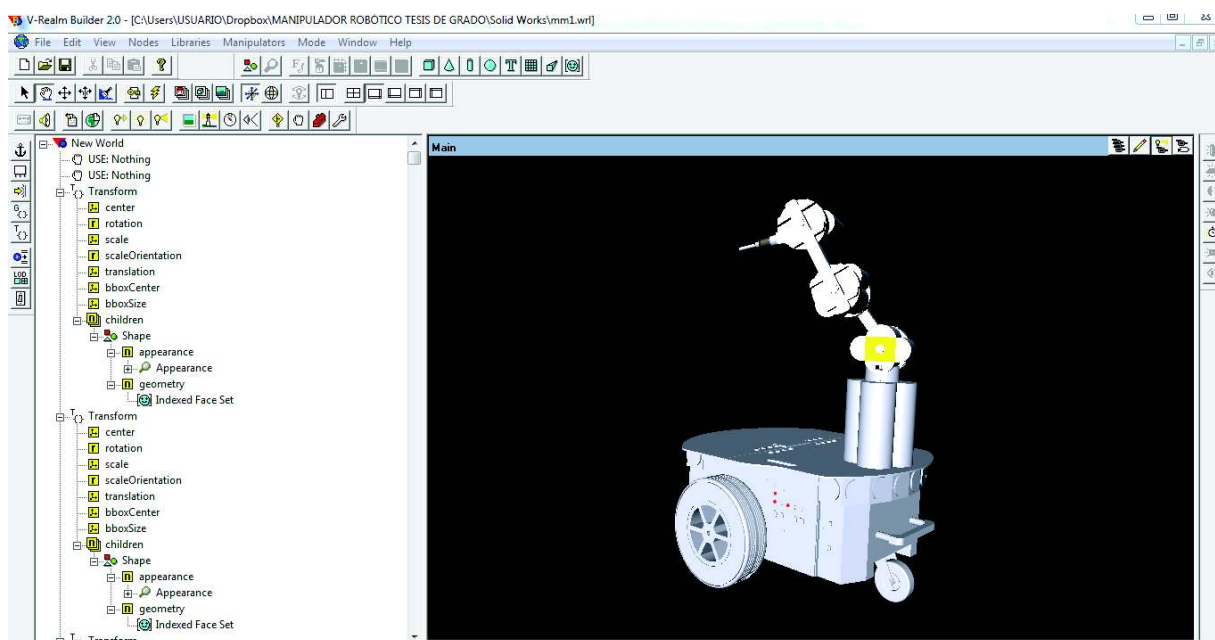


Figura 3.9. Manipulador móvil en el editor de VRML de Matlab

En este editor se construye el mundo virtual, mediante las herramientas que proporciona este software.

En primer lugar se crea una relación “padre – hijo” entre los componentes del manipulador móvil, esto significa que el movimiento de las piezas declaradas en el programa como “hijos” dependerán del movimiento de las piezas declaradas como “padre”. Después de esto se construyen las paredes, el piso y se agregan puntos de vista para tener una mejor apreciación del entorno virtual. Finalmente se agregan

colores y texturas a cada componente del entorno creado, como se muestra en la Figura 3.10.

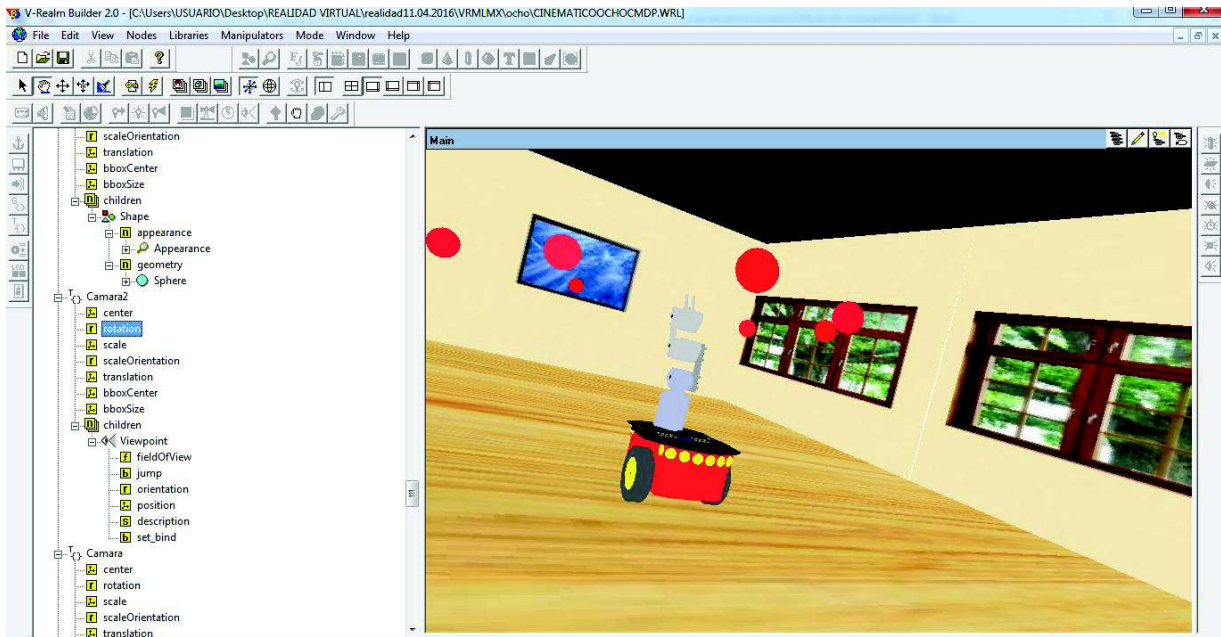


Figura 3.10. Manipulador móvil en el editor de VRML de Matlab

Este mundo virtual recibe información de las simulaciones del control de posición y seguimiento de trayectorias del manipulador móvil, para tener una mejor visualización del control implementado.

CAPÍTULO 4

PRUEBAS DE SIMULACIÓN

En este capítulo se detallan las pruebas realizadas mediante simulación en el software Matlab, se presentan los resultados de la aplicación de los sistemas diseñados tanto para control de posición como seguimiento de trayectoria en los modelos obtenidos para el manipulador móvil.

Los cuatro controladores implementados son:

- Control basado en mínima norma tipo proporcional integral diferencial (PID)
- Control basado en álgebra lineal (EULER)
- Control por modos deslizantes, superficie tipo proporcional (CMD-P)
- Control por modos deslizantes, superficie tipo proporcional integral (CMD-PI)

Las pruebas realizadas con cada uno de los controladores diseñados son las siguientes:

- Prueba 1. Realidad Virtual
- Prueba 2. Control de Posición aplicado al modelo cinemático.
- Prueba 3. Control de Posición aplicado al modelo dinámico.
- Prueba 4. Seguimiento de Trayectorias aplicado al modelo cinemático
- Prueba 5. Seguimiento de Trayectorias aplicado al modelo dinámico.

En total se realizan 16 pruebas, para cada una de ellas se presenta los resultados de la simulación en realidad virtual, el análisis de la evolución en el tiempo del error de posición y seguimiento de trayectorias del efector final para cada eje, finalmente se analiza el rendimiento de los controladores implementados, mediante el índice de rendimiento IAE.

PRUEBA 1: REALIDAD VIRTUAL VRML

En esta sección se presenta la visualización de los resultados de los experimentos en realidad virtual de Matlab. Para simular el movimiento del robot, se exportan las

variables de posiciones y ángulos de rotación desde Simulink al entorno creado en realidad virtual. Esto se realiza para el control de posición y para el control de seguimiento de trayectorias: i) seno, ii) cuadrada, iii) tipo ocho y iv) circular.

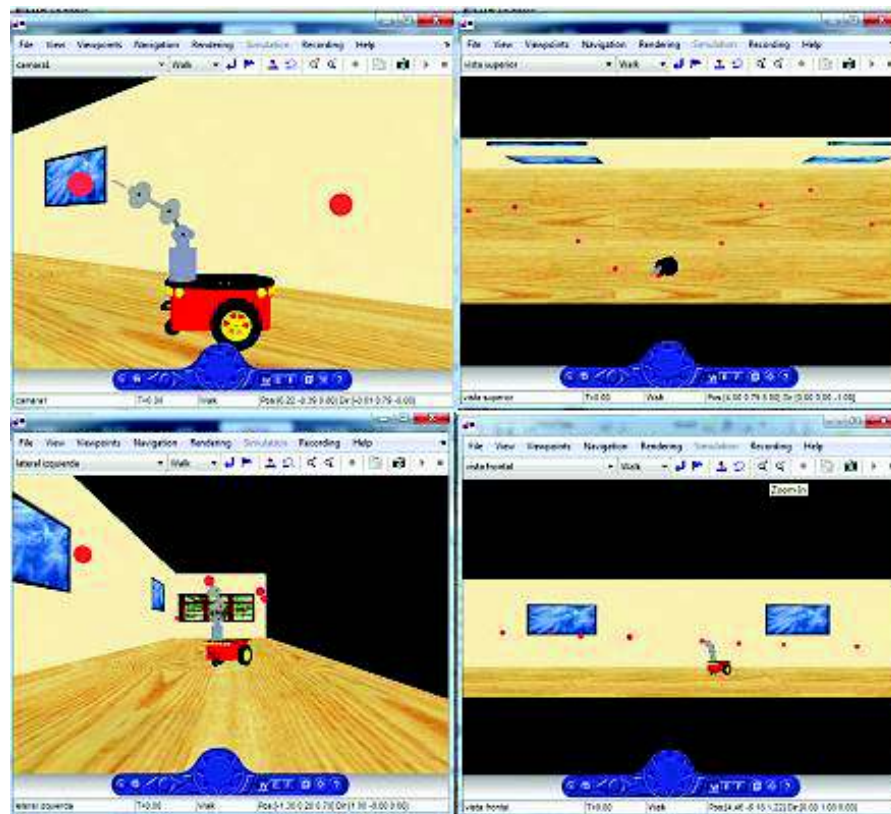


Figura 4.1. Seguimiento de trayectoria seno en realidad virtual

En la Figura 4.1 se puede observar la simulación del seguimiento de trayectoria seno del efector final, en el ambiente de realidad virtual diseñado y desde diferentes puntos de vista. El entorno virtual implementado se puede visualizar desde cuatro perspectivas o puntos de vista: superior, frontal, lateral y un punto de vista que se mueve junto con el manipulador móvil. Las trayectorias que describe el efector final del manipulador móvil se visualizan en el entorno virtual mediante esferas colocadas una a continuación de otra.

Con esto se consigue una mejor visualización del comportamiento de los controladores, esto constituye una primera aproximación a la implementación del control en un sistema real.

En las Figuras 4.2, 4.3 y 4.4 se puede observar la simulación en realidad virtual para las demás trayectorias.

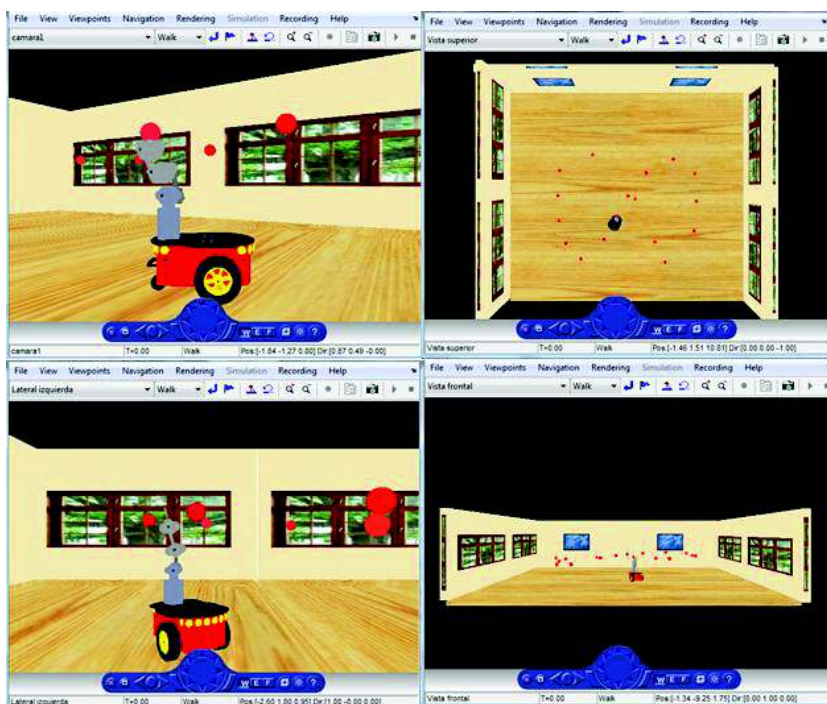


Figura 4.2. Seguimiento de trayectoria tipo ocho en realidad virtual

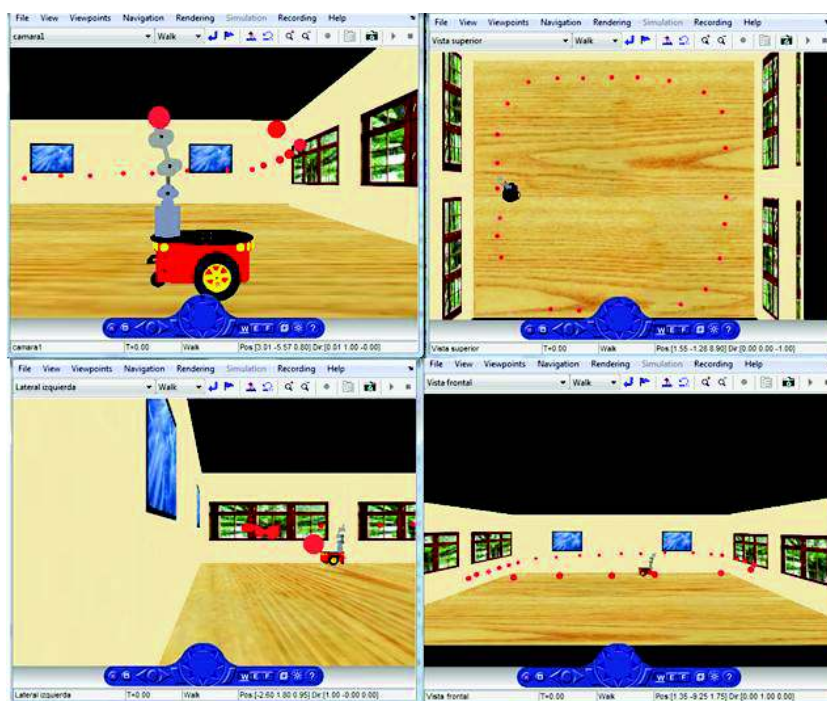


Figura 4.3. Seguimiento de trayectoria cuadrada en realidad virtual

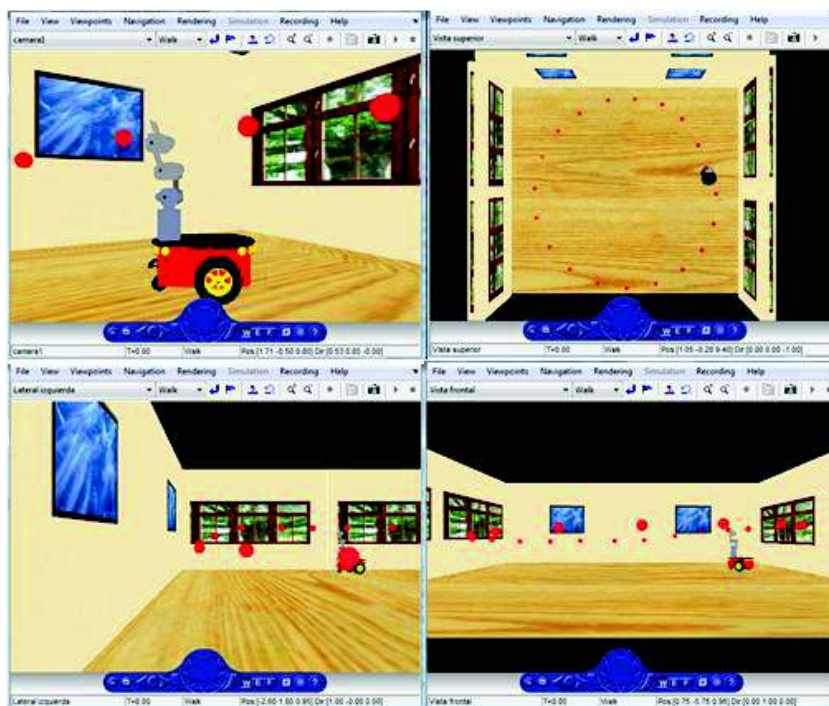


Figura 4.4. Seguimiento de trayectoria circular en realidad virtual

Los resultados obtenidos de la simulación en realidad virtual permitieron observar el comportamiento de los componentes del manipulador ante las acciones de control, lo que puede servir como referencia para la implementación en un robot real en futuros trabajos.

Parámetros Geométricos y dinámicos

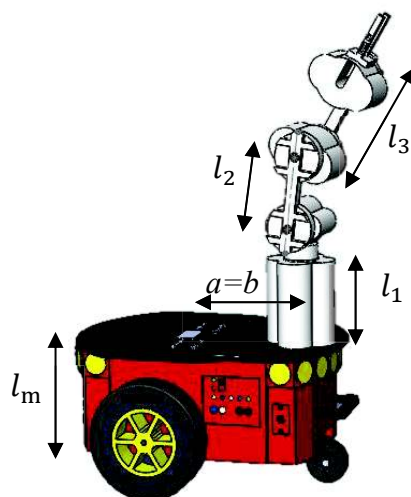


Figura 4.5. Parámetros geométricos del manipulador móvil

El manipulador móvil simulado en el presente proyecto se compone de dos robots comerciales, como se muestra en la Figura 4.5.

Como base móvil se utiliza el robot PIONEER 3-DX y el brazo manipulador MOVER 4, presentados en las secciones 1.3.1 y 1.4.1 respectivamente. Los parámetros geométricos de estos robots son necesarios para la simulación del modelo del manipulador móvil, presentado en el Capítulo 2. Las medidas geométricas de los robots utilizados para la simulación del manipulador móvil, se presentan en la tabla 4.1.

Tabla 4.1 Medidas geométricas del manipulador móvil

Parámetro	Medida [m]
a	0.205
b	0.205
l_1	0.206
l_2	0.190
l_3	0.220
l_m	0.237

Los parámetros dinámicos identificados para el robot móvil PIONEER 3-DX y validados en [41], se presentan en la tabla 4.2.

Tabla 4.2. Parámetros dinámicos identificados para el robot Pioneer 3-DX

Parámetro	Valor
B_1	0.2604
B_2	0.2509
B_3	-0.000499
B_4	0.9965
B_5	0.00263
B_6	1.0768

4.1 PRUEBA 2: CONTROL DE POSICIÓN APLICADO AL MODELO CINEMÁTICO

Como se explicó en la sección 1.7.4, la finalidad de este tipo de control es alcanzar un punto de coordenadas (x, y, z) . El objetivo de control es que el efector final del

manipulador móvil, alcance el punto P_{REF} de coordenadas: $x = 20m$, $y = 15m$, $z = 0.75m$, desde una posición inicial $x_0 = 2m$, $y_0 = -0.5m$, $z_0 = 0.6m$, y para esto se implementó cuatro técnicas de control: i) Control tipo PID basado en mínima norma, ii) Control basado en álgebra lineal, iii) Control basado en modos deslizantes tipo P, iv) Control basado en modos deslizantes tipo PI, estas técnicas se aplican al modelo cinemático y modelo dinámico.

Los valores de las constantes de calibración de cada controlador, se detallan en la tabla 4.3.

Estas constantes fueron determinadas por prueba y error basado en el índice de rendimiento integral del error absoluto IAE, se realizaron pruebas con distintos valores de constantes de calibración y se eligió las constantes para las cuales el valor del índice IAE fue menor.

Los rangos de valores que pueden tomar estas constantes se explican en el análisis de estabilidad de cada controlador, en el Capítulo 3.

Tabla 4.3 Constantes de calibración para control de posición

Controlador	Constantes		
PID	$k_x = 5$	$k_y = 5$	$k_z = 5$
EULER	$k_x = 0.8$	$k_y = 0.8$	$k_z = 0.8$
CMD-P	$k_1 = 5$		$\delta = 0.1$
CMD-PI	$k_1 = 8$	$\lambda = 6$	$\delta = 0.85$

En la Figura 4.6 se muestra la evolución de la trayectoria del efector final para los cuatro controladores. Como se observa, el punto P_{REF} es alcanzado por todos los controladores, también se puede observar que cada controlador tiene un comportamiento diferente para alcanzar el objetivo de control. El efector final del manipulador móvil, para el caso de los controladores PID y EULER describe una línea recta entre los puntos inicial y final, mientras que para los controladores CMD-P y

CMD-PI describe trayectorias curvas, por lo que en el caso de los dos últimos controladores tarda un mayor tiempo en completar la tarea.

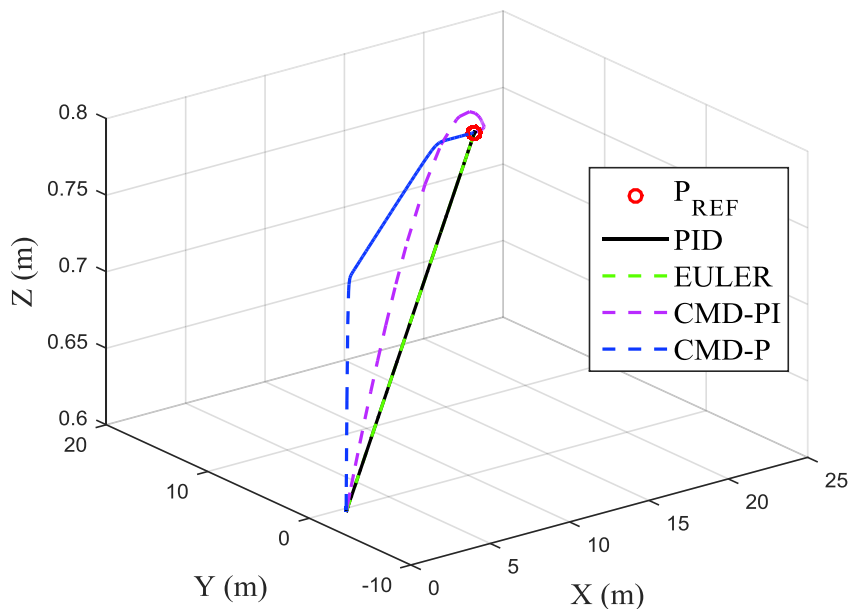


Figura 4.6. Evolución de trayectorias del efector final, para el control de Posición

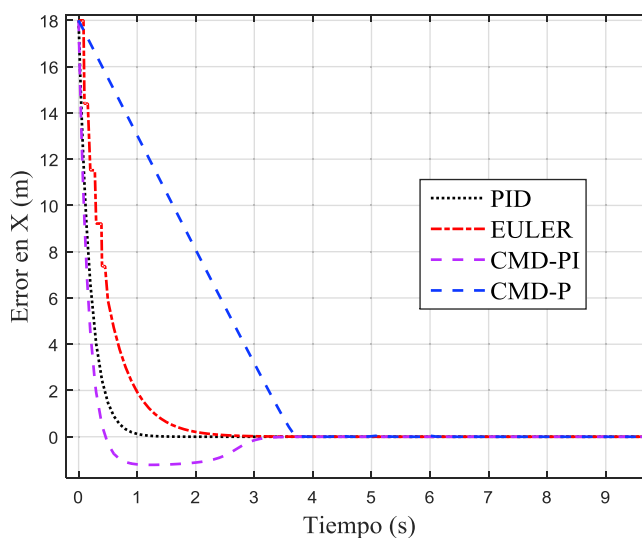


Figura 4.7. Evolución del error en X, para el control de posición

En las Figuras 4.7 y 4.8 se presenta la evolución en el tiempo del error de posición del efector final para los ejes X e Y respectivamente. Se puede ver que el manipulador móvil con el controlador CMD-P, tarda un mayor tiempo en alcanzar el objetivo de control, mientras que con el controlador PID lo alcanza de una manera más rápida.

También se puede observar que los cuatro controladores cumplen con el objetivo de control, puesto que el error de cada uno tiende a cero a medida que transcurre el tiempo de simulación.

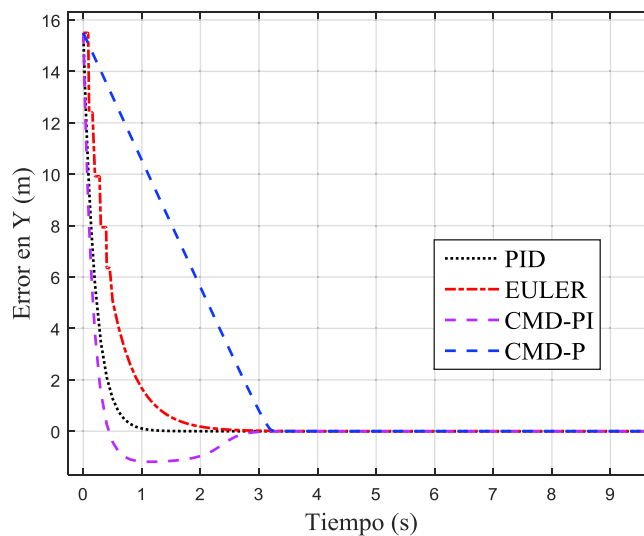


Figura 4.8. Evolución del error en Y, para el control de posición

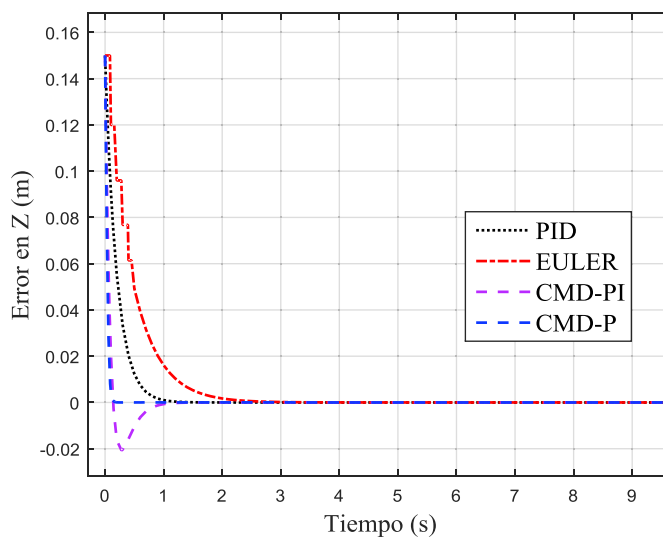


Figura 4.9. Evolución del error en Z, para el control de posición

En la figura 4.9 se puede observar que el controlador CMD-P alcanza el objetivo en un tiempo menor.

A continuación se presenta el análisis de rendimiento de los controladores implementados para el control de posición, mediante el índice IAE.

4.1.1 RENDIMIENTO DE LOS CONTROLADORES CON EL MODELO CINEMÁTICO

El índice IAE es la medida de la integral en el tiempo del error, representa el área bajo la curva de la evolución del error en el tiempo, por lo que un índice IAE de valor pequeño indica un menor error y por tanto un mejor desempeño del controlador.

Para el obtener del índice IAE de cada controlador, se aplica la definición dada en la sección 1.8.1, calculando la integral del error en cada uno de los ejes del sistema de coordenadas, es decir se calculan tres índices IAE_x , IAE_y e IAE_z , para obtener el valor del índice total se calcula la norma euclidiana del vector formado por los tres índices $IAE_{xyz} = (IAE_x, IAE_y, IAE_z)$.

$$IAE = \|IAE_{xyz}\| \quad (4.1)$$

Este análisis se realiza para el control de posición y seguimiento de trayectorias del manipulador móvil con el modelo cinemático y dinámico.

En la tabla 4.4 se presenta el índice de rendimiento IAE para cada uno de los controladores implementados en el modelo cinemático.

Tabla 4.4 Índice de rendimiento IAE de cada controlador, para control de posición

IAE PARA CONTROL DE POSICIÓN CON EL MODELO CINEMÁTICO	CONTROLADOR			
	PID	EULER	CMD-P	CMD-PI
	4.80	11.27	40.83	6.38

De los resultados presentados en la tabla 4.4, se puede observar que el controlador PID presenta un mejor rendimiento, ya que su valor de IAE es menor al de los otros controladores. También se observa que el controlador CMD-P tiene el valor más alto de IAE, por lo que su rendimiento es más bajo comparado con el de los demás controladores.

Sin embargo todos los controladores cumplen el objetivo de control, que es posicionar el efector final en el punto de referencia.

4.2 PRUEBA 3: CONTROL DE POSICIÓN APLICADO AL MODELO DINÁMICO

En este experimento se hace un ajuste de las constantes del controlador para alcanzar los objetivos de control. Los valores de las constantes de cada controlador se presentan en la tabla 4.5.

Tabla 4.5 Constantes de calibración para control de posición modelo dinámico

Controlador	Constantes		
PID	$k_x = 2$	$k_y = 2$	$k_z = 2$
EULER	$k_x = 0.85$	$k_y = 0.85$	$k_z = 0.85$
CMD-P	$k_1 = 3$		$\delta = 0.75$
CMD-PI	$k_1 = 4$	$\lambda = 2$	$\delta = 0.75$

El valor de las constantes presentadas en la tabla 4.5 se obtuvo mediante prueba y error basado en el índice integral del error absoluto IAE.

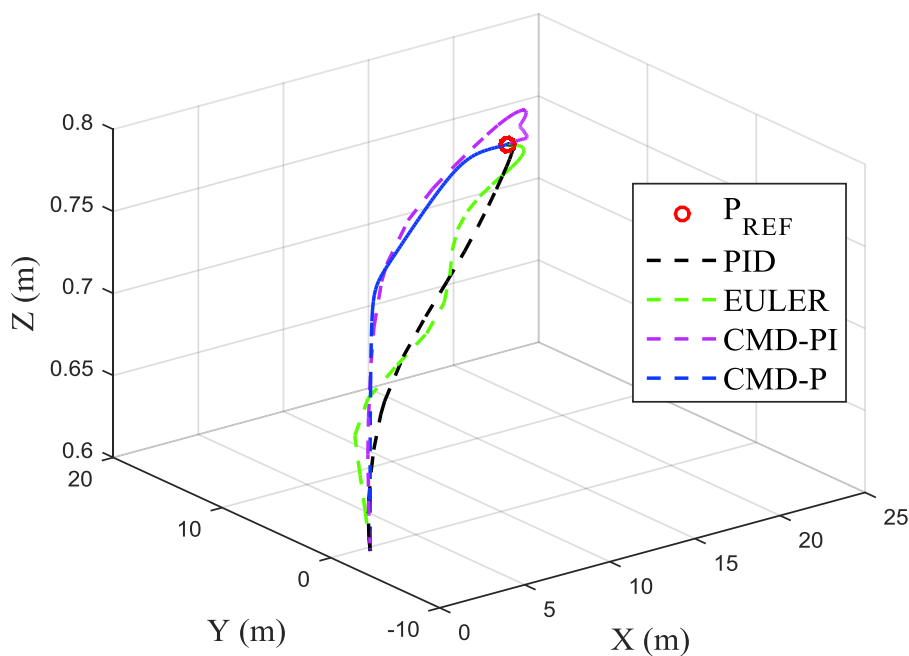


Figura 4.10. Evolución de trayectorias del efector final, para el control de Posición

En la Figura 4.10 se observa la evolución de la trayectoria descrita por el efector final para los cuatro controladores. Se verifica que los controladores alcanzan el objetivo de control. Cada controlador presenta un comportamiento diferente al observado con el modelo cinemático, esto debido a que la dinámica del modelo tiene un comportamiento diferente al modelo cinemático.

En las Figuras 4.11, 4.12 y 4.13 se observa la evolución en el tiempo del error de posición del efector final para los ejes X e Y respectivamente para cada controlador.

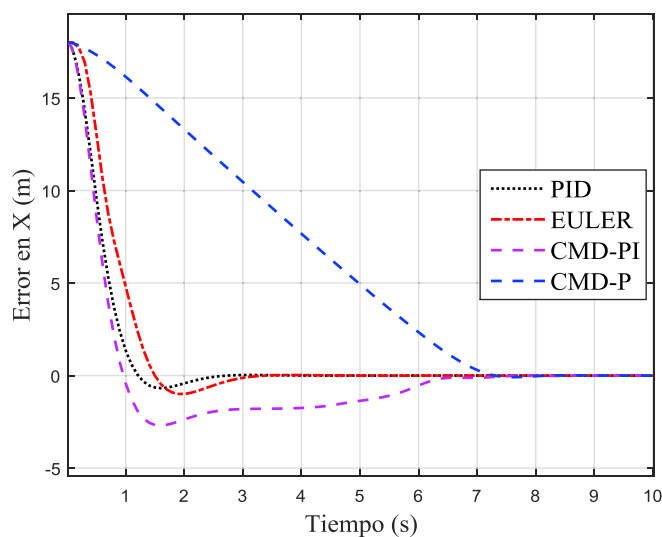


Figura 4.11. Evolución del error en X, para el control de posición

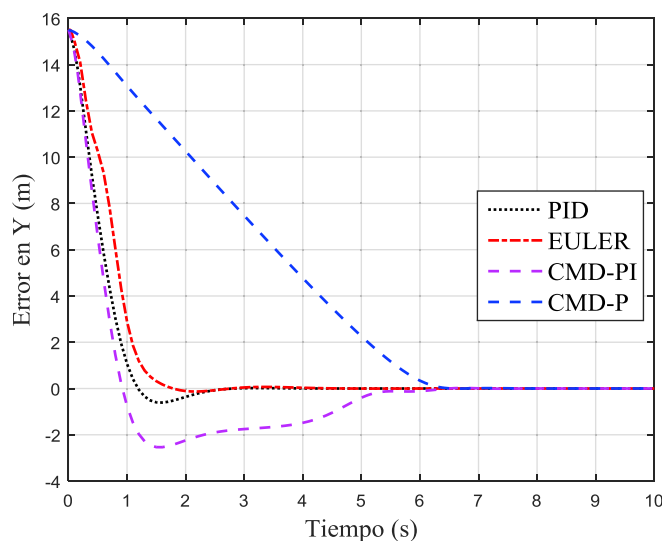


Figura 4.12. Evolución del error en Y, para el control de posición

En los resultados observados se verifica que el error en todos los casos tiende a cero y que en el caso del controlador CMD-P tarda un mayor tiempo en alcanzar el objetivo de control. También se observa que el controlador CMD-PI alcanza el objetivo de control presentando una oscilación.

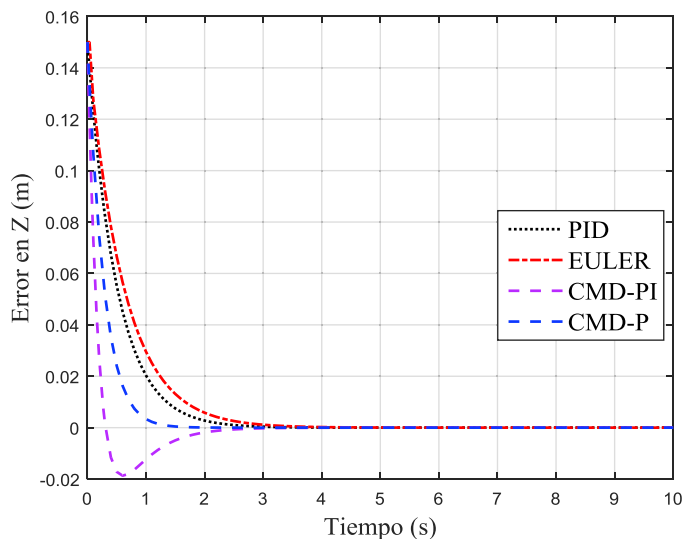


Figura 4.13. Evolución del error en Z, para el control de posición

Se verifica que el error de posición del efector final para los cuatro controladores tiende a cero.

4.2.1 RENDIMIENTO DE LOS CONTROLADORES CON EL MODELO DINÁMICO

Para medir el rendimiento de los controladores implementados se calcula el índice integral de error absoluto IAE para cada controlador, tal como se realizó con el modelo cinemático.

En la tabla 4.6 se presenta el índice de rendimiento IAE para cada uno de los controladores implementados en el modelo dinámico.

Tabla 4.6 Índice de rendimiento IAE de cada controlador, para control de posición

IAE PARA CONTROL DE POSICIÓN CON EL MODELO DINÁMICO	CONTROLADOR			
	PID	EULER	CMD-P	CMD-PI
	13.54	18.08	78.88	22.76

Como se observa en la tabla 4.6 el controlador PID presenta el menor valor de IAE, por lo que tiene un mejor rendimiento que los demás controladores. El controlador CMD-PI presenta un rendimiento más bajo, pues tiene un valor mayor del índice IAE.

De los resultados observados, se verifica que los controladores implementados cumplen el objetivo de control de posición para los modelos cinemático y dinámico.

4.3 PRUEBA 4: SEGUIMIENTO DE TRAYECTORIA APLICADO AL MODELO CINEMÁTICO

El objetivo de control es que el efector final, siga cuatro tipos de trayectorias: i) Seno, ii) Cuadrada, iii) Tipo Ocho y iv) Circular.

Las trayectorias implementadas se describen en la tabla 4.7.

Tabla 4.7. Trayectorias propuestas

Trayectoria	Componentes		
	Eje X	Eje Y	Eje Z
Seno	$f(t) = t$	$f(t) = \text{sen}(t)$	$f(t) = \frac{2}{10} \text{sen}\left(\frac{1}{2}t\right) + \frac{1}{2}$
Tipo Ocho	$f(t) = 2\cos\left(\frac{2}{5}t\right)$	$f(t) = 2\text{sen}\left(\frac{4}{5}t\right)$	$f(t) = \frac{2}{10} \text{sen}\left(\frac{1}{2}t\right) + \frac{1}{2}$
Cuadrada	$f(t) = \begin{cases} 2t & 0 \leq t \leq 2.5 \\ 5 & 2.5 \leq t \leq 5 \\ -2t & 5 \leq t \leq 7.5 \\ 0 & 7.5 \leq t \leq 10 \end{cases}$	$f(t) = \begin{cases} 0 & 0 \leq t \leq 2.5 \\ 2t & 2.5 \leq t \leq 5 \\ 5 & 5 \leq t \leq 7.5 \\ -2t & 7.5 \leq t \leq 10 \end{cases}$	$f(t) = \frac{2}{10} \text{sen}\left(\frac{1}{2}t\right) + \frac{1}{2}$
Circular	$f(t) = 2\text{sen}(0.63t)$	$f(t) = 2\cos(0.63t)$	$f(t) = \frac{2}{10} \text{sen}\left(\frac{1}{2}t\right) + \frac{1}{2}$

Para realizar una adecuada presentación de los resultados experimentales se ha establecido diferentes tiempos del experimento, para las trayectorias seno, cuadrada y circular el tiempo del experimento es 10s, mientras que para la trayectoria tipo ocho el tiempo del experimento es 16s. Recalcar que la función $f(t)$ en el eje Z es la misma para todas las trayectorias, y es una función de amplitud más pequeña debido a que

el rango de operación en este eje es limitado por las propiedades físicas del manipulador móvil.

A continuación se presenta el resultado de la simulación de los controladores diseñados para seguimiento de trayectorias aplicados al modelo cinemático. Para este experimento las constantes de calibración de cada controlador se presentan en la tabla 4.8.

Tabla 4.8 Constantes de calibración para seguimiento de trayectoria, modelo cinemático.

Controlador	Constantes		
PID	$k_x = 5$	$k_y = 5$	$k_z = 5$
EULER	$k_x = 0.7$	$k_y = 0.7$	$k_z = 0.7$
CMD-P	$k_1 = 5$		$\delta = 0.1$
CMD-PI	$k_1 = 8$	$\lambda = 6$	$\delta = 0.85$

El valor de estas constantes se obtuvo mediante prueba y error basado en el índice de rendimiento integral del error absoluto IAE y se mantiene para todas las trayectorias propuestas para el control del modelo cinemático.

4.3.1 SEGUIMIENTO DE TRAYECTORIA SENO

Las coordenadas (x_0, y_0, z_0) de la posición inicial del efector final del manipulador móvil son $(2, -0.5, 0.6)[m]$.

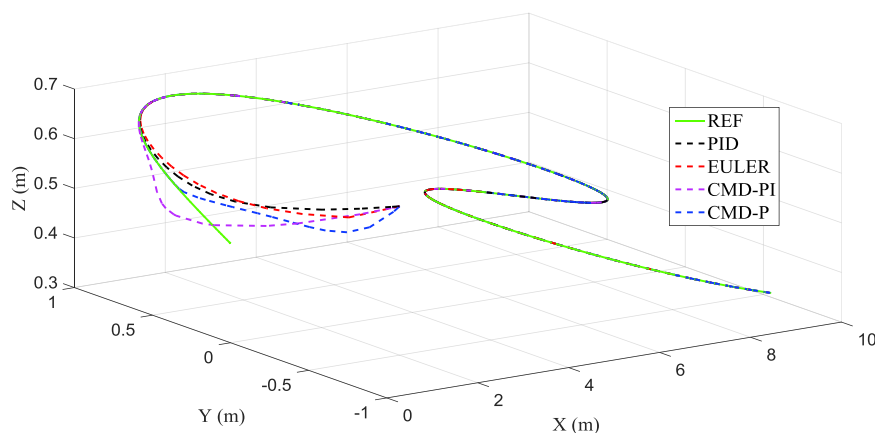


Figura 4.14. Evolución de la trayectoria seno del efector final en 3D

La Figura 4.14 muestra el seguimiento de la trayectoria seno del efector final para cada controlador.

Cuando se realiza el control sobre el modelo cinemático se puede observar que todos los controladores alcanzan la trayectoria deseada.

En la Figura 4.15 se puede observar la trayectoria descrita por el efector final en el plano XY, como se observa, cada controlador alcanza la trayectoria deseada de diferente manera y una vez alcanzada se mantienen sobre la misma durante todo el tiempo de simulación.

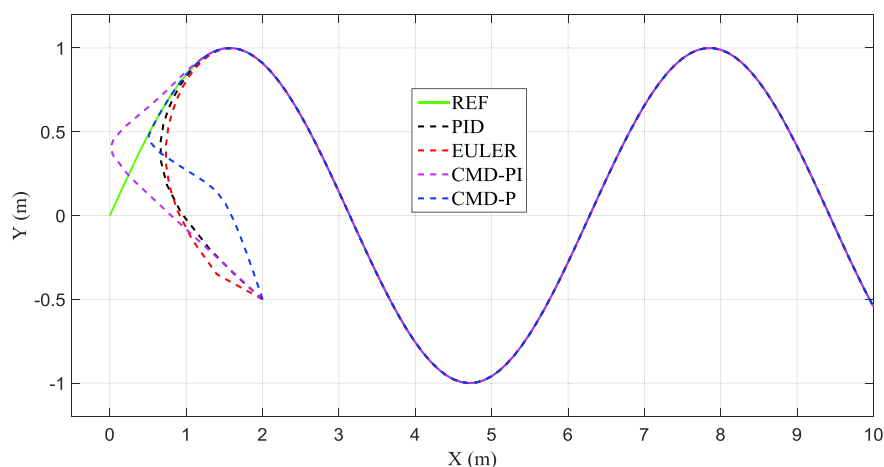


Figura 4.15. Evolución de la trayectoria seno del efector final en el plano XY

En las Figuras 4.16, 4.17 y 4.18 se presentan las evoluciones del error en función del tiempo, para la posición del efector final en los ejes X,Y,Z respectivamente.

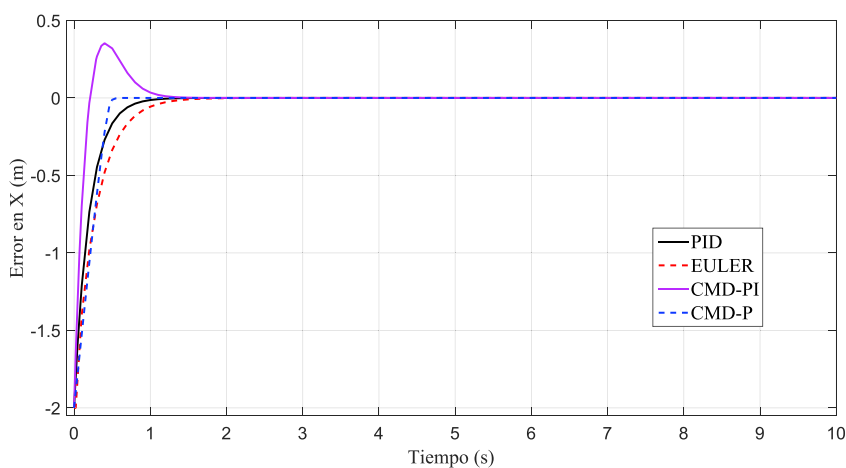


Figura 4.16. Error de trayectoria en el eje X

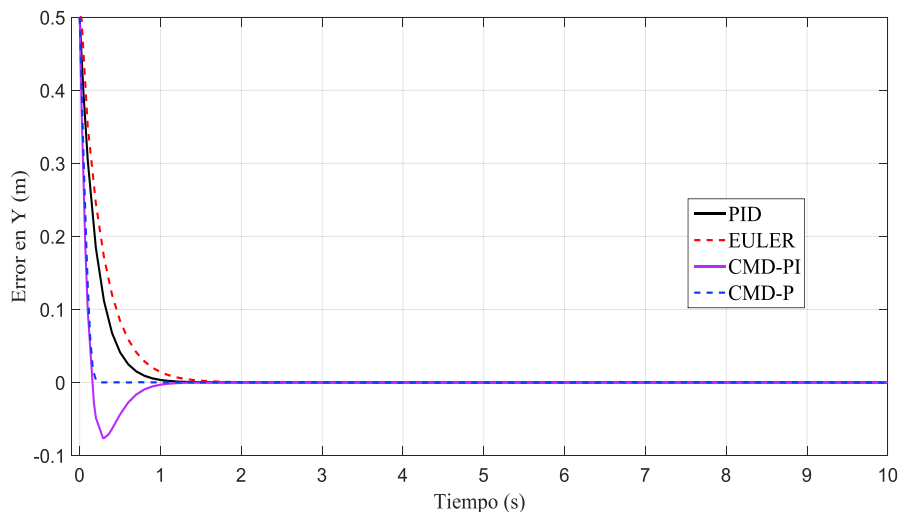


Figura 4.17. Error de trayectoria en el eje Y

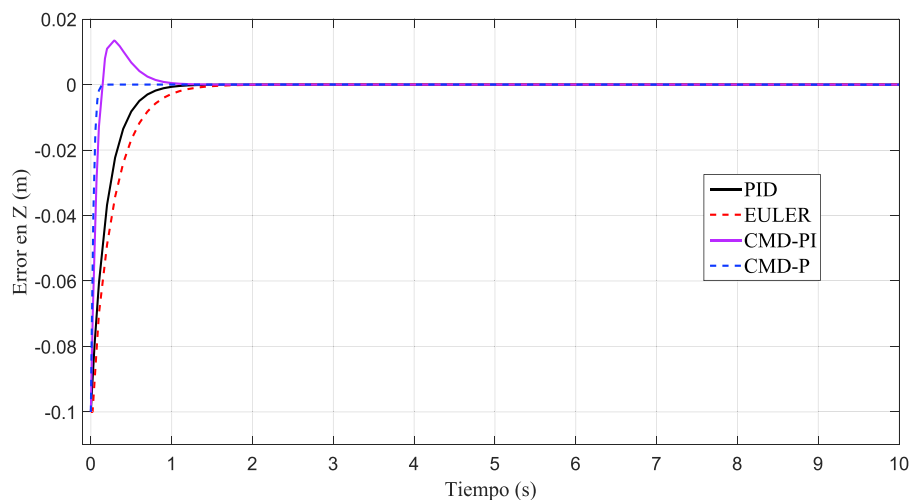


Figura 4.18. Error de trayectoria en el eje Z

Del análisis del error de posición en los tres ejes se puede concluir que los cuatro controladores diseñados cumplen el objetivo de control, que es el seguimiento de trayectoria, destacando los controladores por modos deslizantes CMD-P y CMD-PI, que se estabilizan en un tiempo menor. La cuantificación del rendimiento de cada controlador se analizará posteriormente.

4.3.2 SEGUIMIENTO DE TRAYECTORIA TIPO OCHO

Para esta trayectoria, el efector final se encuentra inicialmente en el punto de coordenadas $(1, -0.5, 0.6)$ [m].

La Figura 4.19 presenta la trayectoria del efector final para una trayectoria tipo ocho en tres dimensiones para los cuatro controladores. Se observa que todos los controladores alcanzan la trayectoria propuesta.

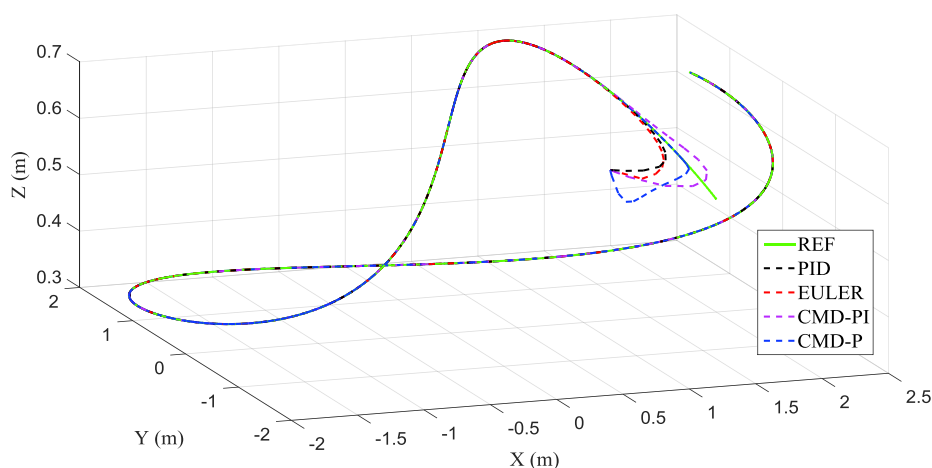


Figura 4.19. Evolución de la trayectoria tipo ocho del efector final en 3D

En la Figura 4.20, se observa la trayectoria del efector final en el plano XY, para los cuatro controladores. Se verifica que el comportamiento del controlador CMD-PI presenta un sobrepico antes de alcanzar el seguimiento de trayectoria, mientras los otros tres controladores alcanzan la trayectoria con pequeños sobrepicos.

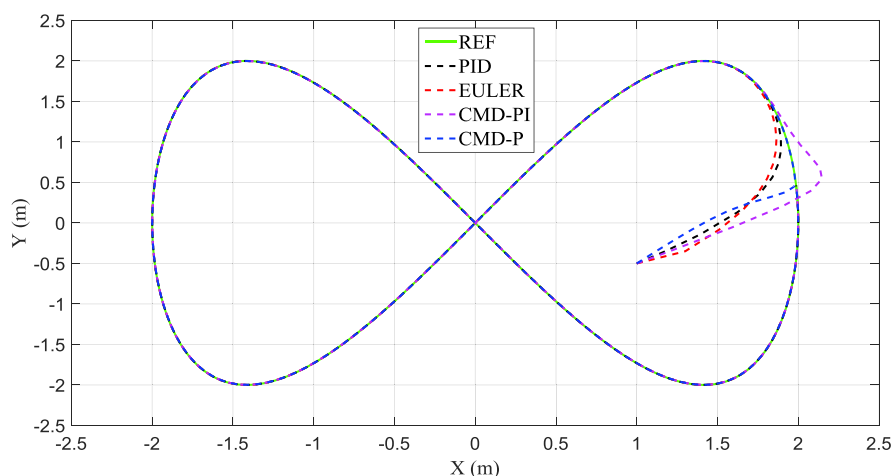


Figura 4.20. Evolución de la trayectoria tipo ocho del efector final en el plano XY

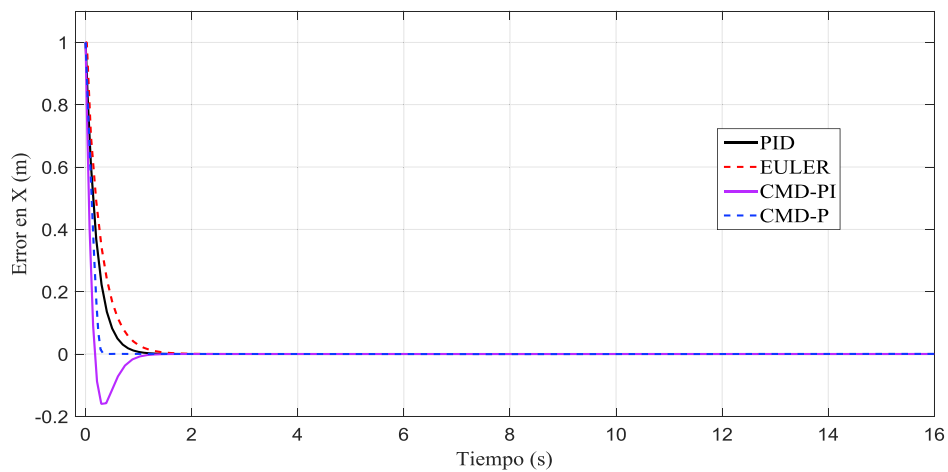


Figura 4.21. Error de trayectoria en el eje X

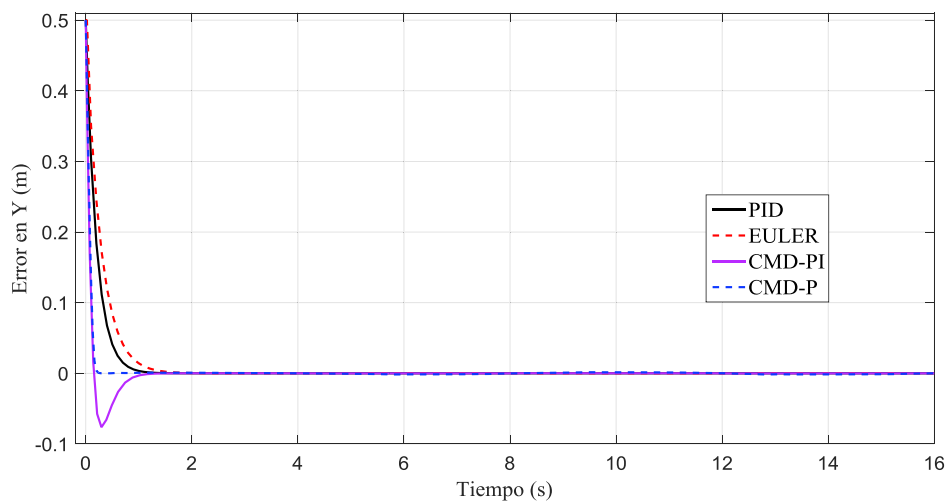


Figura 4.22. Error de trayectoria en el eje Y

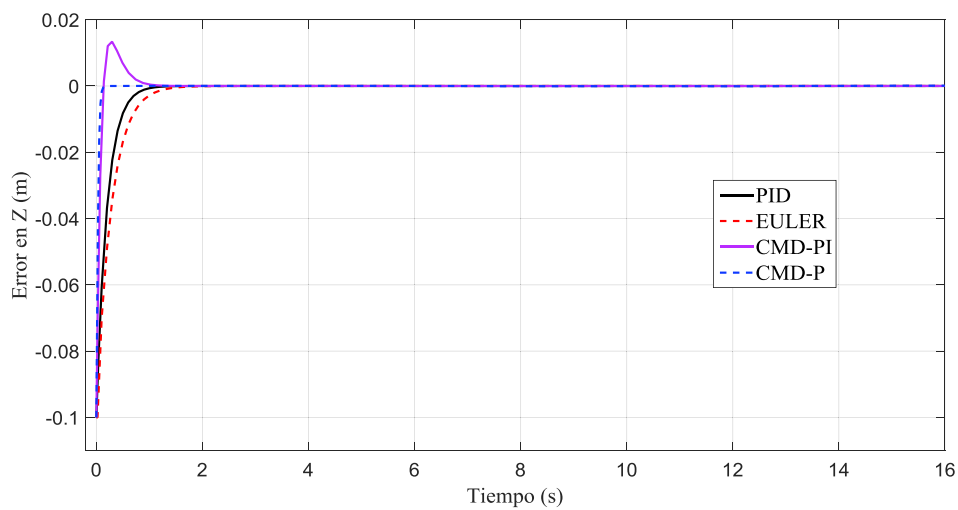


Figura 4.23. Error de trayectoria en el eje Z

En las figuras 4.21, 4.22 y 4.23 se presenta la evolución del error en X, Y, y Z respectivamente, para el seguimiento de trayectoria tipo ocho para los cuatro controladores. Como se puede observar para el controlador CMD-P el error es cero, lo que implica que el efector final alcanza la trayectoria en un tiempo de 0.4 [s], mientras que los controladores PID y EULER se estabilizan en 2.05 y 1.55 [s] respectivamente. En el controlador CMD-PI se observa que alcanza el error cero más rápido que los otros controladores, pero se estabiliza en 1.5 [s].

Del análisis de la evolución del error de posición en el tiempo, los resultados muestran que los cuatro controladores cumplen con el objetivo de control, más adelante se presentará el análisis de rendimiento de cada controlador mediante el índice IAE.

4.3.3 SEGUIMIENTO DE TRAYECTORIA CUADRADA

Para esta trayectoria, el efector final se encuentra inicialmente en el punto de coordenadas (2,4,0.6) [m].

En la Figura 4.24 se observa el seguimiento de la trayectoria cuadrada en tres dimensiones, al aplicar los controladores diseñados al modelo cinemático, se puede observar que los cuatro controladores logran alcanzar la trayectoria.

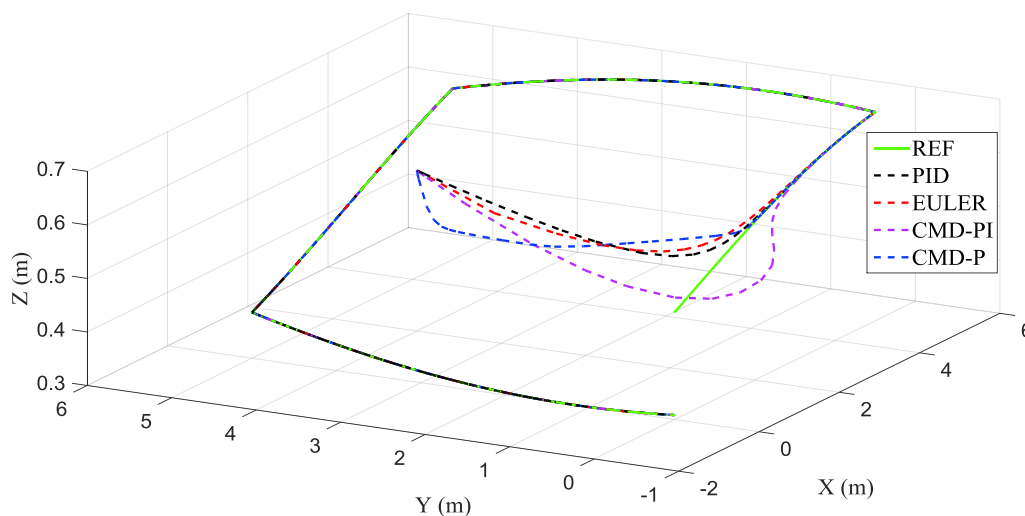


Figura 4.24. Evolución de la trayectoria cuadrada del efector final en 3D

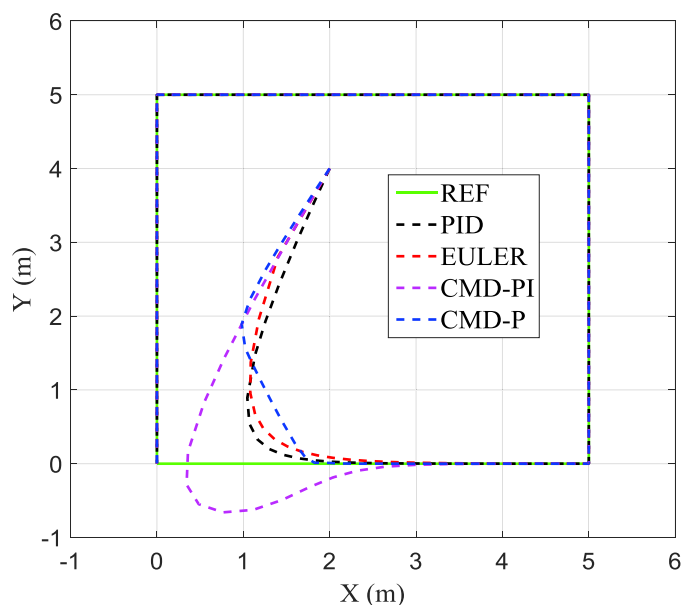


Figura 4.25. Evolución de la trayectoria cuadrada del efector final en el plano XY

La Figura 4.25 muestra el seguimiento de trayectoria cuadrada en el plano XY, se puede apreciar que todos los controladores alcanzan la trayectoria y se mantienen sobre la misma, a pesar del cambio brusco que las esquinas de la trayectoria representan para el controlador.

Las Figuras 4.26, 4.27 y 4,28 muestran la evolución del error en el tiempo, para la posición del efector final en los ejes X, Y y Z respectivamente.

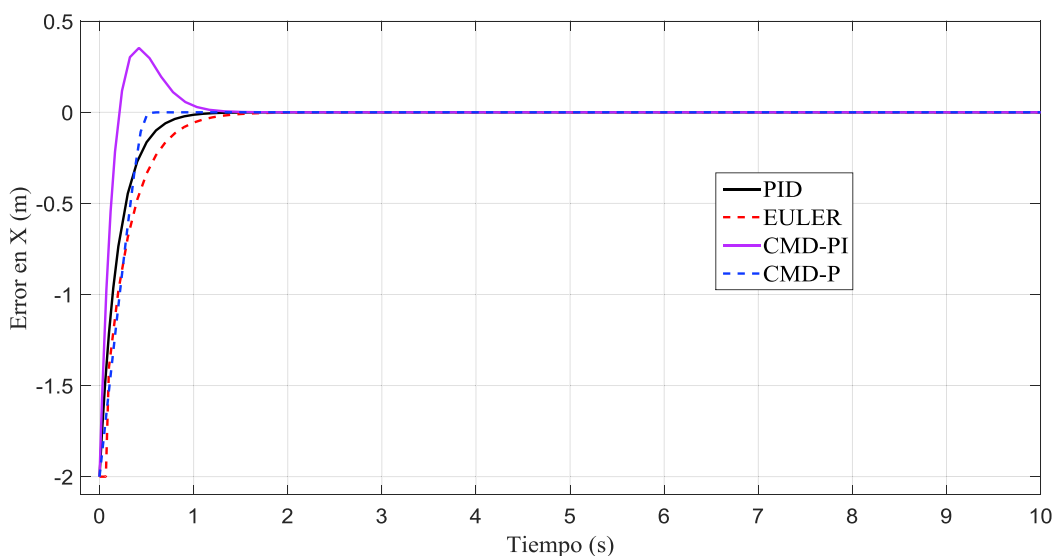


Figura 4.26. Error de trayectoria en el eje X

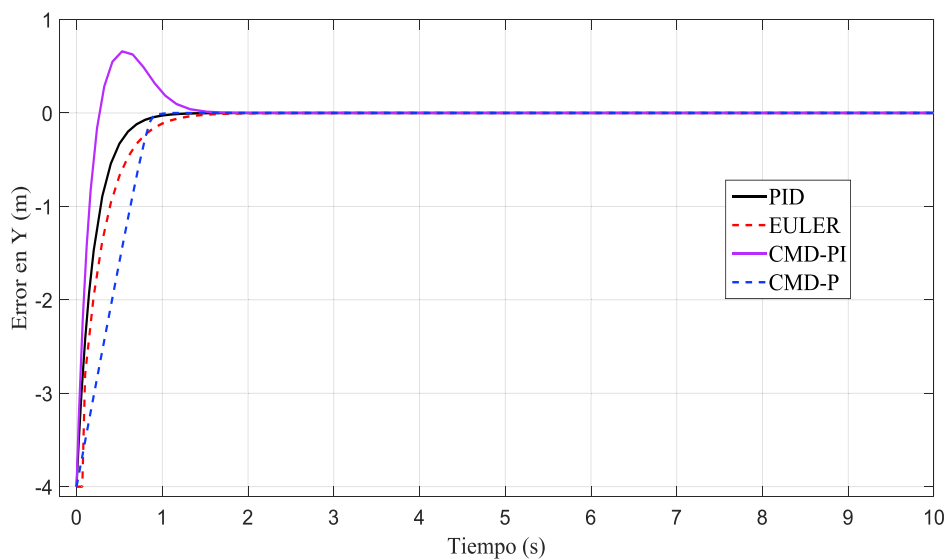


Figura 4.27. Error de trayectoria en el eje Y

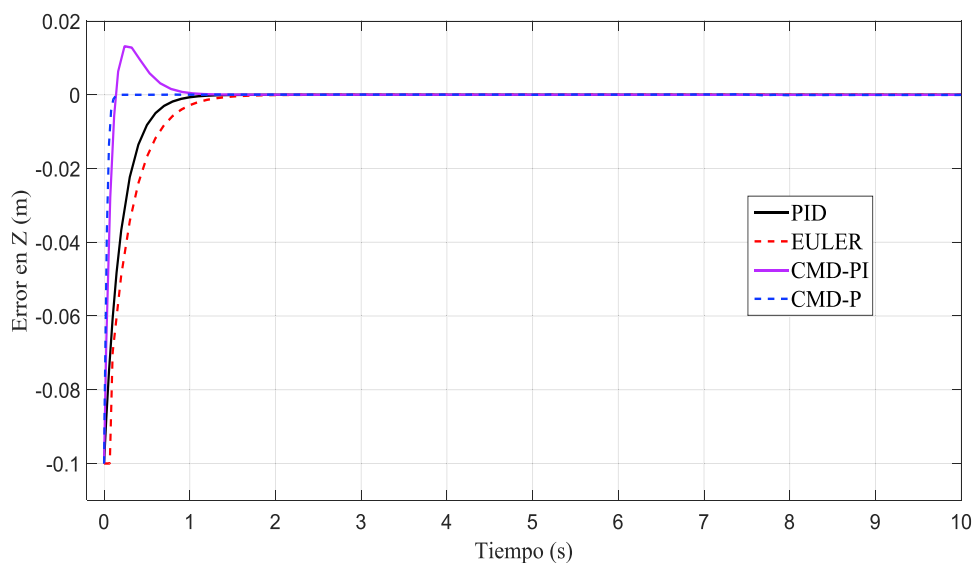


Figura 4.28. Error de trayectoria en el eje Z

Se observa para el controlador PID un tiempo de establecimiento de 1.3 [s], para el controlador EULER es de 1.6 [s] y para el controlador CMD-P es de 0.7 [s]. Para el controlador CMD-PI se observa un sobrepico antes de estabilizarse en 1.2 [s]. En la figura 4.24 se presenta la evolución del error en Z, se observa que es similar al de la trayectoria seno y ocho.

De los resultados observados para esta trayectoria, se puede concluir que los cuatro controladores cumplen el objetivo de control en un tiempo menor a 1.6 [s], y que se mantienen sobre la trayectoria durante todo el tiempo de simulación.

4.3.4 SEGUIMIENTO DE TRAYECTORIA CIRCULAR

Para esta trayectoria, el efector final se encuentra inicialmente en el punto de coordenadas $(-0.5, 0.5, 0.6)$ [m].

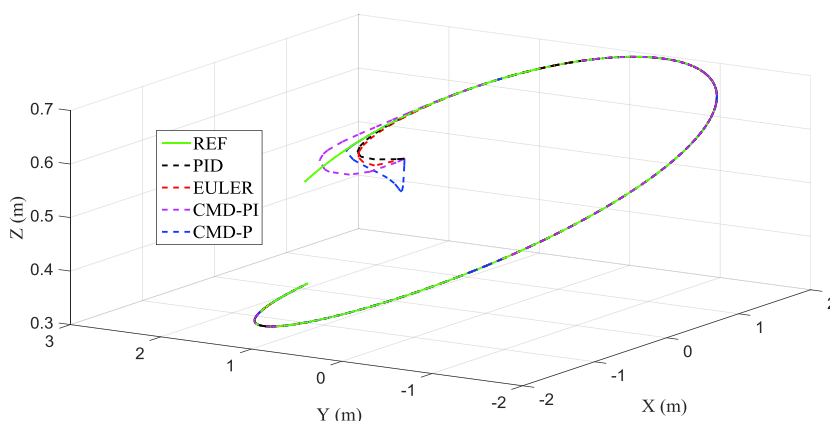


Figura 4.29. Evolución de la trayectoria cuadrada del efector final en 3D

En la Figura 4.29 se presenta la trayectoria circular descrita por el efector final en tres dimensiones, donde se observa que los cuatro controladores cumplen el objetivo de control. En la Figura 4.30 se muestra la trayectoria del efector final en el plano XY, se verifica que el controlador CMD-PI presenta un sobrepico antes de estabilizarse.

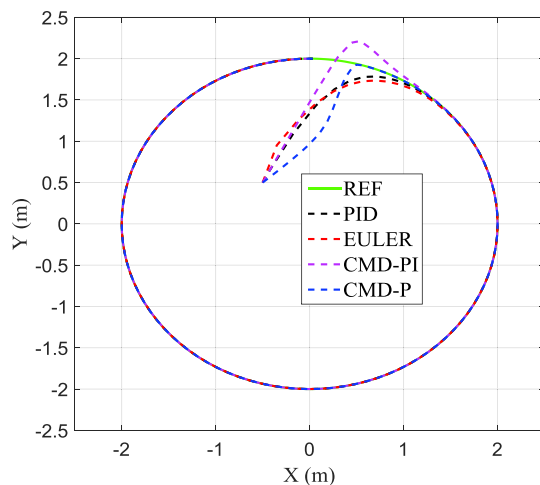


Figura 4.30. Evolución de la trayectoria circular del efector final en el plano XY

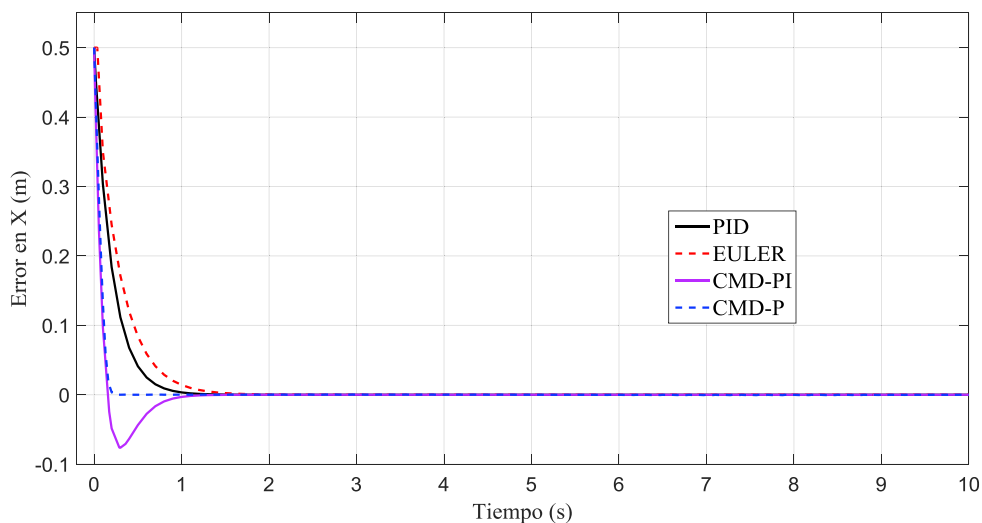


Figura 4.31. Error de trayectoria en el eje X

En la Figura 4.31 se presenta la evolución en el tiempo del error en X, donde se observa que el controlador CMD-PI presenta un sobrepico antes de estabilizarse en un tiempo de 1.2 [s], el controlador CMD-P se estabiliza en 0.3 [s].

Los controladores PID y EULER, alcanzan el estado estable en un tiempo de 1.3 [s] y 1.6 [s] respectivamente.

En las figuras 4.32 y 4.33 se muestra la evolución del error en función del tiempo para los ejes Y y Z respectivamente.

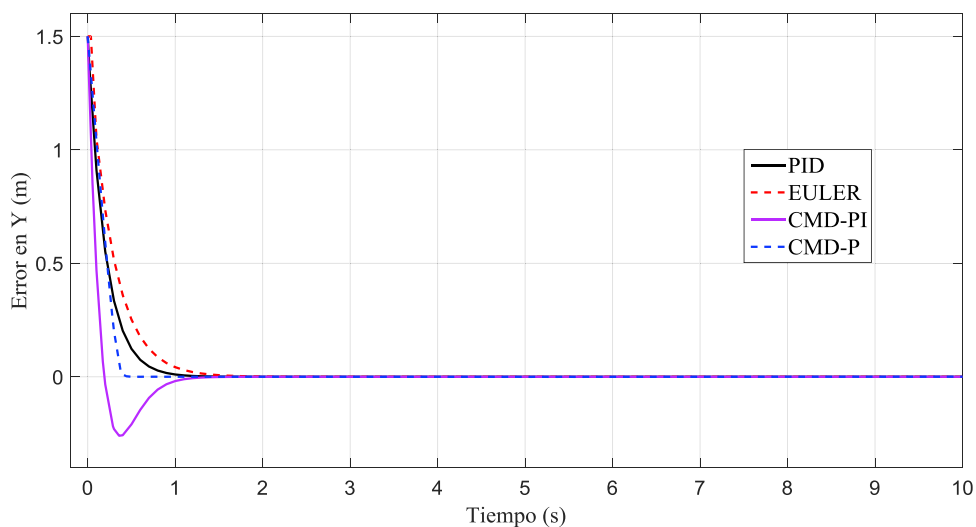


Figura 4.32. Error de trayectoria en el eje Y

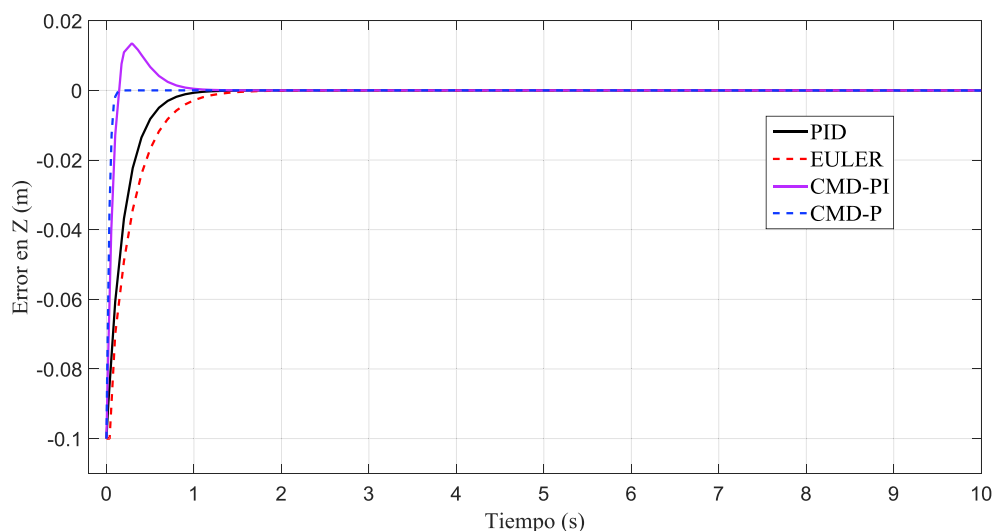


Figura 4.33. Error de trayectoria en el eje Z

Como se observa los errores para los cuatro controladores tienden a cero, lo que significa que el efector final alcanza la trayectoria planteada.

Después de analizar los errores de control, se concluye que los controladores diseñados alcanzan la trayectoria circular, así como las otras tres trayectorias planteadas, mostrando comportamientos similares para cada trayectoria, con ligeras variaciones en tiempos de establecimiento.

4.3.5 RENDIMIENTO DE LOS CONTROLADORES CON EL MODELO CINEMÁTICO

A continuación se analiza el rendimiento de cada controlador para el seguimiento de las trayectorias propuestas, mediante el índice integral del error absoluto (IAE).

Este índice se calcula en base a la integral de la curva del error en función del tiempo, por lo que un valor pequeño del índice significa que hay un error bajo y por tanto un mejor desempeño del controlador.

En la tabla 4.9 se muestra el índice de rendimiento integral del error absoluto (IAE), para cada controlador en cada una de las trayectorias propuestas.

En los resultados obtenidos se observa que el controlador CMD-PI presenta un mejor rendimiento para las trayectorias seno, cuadrado y círculo, mientras que para la

trayectoria en ocho el controlador CMD-P muestra un menor índice, aunque este valor es muy cercano al valor del CMD-PI.

Tabla 4.9 Índice de rendimiento IAE para cada controlador y trayectoria, modelo cinemático.

TRAYECTORIA	CONTROLADOR			
	PID	EULER	CMD-P	CMD-PI
Seno	0.42	0.59	0.44	0.33
Ocho	0.23	0.32	0.13	0.15
Cuadrado	0.90	1.31	1.74	0.86
Círculo	0.32	0.46	0.26	0.23

Se observa además que los valores más altos del índice para todos los controladores se presentan en la trayectoria cuadrada, por lo que esta trayectoria es más difícil de seguir para los controladores debido a los cambios bruscos de orientación que presenta. De manera similar, la trayectoria que presenta menos dificultad de seguimiento para los controladores es la trayectoria en ocho, puesto que presenta los valores más bajos de IAE. Esto se ve reflejado en las curvas de evolución del error en el tiempo presentadas en las secciones anteriores, donde se observan menores tiempos de establecimiento para la trayectoria en ocho y mayores para la trayectoria cuadrada.

Los resultados observados muestran en general un mejor rendimiento del controlador CMD-PI, para saber en qué porcentaje este rendimiento es mayor, se presenta a continuación una comparación entre índices de rendimiento. La forma de obtener esta comparación se expresa en (4.2).

$$\Delta\% = \frac{I_{AEi} - I_{AEj}}{|I_{AEi} + I_{AEj}| \div 2} \times 100 \quad (4.2)$$

Donde se compara el índice de rendimiento IAE de un controlador i con el de un controlador j .

Los resultados de estas comparaciones son presentados en la tabla 4.10.

Tabla 4.10 Comparación porcentual de rendimiento entre CMD-PI y los demás controladores.

CONTROLADOR 1	CONTROLADOR 2	TRAYECTORIA	Δ %
CMD-PI	PID	Seno	23.7%
		Ocho	40.1%
		Cuadrado	5.5%
		Círculo	32.3%
	EULER	Seno	56.7%
		Ocho	71.8%
		Cuadrado	42.2%
		Círculo	65.2%
	CMD-P	Seno	29.2%
		Ocho	-12.9%
		Cuadrado	68.2%
		Círculo	11.5%

Esta comparación permite saber en qué medida un controlador supera a otro en rendimiento, y en base a esto elegir el mejor controlador. Mientras mayor sea el porcentaje de comparación mejor será el comportamiento del controlador.

En los resultados obtenidos se puede observar que el controlador CMD-PI resulta más favorable que los otros controladores en casi todas las trayectorias, con excepción de la trayectoria en ocho, en donde el controlador CDM-P resulta mejor en un 12%, sin embargo comparando los mismos dos controladores para la trayectoria cuadrada el controlador CDM-PI es más favorable en un 68%.

Estos resultados sirven de base para concluir que el controlador más recomendado para el seguimiento de trayectoria del modelo cinemático es el controlador CDM-PI.

4.4 PRUEBA 5: SEGUIMIENTO DE TRAYECTORIAS APLICADO AL MODELO DINÁMICO

En este experimento se aplican los cuatro controladores diseñados, con la finalidad de controlar el modelo dinámico del manipulador móvil, el objetivo de control es que el efector final siga cuatro tipos de trayectorias: i) Seno, ii) Cuadrada, iii) Tipo Ocho y iv) Circular.

Debido a que la dinámica del sistema tiene un comportamiento distinto a la cinemática, se ajustan las constantes de calibración de cada controlador en base al índice de rendimiento IAE, los valores de estas constantes se presentan en la tabla 4.11.

Tabla 4.11 Constantes de calibración para seguimiento de trayectoria, modelo dinámico.

Controlador	Constantes		
PID	$k_x = 40$	$k_y = 40$	$k_z = 40$
EULER	$k_x = 0.6$	$k_y = 0.6$	$k_z = 0.6$
CMD-P	$k_1 = 30$		$\delta = 0.6$
CMD-PI	$k_1 = 60$	$\lambda = 0.1$	$\delta = 0.5$

A continuación se presenta el resultado de la simulación de los controladores diseñados para seguimiento de trayectorias aplicados al modelo dinámico.

4.4.1 SEGUIMIENTO DE TRAYECTORIA SENO

Para esta trayectoria, las coordenadas (x_0, y_0, z_0) de la posición inicial del efector final del manipulador móvil son $(2, -0.5, 0.6)[m]$.

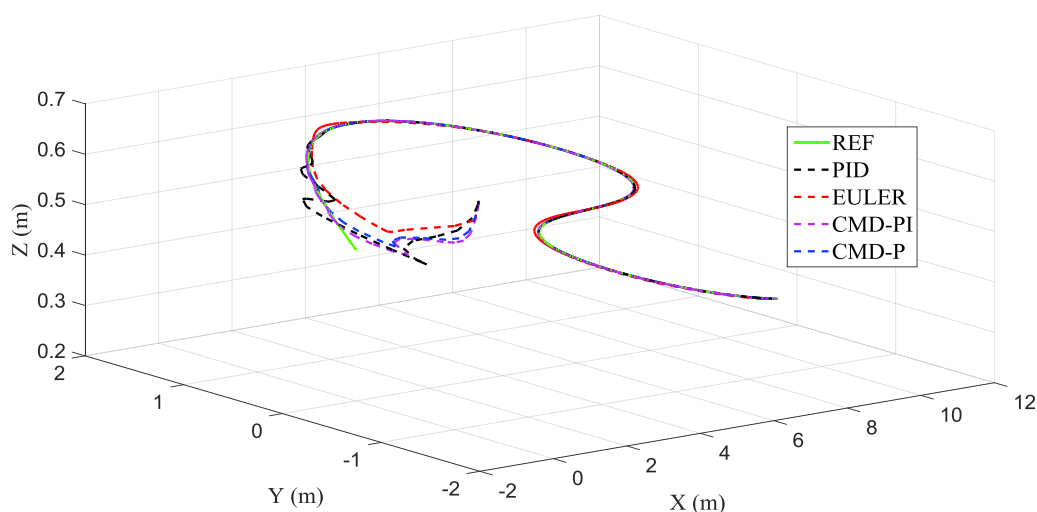


Figura 4.34. Evolución de la trayectoria seno del efector final en 3D

En la Figura 4.34 se observa el seguimiento de trayectoria seno del efector final en tres dimensiones para los cuatro controladores implementados, se verifica que el

efector final alcanza la trayectoria deseada, excepto en el controlador EULER, el cual solo logra su objetivo en ciertas partes de la trayectoria.

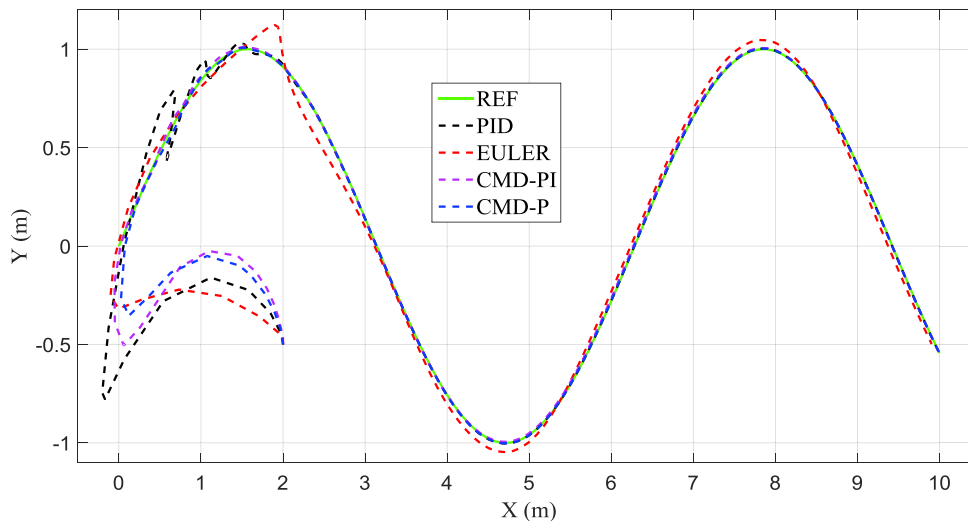


Figura 4.35. Evolución de la trayectoria seno del efector final en el plano XY

En la Figura 4.35 se observa la trayectoria seno del efector final en el plano XY, se verifica que cada controlador alcanza la trayectoria deseada de diferente manera y se mantiene sobre la misma, excepto el controlador EULER.

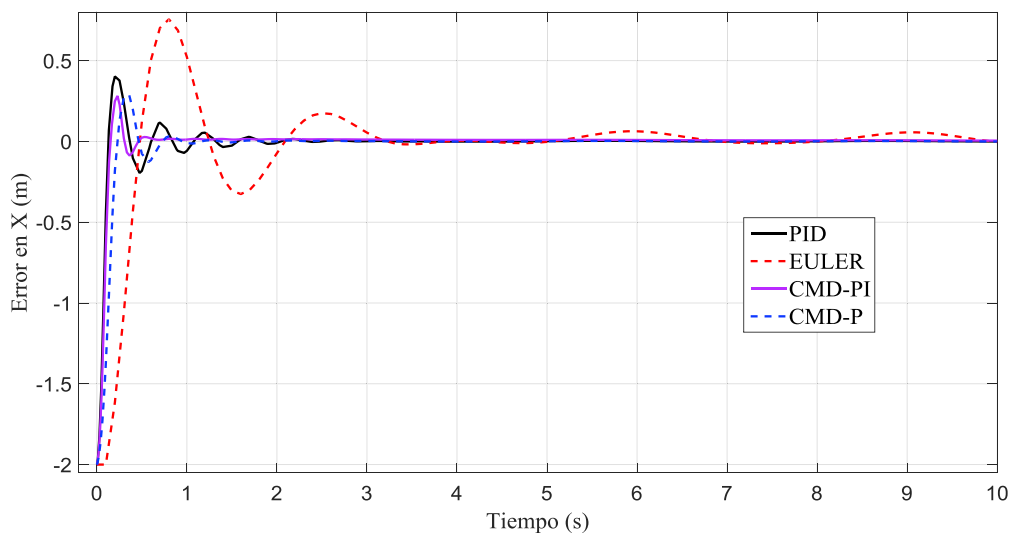


Figura 4.36. Error de trayectoria en el eje X

En la figura 4.36 se observa el comportamiento de los controladores para alcanzar la trayectoria deseada, cada uno de estos responde de diferente manera, se puede

observar que para cada controlador el error converge a cero. El controlador PID se estabiliza en 3.3 [s], el control EULER no cumple el objetivo de control. El control CMD-PI se estabiliza en 1.13 [s], mientras que el control por CMD-P se estabiliza en 1.8 [s].

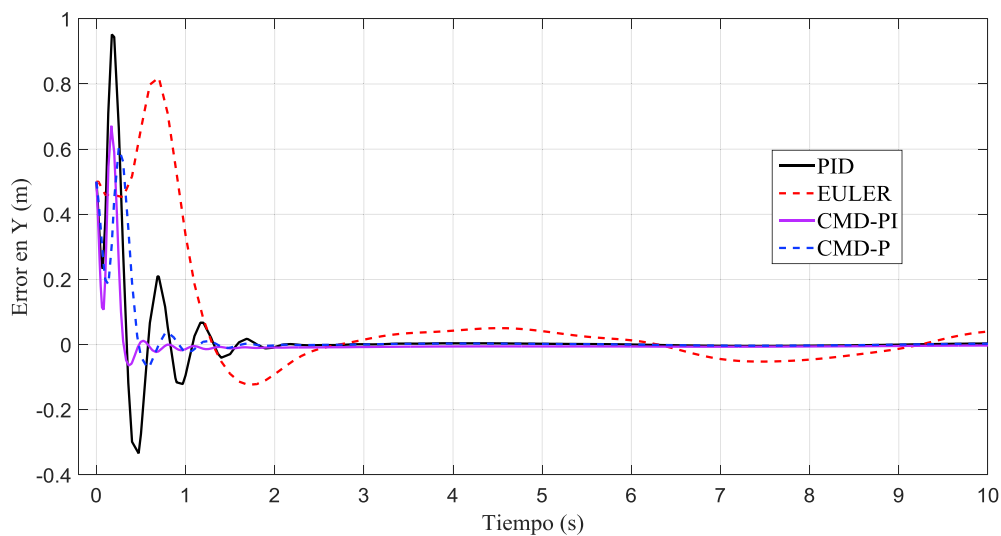


Figura 4.37. Error de trayectoria en el eje Y

En la Figura 4.37 se observa que el control CMD-P se estabiliza en 1.96 [s], superando los otros controladores. En la Figura 4.38 se presenta el error en el eje Z, el cual presenta un mejor tiempo de estabilización en el control CMD-PI de 0.1 [s].

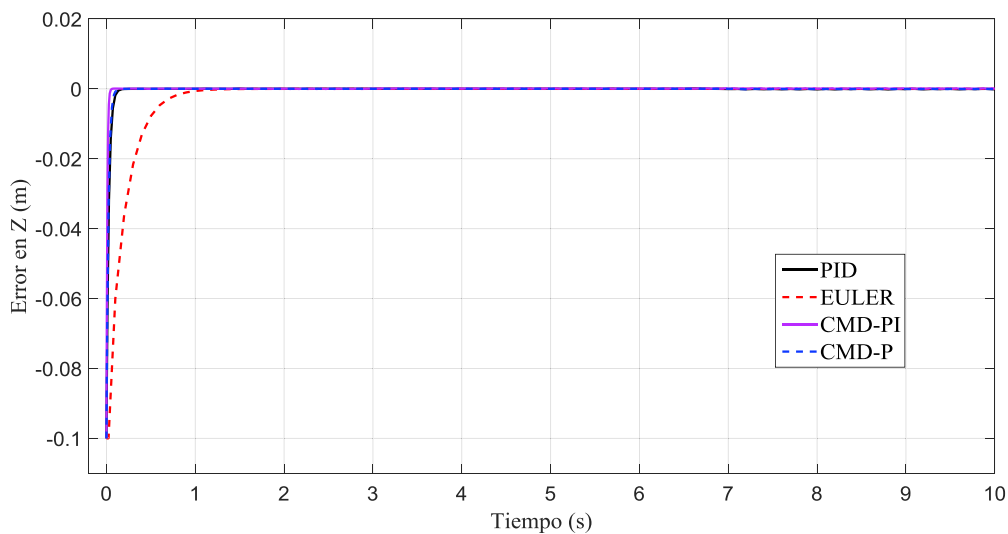


Figura 4.38. Error de trayectoria en el eje Z

Al observar los errores se verifica que el controlador EULER alcanza la trayectoria y se mantiene en ciertos tramos, mientras que en otros sale de la misma, debido a que la dinámica del sistema no está contemplada en el diseño del controlador. Este problema se soluciona en los otros controladores aumentando el valor de las constantes de calibración, pero en el controlador EULER el rango de variación de estas constantes es limitado, por lo que no se puede variar este valor para alcanzar el objetivo de control.

Del análisis de la evolución del error de seguimiento de trayectoria, los resultados obtenidos muestran la robustez de los controladores PID, CMDP y CMD-PI.

Más adelante se presenta el rendimiento de cada controlador mediante el índice IAE.

4.4.2 SEGUIMIENTO DE TRAYECTORIA TIPO OCHO

Para esta trayectoria, las coordenadas (x_0, y_0, z_0) de la posición inicial del efector final del manipulador móvil son $(1, -0.5, 0.6)[m]$.

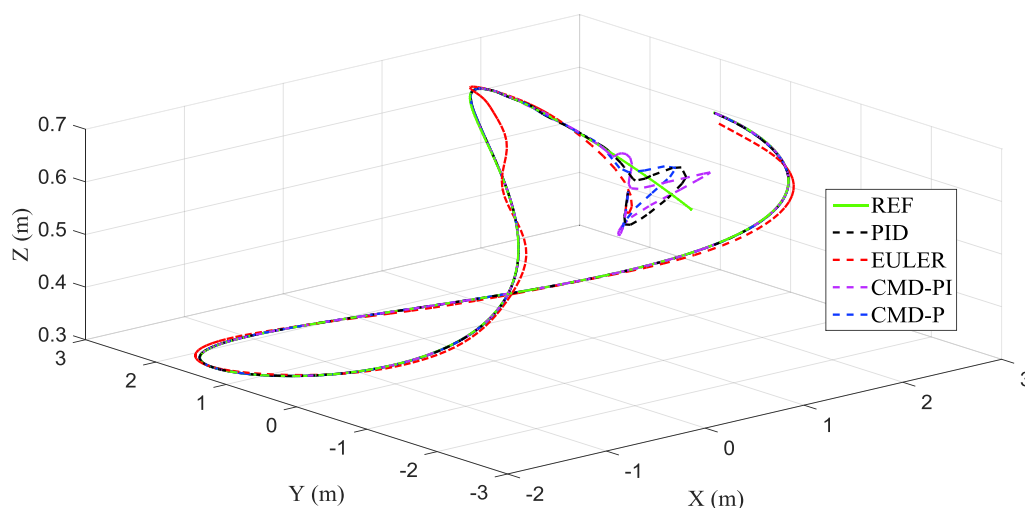


Figura 4.39. Evolución de la trayectoria tipo ocho del efector final en 3D

En la Figura 4.39 se observa el seguimiento de la trayectoria tipo ocho en tres dimensiones para los cuatro controladores, donde se verifica que se cumple el objetivo de control, excepto en el controlador EULER, el cual no alcanza la trayectoria deseada.

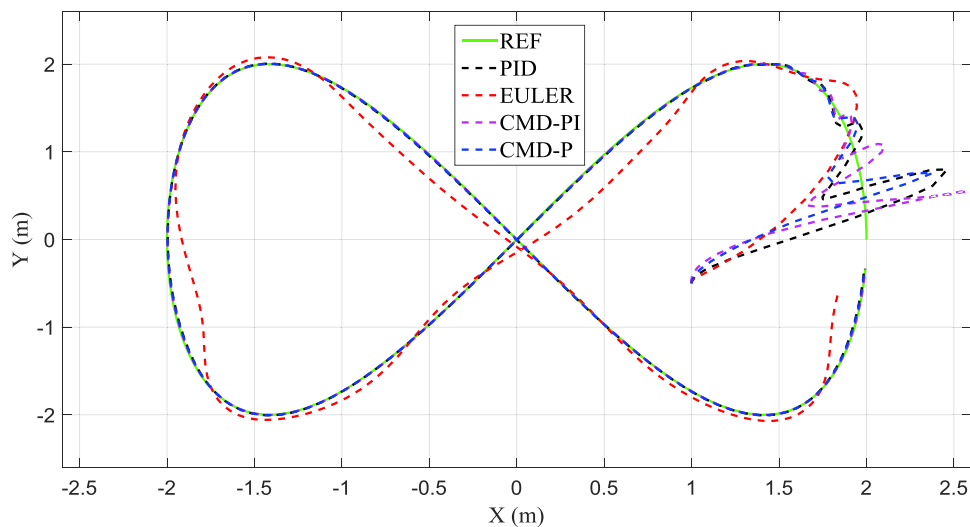


Figura 4.40. Evolución de la trayectoria tipo ocho del efector final en el plano XY

En la Figura 4.40 se observa la trayectoria tipo ocho del efector final en el plano XY, los controladores alcanzan el objetivo de control, salvo el control EULER, el cual presenta dificultad para alcanzar la trayectoria propuesta. En todos los controladores se observa un sobrepico antes de alcanzar la trayectoria.

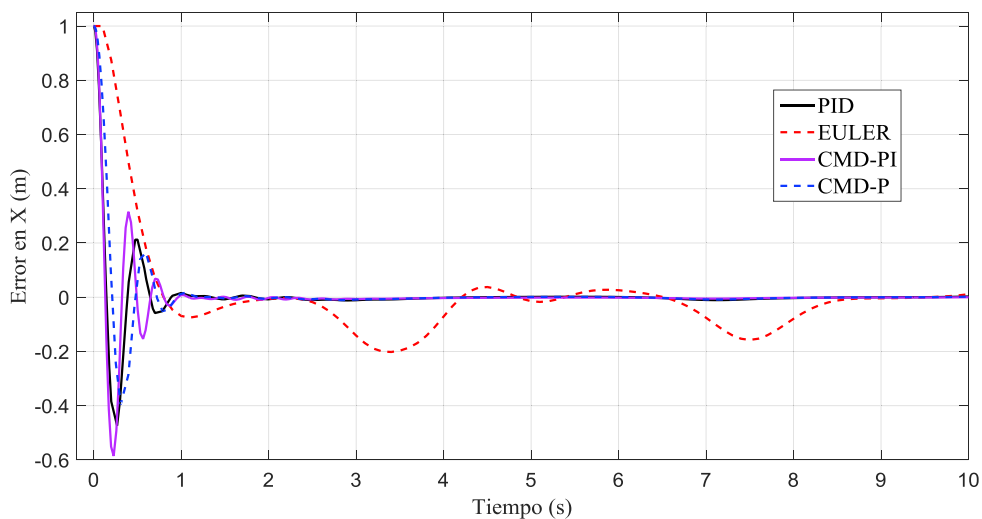


Figura 4.41. Error de trayectoria en el eje X

En la Figura 4.41 se observa la evolución del error en el eje X del efector final para cada controlador. De los resultados obtenidos se tiene que el control CMD-PI se estabiliza en un tiempo de 2.8 [s], el control CMD-P y PID se estabilizan en 2.4 [s] y 3.7 [s] respectivamente. El controlador EULER no alcanza la trayectoria deseada.

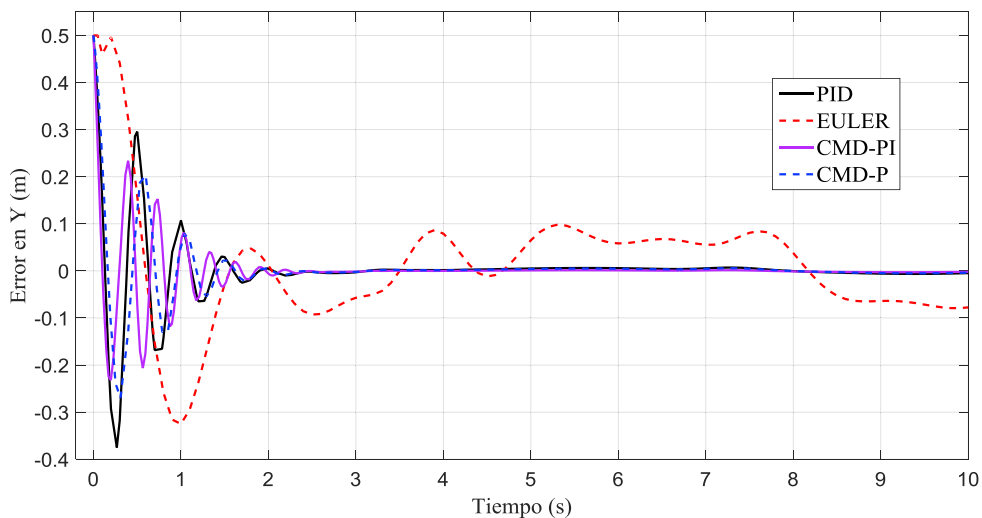


Figura 4.42. Error de trayectoria en el eje Y

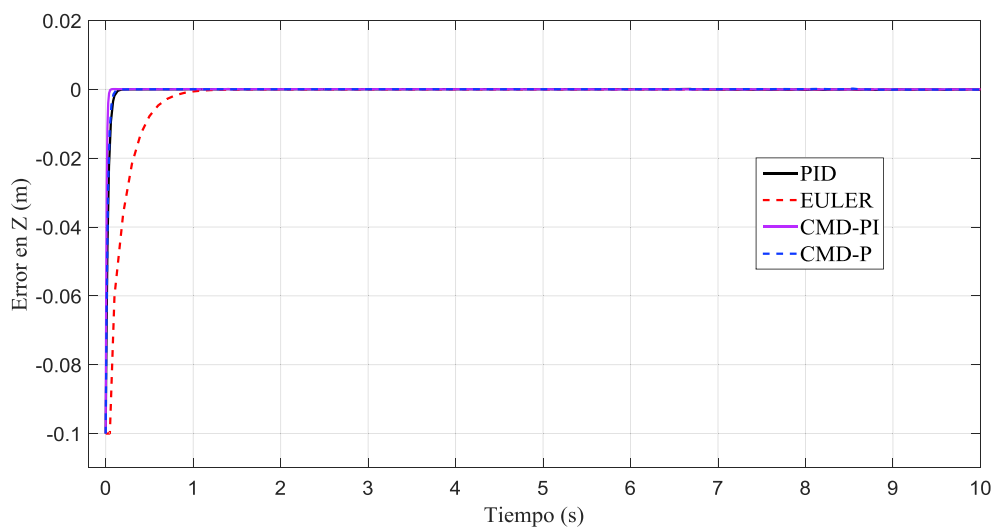


Figura 4.43. Error de trayectoria en el eje Z

En la Figura 4.42 se muestran resultados de la evolución del error en el eje Y, el comportamiento de los controladores implementados es semejante a la evolución del error en el eje X, con un comportamiento oscilatorio.

En los resultados obtenidos para el seguimiento de la trayectoria tipo ocho del efector final con el modelo dinámico, se puede observar la robustez de los controladores PID, CMD-P y CMD-PI, ya que alcanzan el objetivo de control a pesar de emplear un modelo dinámico.

4.4.3 SEGUIMIENTO DE TRAYECTORIA CUADRADA

Para esta trayectoria, las coordenadas (x_0, y_0, z_0) de la posición inicial del efector final del manipulador móvil son $(3, -2, 0.6)[m]$.

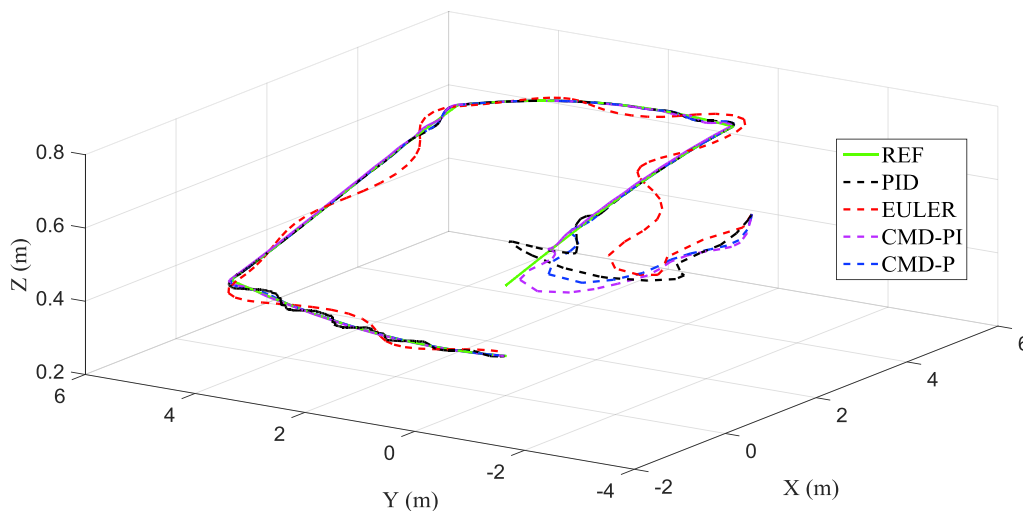


Figura 4.44. Evolución de la trayectoria cuadrada del efector final en 3D

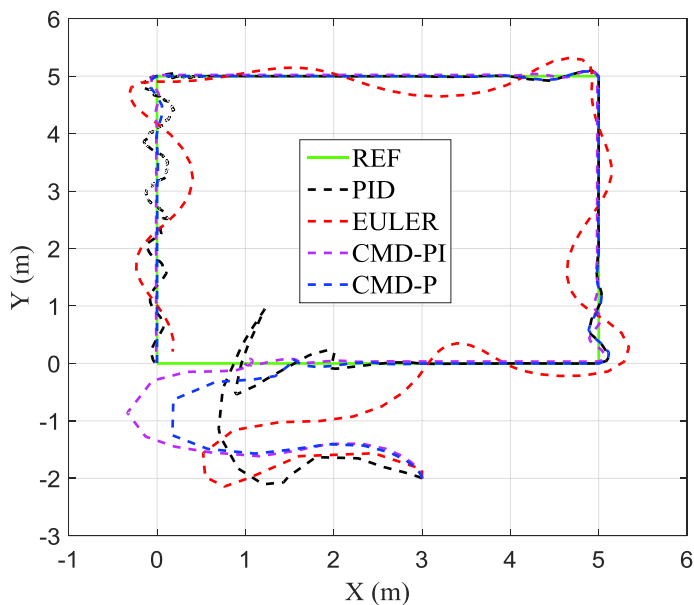


Figura 4.45. Evolución de la trayectoria cuadrada del efector final en el plano XY

En la Figura 4.44 se muestran los resultados del experimento para el seguimiento de trayectoria cuadrada del efector final en tres dimensiones, los resultados muestran que los controladores CMD-P y CMD-PI alcanzan la trayectoria y se mantienen sobre la

misma, el controlador PID alcanza la trayectoria en algunos tramos, mientras el controlador EULER no alcanza la trayectoria.

En la Figura 4.45 se observa el seguimiento de trayectoria en el plano XY, los resultados indican que los controladores que cumplen el objetivo de control son el CMD-PI y CMD-P, mientras que el controlador PID alcanza la trayectoria pero en la tercera esquina se pierde el control. El controlador EULER no alcanza la trayectoria deseada, más adelante se presenta el rendimiento de cada controlador mediante el índice IAE.

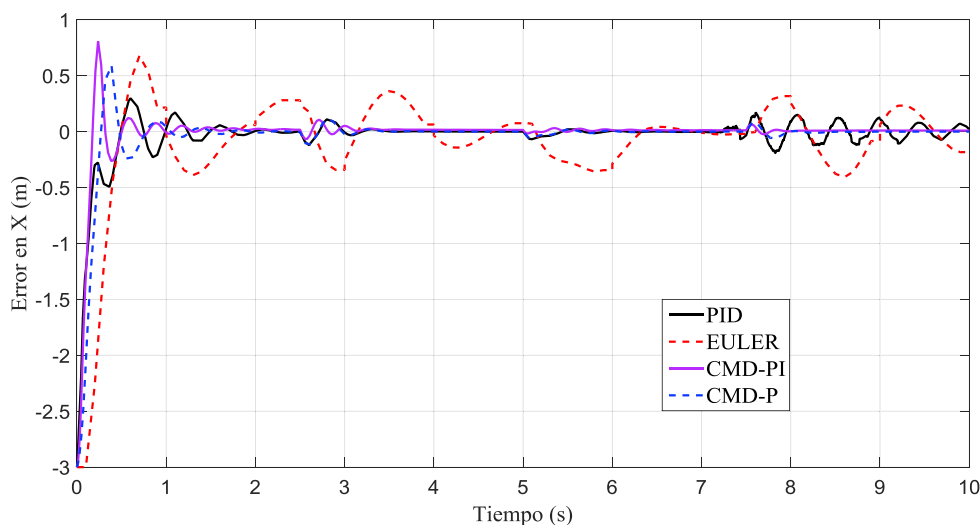


Figura 4.46. Error de trayectoria en el eje X

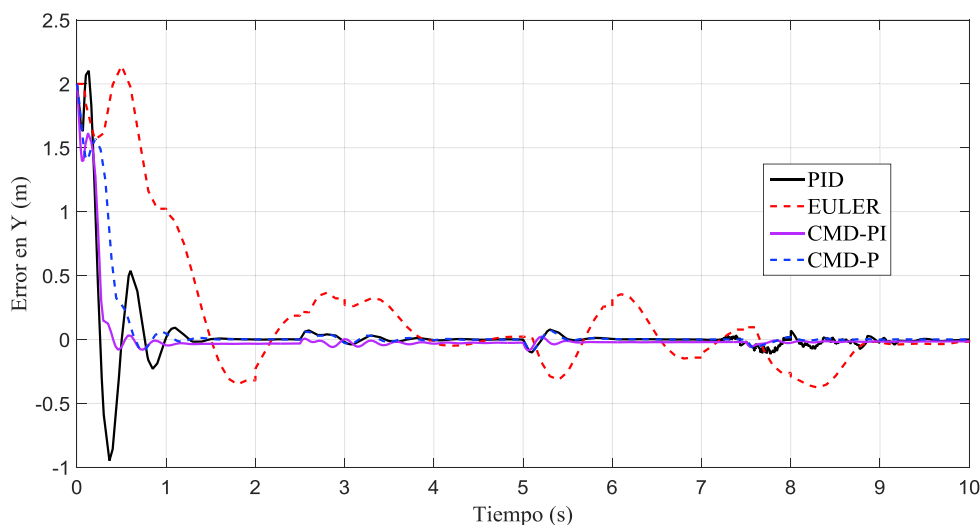


Figura 4.47. Error de trayectoria en el eje Y

En la Figura 4.46 se observa la evolución en el tiempo del error de trayectoria del efector final en el eje X, los controladores que mejor responden son CMD-PI y CMD-P. Los controladores PID y EULER no alcanzan la trayectoria deseada.

En la Figura 4.47 se muestra la evolución del error en Y, la cual es similar a la evolución del error en X, los controladores que más se acercan al error cero son CMD-PI, CMD-P, mientras que los controladores EULER y PID no cumplen con el objetivo de control.

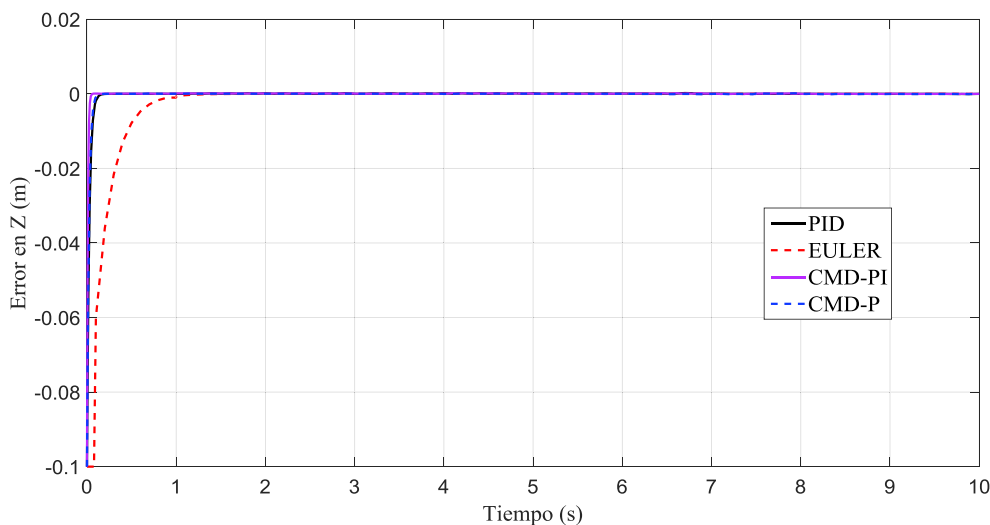


Figura 4.48. Error de trayectoria en el eje Z

En la Figura 4.48 se observa que la evolución del error en Z, donde se observa menos oscilaciones.

Analizando los resultados se puede concluir que se cumple con el seguimiento de trayectorias para cada controlador aplicado, excepto para los controladores EULER y PID que no cumplen con el seguimiento de trayectoria.

4.4.4 SEGUIMIENTO DE TRAYECTORIA CIRCULAR

Para esta trayectoria, las coordenadas (x_0, y_0, z_0) de la posición inicial del efector final del manipulador móvil son $(2.5, 3, 0.6)[m]$.

En la Figura 4.49 se observa el seguimiento de trayectoria circular del efector final en tres dimensiones para los controladores implementados, cada controlador alcanza la

trayectoria deseada, se puede observar que el control por EULER le toma mayor tiempo en alcanzar la trayectoria.

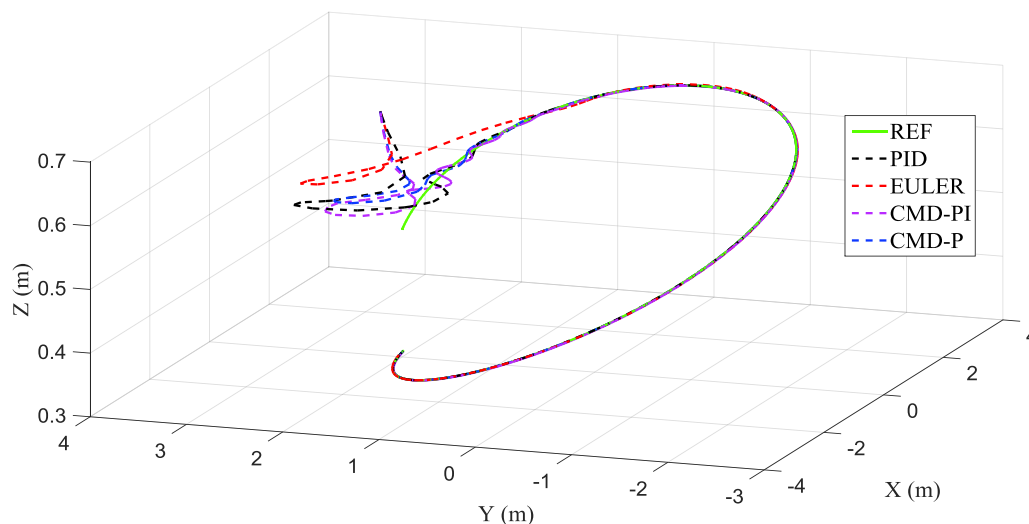


Figura 4.49. Evolución de la trayectoria circular del efector final en 3D

En la Figura 4.50 se observa el seguimiento de trayectoria en el plano XY, en este experimento cada controlador alcanza la trayectoria deseada, se puede observar que antes de estabilizarse los controladores presentan un sobrepico, el controlador que mayor tiempo tarda en alcanzar la trayectoria es el EULER.

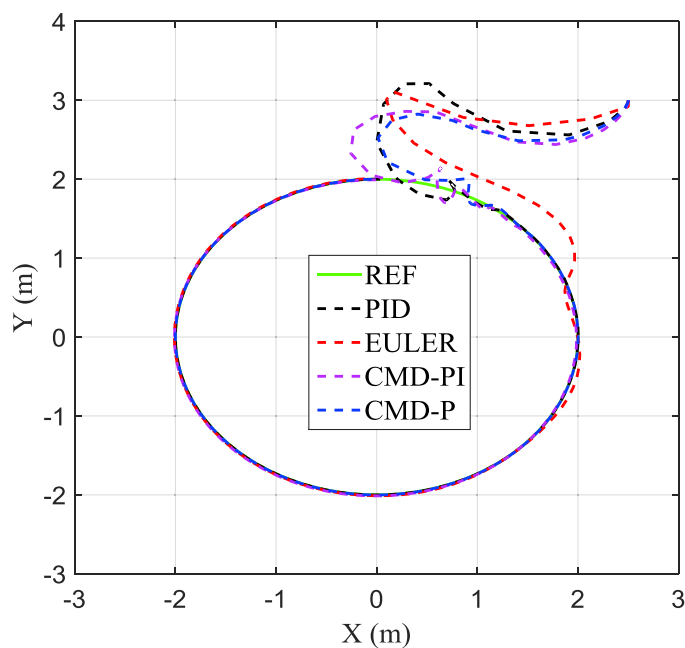


Figura 4.50. Evolución de la trayectoria circular del efector final en el plano XY

En las Figuras 4.51, 4.52 y 4.53 se observa la evolución del error en el tiempo para los controladores.

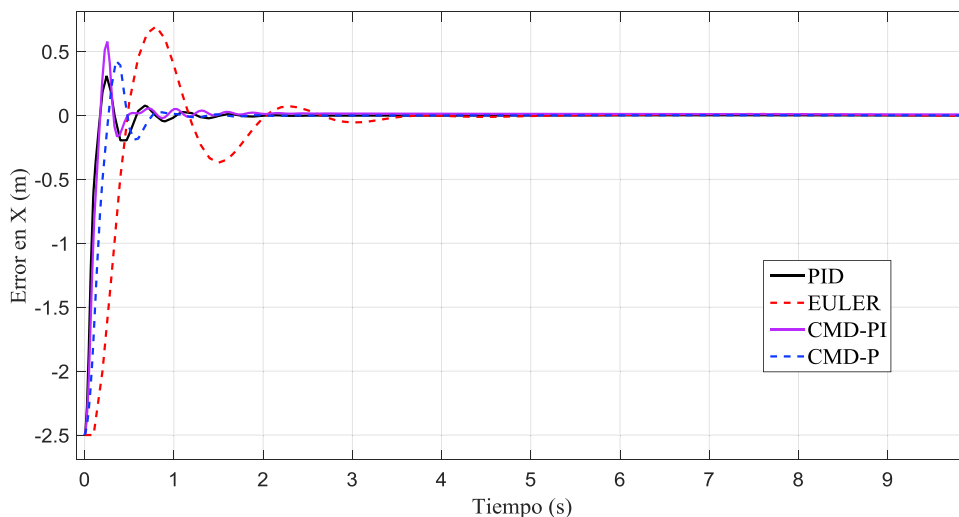


Figura 4.51. Error de trayectoria en el eje X

En la Figura 4.51 se observa la evolución del error en X en el tiempo. El controlador CMD-PI se estabiliza en 2.02 [s] desde este punto el error tiende a cero, para el control CMD-P pasa lo mismo que el anterior control pero el tiempo donde se estabiliza es 1.86 [s], el controlador EULER inicia su tiempo de estabilización en 4.6 [s], el control PID se estabiliza en 2.4 [s], el error en estado estable para este controlador es el que más se acerca a cero.

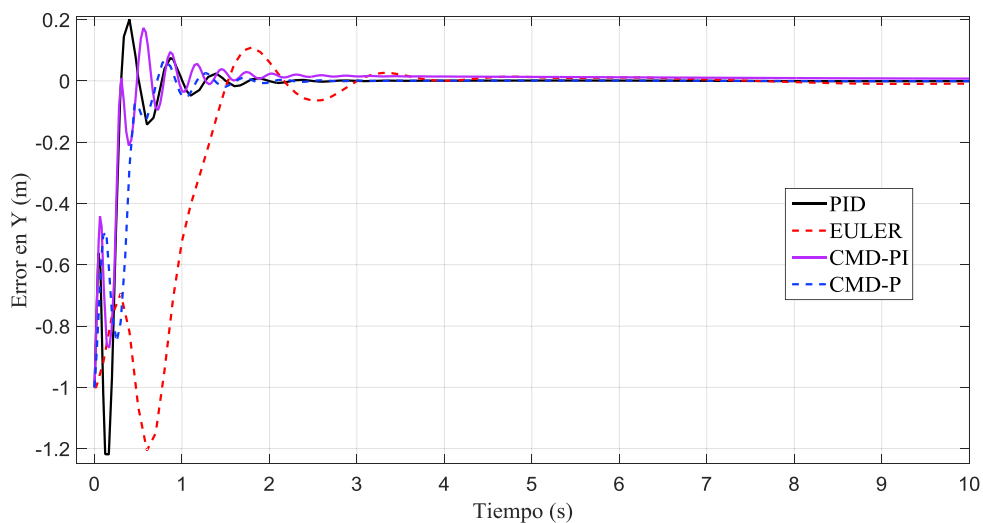


Figura 4.52. Error de trayectoria en el eje Y

En la Figura 4.52 se observa la evolución del error en Y, los controladores presentan un comportamiento similar al error en X. Donde el control PID presenta un mejor acercamiento del error a cero.

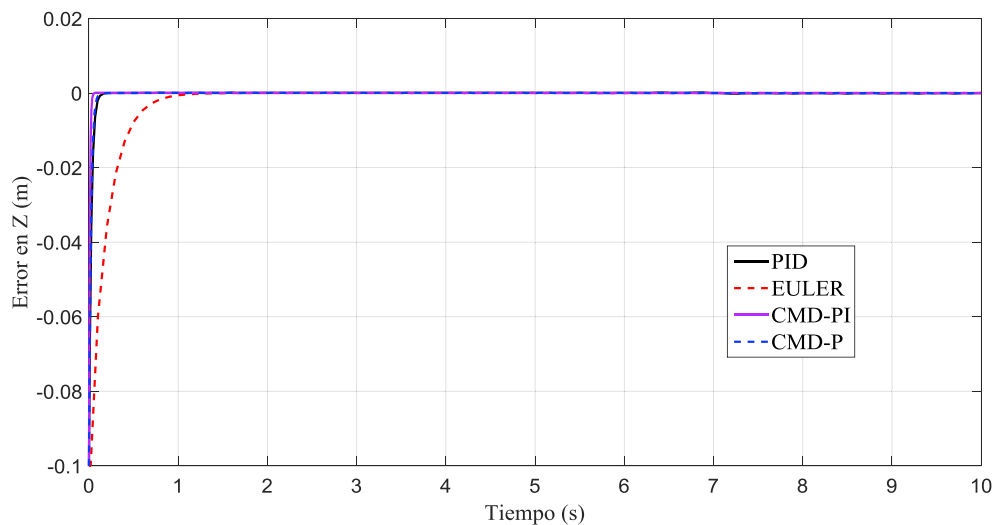


Figura 4.53. Error de trayectoria en el eje Z

En la Figura 4.53 se observa que la evolución del error en el eje Z, observándose menos oscilaciones.

A partir de los resultados observados en las simulaciones, se puede concluir que los controladores PID, CMD-P y CMD-PI, pueden ser ajustados para que puedan controlar al sistema dinámico, lo que demuestran que son controladores robustos y pueden aplicarse al modelo dinámico, a pesar de contemplar solo la parte cinemática en su desarrollo.

4.4.5 RENDIMIENTO DE LOS CONTROLADORES CON EL MODELO DINÁMICO

A continuación se presenta la medida del rendimiento de cada controlador mediante el índice IAE.

En la tabla 4.12 se presenta el índice de rendimiento IAE para cada controlador implementado sobre el modelo dinámico.

Los resultados observados muestran similitud en el comportamiento de los controladores PID, CMD-P y CMD-PI, mientras que el controlador EULER presenta un índice de mayor valor para todas las trayectorias, lo que demuestra el menor rendimiento observado en las gráficas de evolución del error en el tiempo.

Tabla 4.12. Índice de rendimiento IAE para cada controlador y trayectoria, modelo dinámico.

TRAYECTORIA	CONTROLADOR			
	PID	EULER	CMD-P	CMD-PI
Seno	0.45	1.7	0.39	0.34
Ocho	0.38	1.72	0.36	0.31
Cuadrado	1.14	4.11	0.99	0.84
Círculo	0.43	1.78	0.56	0.54

El mejor rendimiento se observa en el controlador CMD-PI para la trayectoria en ocho, mientras que el rendimiento más bajo se presenta en el controlador EULER para la trayectoria cuadrada.

La trayectoria donde los controladores presentan mayor facilidad de seguimiento es la trayectoria en ocho, mientras que la más difícil de seguir es la trayectoria cuadrada, comportamiento similar al observado en el experimento realizado con el modelo cinemático.

En la tabla 4.13, se presenta la comparación entre los controladores en base al rendimiento al rendimiento calculado con el índice IAE. La comparación se realiza de manera similar a la realizada con el modelo cinemático.

De los resultados observados, la utilización del controlador CMD-PI es más favorable que los demás controladores, excepto en la trayectoria circular, donde el rendimiento para el controlador PID resulta mejor y en comparación con CMD-P solo se diferencia en un 3%. La diferencia más notoria se encuentra al realizar la comparación con el controlador EULER, donde todas las comparaciones con el controlador CMD-PI superan el 100%.

Tabla 4.13. Comparación porcentual de rendimiento entre CMD-PI y los demás controladores.

CONTROLADOR 1	CONTROLADOR 2	TRAYECTORIA	Δ %
CMD-PI	PID	Seno	27.8%
		Ocho	19.5%
		Cuadrado	30.7%
		Círculo	-23.8%
	EULER	Seno	133.3%
		Ocho	139.2%
		Cuadrado	132.2%
		Círculo	106.5%
	CMD-P	Seno	15.4%
		Ocho	16.1%
		Cuadrado	16.8%
		Círculo	3.3%

Como se pudo observar en los resultados de los distintos experimentos, el controlador EULER no cumple el objetivo de control al aplicarse al modelo dinámico, esto se debe a que el rango de las ganancias de ajuste para este controlador en tiempo discreto, es menor al de los controladores en tiempo continuo, pudiendo estos últimos tomar valores en teoría desde cero a infinito, mientras que para el controlador EULER solo pueden tener un valor entre cero y uno.

CAPÍTULO 5

CONCLUSIONES Y RECOMENDACIONES

5.1 CONCLUSIONES

- Se estudió los modelos cinemático y dinámico del manipulador móvil de tres grados de libertad y se realizó la simulación en lazo abierto de dichos modelos, por simulación mediante el software Matlab.
- Se estudiaron tres técnicas de control: i) control basado en mínima norma, ii) control basado en álgebra lineal y iii) control por modos deslizantes.
- Se diseñó e implementó cuatro controladores: i) Control PID basado en mínima norma, ii) Control basado en álgebra lineal, iii) Control por modos deslizantes tipo proporcional y iv) Control por modos deslizantes tipo proporcional integral, para control de posición y seguimiento de trayectorias del efector final de un manipulador móvil de tres grados de libertad.
- Los controladores diseñados fueron implementados mediante simulaciones en el programa computacional Matlab, en el cual se verificó que se cumplieron los objetivos de control.
- Se analizó el rendimiento de cada controlador implementado, mediante el índice integral del error absoluto IAE, obteniendo que el control por modos deslizantes presenta mejor rendimiento que las otras técnicas implementadas.
- Mediante pruebas de simulación se comprobó que el rendimiento de los controladores diseñados por modos deslizantes es superior a las otras técnicas implementadas, presentando además mayor robustez ante el cambio del modelo de cinemático a dinámico.

- Para visualizar el comportamiento de cada controlador en un entorno gráfico, se desarrolló una interfaz gráfica utilizando el software de diseño asistido por computador (CAD) Solidworks y las herramientas de realidad virtual de Matlab.

5.2 RECOMENDACIONES

- Se recomienda que las constantes de ajuste de los cuatro controladores sean elegidas en base a un indicador de rendimiento, para encontrar los valores que describan un buen comportamiento del controlador.
- Se recomienda realizar pruebas con perturbaciones e incertidumbres en el modelo, para comprobar la robustez de los sistemas de control diseñados.
- Se recomienda analizar la estabilidad de los controladores mediante el uso de herramientas matemáticas.
- Se recomienda diseñar un controlador dinámico para mejorar las características y robustez del sistema.

REFERENCIAS BIBLIOGRÁFICAS

- [1] A. Barrientos, L. Peñín, C. Balaguer y R. Aracil, "Fundamentos de Robótica," 2da ed., Madrid: McGraw-Hill, 2007.
- [2] M. Pérez, E. Cuevas y D. Zaldívar, "Fundamentos de Robótica y Mecatrónica con Matlab Y Simulink," 3ra ed., México D.F.: Alfaomega, 2015.
- [3] A. Ollero, "Robótica, manipuladores y robots móviles", Barcelona: Marcombo, 2001.
- [4] International Federation of Robotics IFR, "Service Robots," [Online]. Available: <http://www.ifr.org/service-robots>.
- [5] J. González y A. Ollero, "Estimación de la Posición de un Robot Móvil", Dpto. de Ingeniería de Sistemas y Automática. Universidad de Málaga, 1996, pp. 1-25.
- [6] G. Andaluz, "Modelación, Identificación y Control de robots móviles", Quito: Escuela Politécnica Nacional, 2011.
- [7] G. Oriolo, "Control of Nonholonomic Systems," Universidad de Roma "La Sapienza", 2005. [Online]. Available: http://www.dis.uniroma1.it/~oriolo/cns/cns_slides.pdf.
- [8] Blog de Robótica móvil, "Robots Móviles," [Online]. Available: <http://roboticamoviles.blogspot.com>.
- [9] F. Reyes, "Control de robots manipuladores", México: Alfaomega, 2011.
- [10] C. Robotics, "User Guide Mover4 Version 2014/06 (SW V902-06-014)," 2014. [Online]. Available: www.cpr-robots.com/ressourcen/UserGuide_Mover4.pdf.

- [11] Robotnik, "Robotnik," [Online]. Available: <http://www.robotnik.es/>.
- [12] G. Salazar, J. Álvarez y M. Moreno, "Una metodología integral para el modelado cinemático de manipuladores móviles con ruedas", Instituto Politécnico Nacional, 2009, pp. 1-6.
- [13] M. Chen y A. Zalzalá, "A Genetic Approach to the Motion Planning of Redundant Mobile Manipulator Systems Considering Safety and Configuration", Journal of Robotic Systems, 1995, pp. 1-16.
- [14] B. Siciliano, "Kinematic Control of Redundant Robot Manipulators", Journal of intelligent and Robotic Systems, 1990, pp. 1-12.
- [15] H. Carvajal, "Modelamiento y Diseño de Robots Industriales", Bogotá: UNISALLE, 2014.
- [16] S. Kumar, "Introducción a la Robótica," 2da ed., México D.F: McGraw-Hill, 2010.
- [17] S. Lipschutz, "Álgebra lineal", Madrid: Mc Graw Hill, 1992.
- [18] R. López, "Matrices, Apuntes de clase," Universidad de Granada, 2003. [Online]. Available: <http://www.ugr.es/~rcamino/docencia/geo1-03/g1tema3.pdf>.
- [19] C. López, "Cátedra de Matemática y Elementos de Matemática," Universidad Nacional de la Plata, 2009. [Online]. Available: <http://www.fcnym.unlp.edu.ar/catedras/matematica/teoricos/UT4.pdf>.
- [20] B. Kolman y D. Hill, "Álgebra Lineal," 7ma ed., México: PEARSON EDUCACIÓN, 2008.
- [21] R. Larson, "Fundamentos de algebra lineal," 6ta ed., México D.F.: Cengage Learning Editores, 2015.

- [22] W. Boyce y R. DiPrima, "Ecuaciones Diferenciales y Problemas con Valores en la Frontera," 2da ed., México: Limusa Wiley, 2013.
- [23] L. Capito y P. Proaño, "Seguimiento de Trayectorias mediante cuatro Técnicas de Control Utilizando una Plataforma Robótica Pioneer 3DX y el Sistema Operativo Robótico ROS", Quito: Escuela Politécnica Nacional, 2015.
- [24] R. Kelly y V. Santibáñez, "Control de Movimiento de Robots Manipuladores," 2da ed., Madrid: PEARSON Prentice Hall, 2003.
- [25] A. Holey, "Los Robots de Nissan crean hasta 113 carros por hora," CNET, 16 Agosto 2014. [Online]. Available: <http://www.cnet.com/es/imagenes/robots-nissan-crean-hasta-113-carros-por-hora-fotos/?query=robots+nissan>.
- [26] M. Seron and J. Braslavsky, "Sistemas No Lineales, notas de clase," Universidad Nacional de Quilmes, Septiembre 2000. [Online]. Available: <http://www.fceia.unr.edu.ar/control/snl/Apunte.pdf>.
- [27] J. Salt, Á. Cuenca, V. Casanova y A. Correcher, "Control Automático Tiempo Continuo y Tiempo Discreto", Barcelona: REVERTÉ, 2015.
- [28] A. Rosales, G. Scaglia, V. Mut y F. Di Sciascio, "Control Dinámico Mediante Métodos Numéricos Para Robots Móviles Tipo Uniciclo", Universidad Nacional de San Juan, Argentina, 2007, pp. 1-7.
- [29] O. Camacho y C. Smith, "Sliding Mode control: an approach to regulate nonlinear chemical process", Elsevier Science, 2000, pp. 1-14.
- [30] W. García y Z. Darine, "Control predictivo por modo deslizante para robot manipuladores", vol. 11, Revista Ingeniería UC, 2004, pp. 39-47.
- [31] F. Naranjo, "Control Lineal Moderno, Análisis y Diseño en el Espacio de Estados", Cali: Universidad Autónoma de Occidente, 2012, pp. 169-170.


- [32] E. Muñoz y C. Gaviria, "Control PID Multivariable y Modos Deslizantes de un Robot SCARA", Universidad del Cauca, 2011, pp. 1-6.
- [33] D. Soetano, L. Lapierre y A. Pascoal, "Adaptive Non singular Pathfollowing Control of Dynamic Wheeled Robots", Decision and Control, 2003. Proceedings. 42nd IEEE Conference on , 2003.
- [34] R. Maroto, "Ecuaciones para la sintonización de controladores PID con acción derivativa aplicada a la señal realimentada", Ciudad Universitaria Rodrigo Facio: Universidad de Costa Rica, 2007.
- [35] P. Leica, "Apuntes de Sistemas de Control Automático", Quito: Escuela Politécnica Nacional, 2016.
- [36] F. Ornelas, "Análisis de Sistemas No Lineales", México: Universidad Michoacana de San Nicolás de Hidalgo, 2010.
- [37] A. Hoyo, "Simulación de robots con Matlab y Simulink en escenarios virtuales 3D", I Congreso Iberoamericano de Enseñanza de la Ingeniería, 2008.
- [38] A. Roldán y S. Rodríguez, "Implementación de realidad virtual en robótica móvil, mediante Matlab", 15vo congreso nacional de ingeniería electromecánica y de sistemas, 2015, pp. 1-6.
- [39] SolidWorks Corporation, "SOLIDWORKS," Dassault Systemes, [Online]. Available: <http://www.solidworks.es/>.
- [40] Mobilerobots, "Pioneer 3-DX Datasheet," Adept Technology, [Online]. Available: <http://www.mobilerobots.com/ResearchRobots/PioneerP3DX.aspx>.
- [41] F. Nascimento, R. Carelli, M. Sarcinelli y T. Freire, "Modelo Dinámico y Controlador de Seguimiento para Robots Móviles Tipo Uniciclo", V Jornadas Argentinas de Robótica (JAR'08), 2008, pp. 1-6.

ANEXOS

ANEXO A

HOJA DE CARACTERÍSTICAS DE LAS PLATAFORMAS

Las Figuras A.1 y A.2 muestran las características del robot móvil Pioneer 3-DX, utilizado como base del manipulador móvil implementado.



Pioneer 3-DX

Pioneer 3-DX is a small lightweight two-wheel two-motor differential drive robot ideal for indoor laboratory or classroom use. The robot comes complete with front SONAR, one battery, wheel encoders, a microcontroller with ARCOS firmware, and the Pioneer SDK advanced mobile robotics software development package.

Pioneer research robots are the world's most popular intelligent mobile robots for education and research. Their versatility, reliability and durability have made them the preferred platform for advanced intelligent robotics. Pioneers are pre-assembled, customizable, upgradeable, and rugged enough to last through years of laboratory and classroom use.

Product Features and Benefits

- **Easy to Use** - Comes assembled and integrated with its accessory package.
- **Reliable** - Construction is durable and rugged. Easily handles the small gaps, minor bumping, jarring, or other obstacles that hinder other robotic platforms. Some Pioneer robots have been in service for over 15 years.
- **Pioneer Software Development Kit** - All Adept MobileRobot platforms include Pioneer SDK, a complete set of robotics applications and libraries that accelerate the development of robotics projects. Pioneer SDK is backed by our product support team.
- **Customizable** - Easily accessible by choosing from dozens of supported and tested accessories that integrate with the robotic platform. Additional help is available for future upgrades or added accessories.
- **Reference Platform** - Pioneer robots are a standard in intelligent mobile platforms. Search your preferred robotics journal or conference listings to find many examples of Pioneer platforms in research applications.
- **Technical Support** - Pioneer software and hardware comes fully documented with additional help available through our product support team.

adept
mobilerobots

Specifications

Construction

Body: 7.8 mm aluminum (pioneer-3) robot
Tires: Foam-filled rubber

Operation

Robot Weight: 9 kg
Clearing Height: 12 cm

Differential Drive Movement

Two Motors: 2 x 1
Driving Radius: 26.7 cm
Max. Forward/Backward Speed: 1.2 m/s
Rotation Speed: 300°/s
Max. Traversable Step: 3.3 cm
Max. Traversable Gap: 2 cm
Max. Traversable Obstacle: 23%
Traversable Terrain: Indoor, outdoor, uneven

Batteries

Run Time: 8-10 hours w/1 battery (not recommended)
Charge Time: 12 hours (standard) or 3.4 hrs (optional high-capacity charged)
Available Power Supplies:
5 V @ 1.2 A included
12 V @ 2.2 A included

Batteries

Supports up to 2 at a time
Voltage: 1.2V
Capacity: 7.2 Ah (max)
Chemistry: Lead acid
Not rechargeable Batteries: Yes

Available Backups Options:

- Smart plug-in
- Charging station
- Powercube (2-battery charging bag)

* Batteries are accessible through front-mounted access panel for fast swapping (continuous operation)

Microcontroller: V/D

- System board
- 32 digital inputs
- 8 digital outputs
- 7 analog inputs
- 2 serial expansion ports

* Some parts may not be available if certain accessories are included with the robot

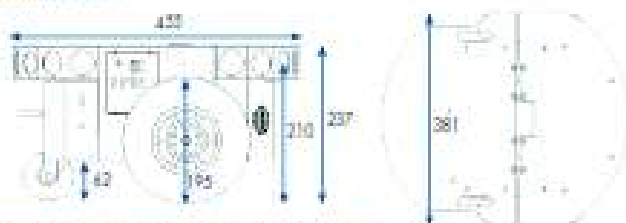
User Control Panel:

- RGB programmable panel keypad
- Motor status indicator
- Battery charge indicator
- 2 A/D sensor controllers
- System reset
- Motor enable pushbutton

Figura A.1 Hoja de Características Robot Pioneer 3-DX, parte 1

Pioneer 3-DX

Dimensions (mm)



Core Software - included with all research platforms

ARIA provides a framework for controlling and receiving data from all MobileRobot platforms, as well as most accessories. Includes open source infrastructure and utilities useful for writing robot control software, support for network sockets, and an extensible framework for client-server network programming.

MobileSim open source simulator which includes all MobileRobot platforms and many accessories.

MobileEye graphical user interface client for remote operation and monitoring of the robot.

Mapper 3-Basic tool for creating and editing map files for use with ARIA, MobileSim, and navigation software.

SONARlib provides sensor-based approximate localization and navigation.

Accessory Support Software - bundled with purchase of robotic accessory

ARML enables robot, laser-based autonomous localization and navigation.

Robotic Arm Support Pioneer arms are packaged with integrated software support.

Speech Recognition and Synthesis Library: Easy-to-use C++ development library for speech recognition based on the open source Sphinx2 system. Speech synthesis (text-to-speech) based on Cepstral synthesis.

ACTS Color Tracking System: Software application which reads images from a camera and tracks the positions and sizes of multiple color regions. Information can be incorporated into your own software via ARIA.

Optional Industrial Grade Internally Mounted Computers

Motorola 68K-37 Dual Core 3.36 MHz - 3-8 GB RAM
6 X USB 2.0 Ports
2 X PC/104+ Slots
4 X RS-232 Serial Ports
2 X Hi/100/1000 Ethernet Ports
Onboard Audio & Video
Solid State Drive
Optional Wireless Ethernet

Optional Accessories:

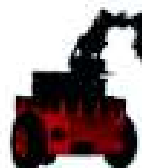
- Laser range finders
- Mono- and stereo-vision cameras
- Real SONAR
- Wireless serial to Ethernet for remote operation
- Robotic arms and grippers
- Gyroscope
- Segmented bumper arrays
- Speakers and microphones
- Joystick
- Many more...

Include our integrated & supported accessories with your Pioneer 3-DX.

Here are some popular configurations to choose from:



Mapping & Vision



Gripping & Manipulation



Audio & Speech

More information:

See our website: www.mobilerobots.com for a full range of supported accessories or contact our sales department to discuss your application.

adept

Adept Technology, Inc. 10 Columbia Drive, Amherst, NH 03003
Tel: 603-881-7760 Email: sales@mobilerobots.com

www.mobilerobots.com

Specifications subject to change without notice.
©2011 Adept Technology, Inc. ALL RIGHTS RESERVED. The information provided in this communication is deemed to be the property of Adept Technology, Inc. and is protected by copyright and other intellectual property laws. In addition, any reference to any Adept Technology, Inc. products are the marks and property of Adept Technology, Inc. (and may be registered trademarks), all other trademarks or trademarks are the property of their respective holders.

01000-0101 Rev. 0

Figura A.2 Hoja de características del Robot Pioneer 3-DX, parte 2

Las Figuras A.3 y A.4 muestran las características del robot Arm Mover, brazo manipulador implementado en la simulación del manipulador móvil.





Specifications

Construction: Serial four servo joints with gear motor, encoder and µC per joint

Weight / Payload: 3.5 kg / 500 g

Reach: 550 mm incl. gripper

Communication: USB, internally CAN at 20 Hz

Power Supply: 12V via mains adapter, <60W

Digital Inputs: 4 x 5V or 3 x 24V

Digital Outputs Base: 3 x 5V or 4 relay outputs

Digital Outputs Flange: 2 digital Outputs, 5V, +12V and GND for Gripper

Software: CPRog (Windows), ROS (Linux)

The Mover4 is not suited to be used in industrial production or continuous operation use cases.

Control Software

The 3D programming environment CPRog allows to move and program the robot in an intuitive way. Graphical and textual program editors allow creating or adapting programs with ease. All motions can be simulated. Alternatively ROS packages for Willow Garages Robot Operating System are available.

Versions

The mover can be equipped with different grippers, a 24V IO extension to communicate with PLCs, and a transport box. The necessary control PC or laptop is not included.

→→→ www.cpr-robots.com

**Mover4Axis
Robot Arm Mover**

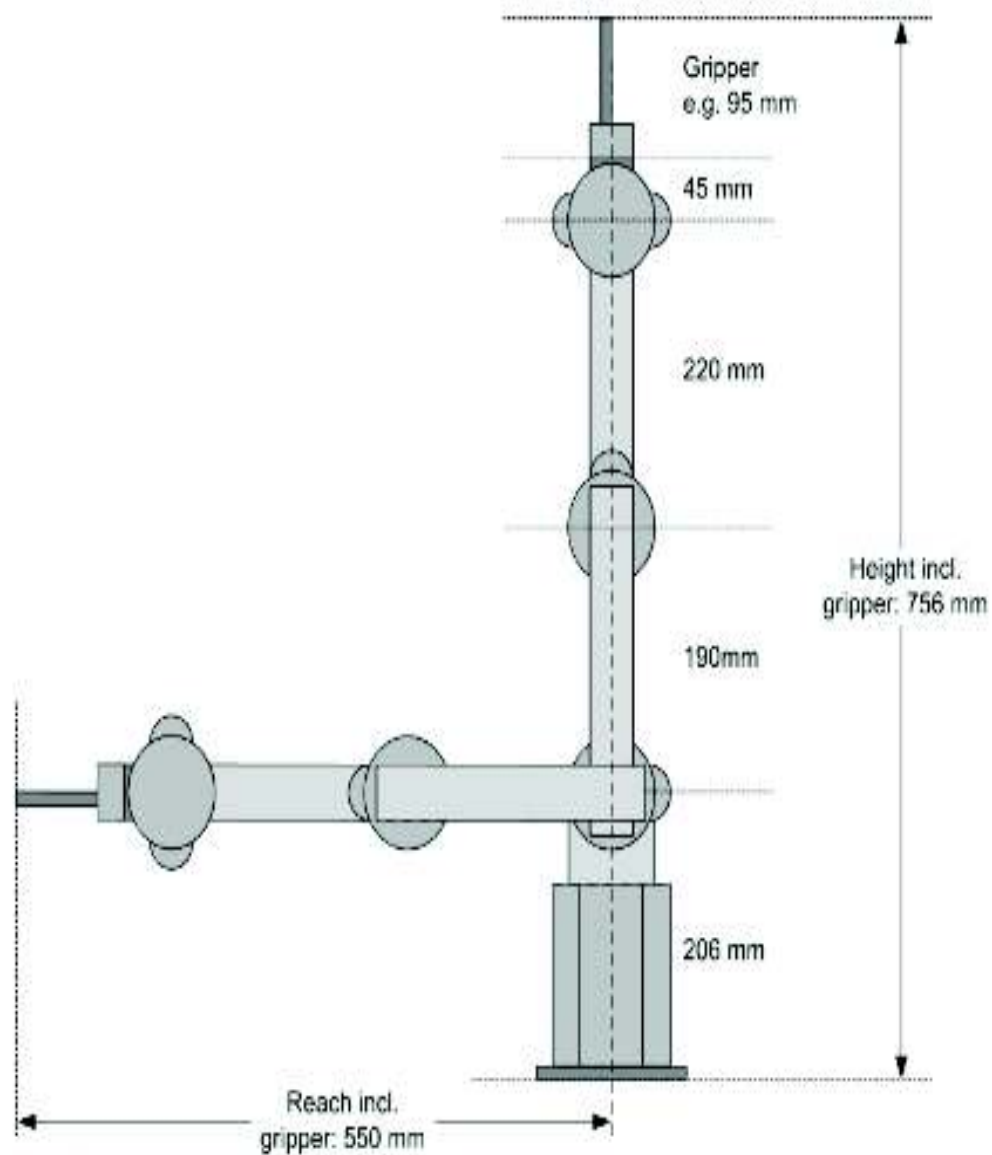
The Commonplace Robotics Mover is a robot arm that has been designed especially to meet the needs of schools, universities and R&D institutions. He offers best functionality with a very high value for money.

Commonplace Robotics GmbH
Osterfeldstraße 1
49326 Melle, Germany

www.cpr-robots.com
info@cpr-robots.com
+49 (0)5429 / 374 983 4

Product Sheet Mover4Axis
Version 0510-06
© Commonplace Robotics GmbH 2014

Figura A.3 Hoja de características del Robot Arm Mover, parte 1



Pic. 3: Mover4 simplified side view

Figura A.4 Hoja de características del Robot Arm Mover, parte 2

ANEXO B

SOLIDWORKS

Es un software de diseño mecánico, que permite crear modelos en 2D y 3D, asociar estos modelos y colocar restricciones en la construcción. La interfaz de SOLIDWORKS es muy amigable con el usuario, el desarrollo de cualquier objeto se lo realiza de una forma más simple y en menor tiempo.



Figura B.1 Pantalla Principal de SOLIDWORKS

Este software contiene herramientas de trabajo empleadas para la construcción o diseño de una pieza, para empezar a trabajar en este entorno se hace clic en archivo y se despliega una lista como muestra la Figura B.2, se hace clic en “nuevo” y se selecciona una de las siguientes opciones:

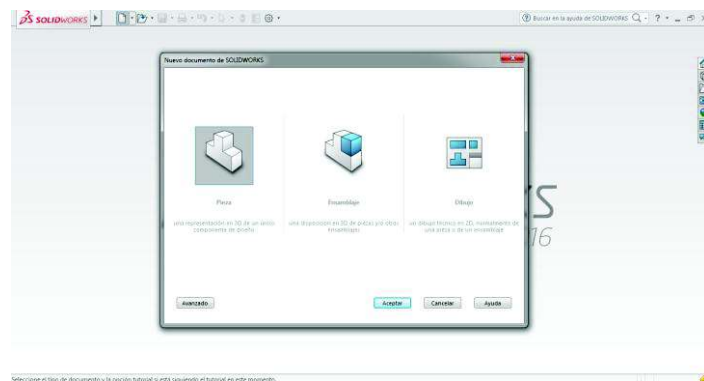


Figura B.2 Ventana de selección del tipo de diseño a realizar

- **PIEZA**, se usa para el diseño de figuras o piezas en 3D.

- ENSAMBLAJE, se usa para unir en una sola pieza diseños trabajados individualmente.
- DIBUJO, permite visualizar las vistas ortogonales de una pieza o ensamble en 2D.

En el diseño del Manipulador móvil se utiliza la opción de PIEZA para construir de manera individual cada una de las piezas de este, se usa la opción de ENSAMBLAJE para unir cada una de las piezas y así obtener el diseño total del manipulador móvil.

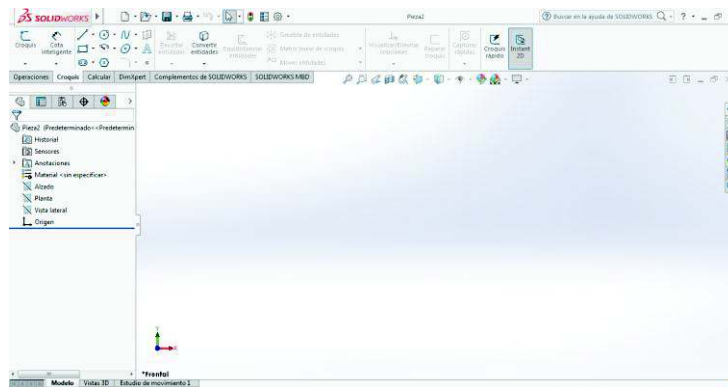


Figura B.3 Interfaz de trabajo

En la figura B.4 se puede apreciar la barra de herramientas que se encuentra en la parte superior izquierda, se encuentra varias opciones que se utilizarán al momento de realizar el diseño, se encuentra las opciones de GUARDAR, ARCHIVO, IMPRIMIR, entre otros.



Figura B.4 Barra de Herramientas

En la Figura B.5 Se presenta las operaciones que se van a realizar para obtener la figura deseada, en la opción Croquis es donde se establece el plano de trabajo y se coloca las características de la figura como cotas, unidades. La opción operaciones permite Extruir la figura lo cual significa dar una forma tridimensional de la pieza diseñada, de la misma forma permite realizar una Revolución de saliente/base, lo cual es una proyección de forma tridimensional del dibujo.



Figura B.5 Menú Opciones

Se hace una breve explicación del diseño de una pieza en Solidworks en la Figura B.6 El primer paso es seleccionar en la Interfaz la opción Archivo, nuevo, luego elegir la opción PIEZA. Se selecciona el plano en que se va a trabajar y el croquis en el cual permite realizar cualquier tipo de figura. Una vez seleccionada la figura se aplica las propiedades como Cota Inteligente que permite dar las medidas a la pieza.

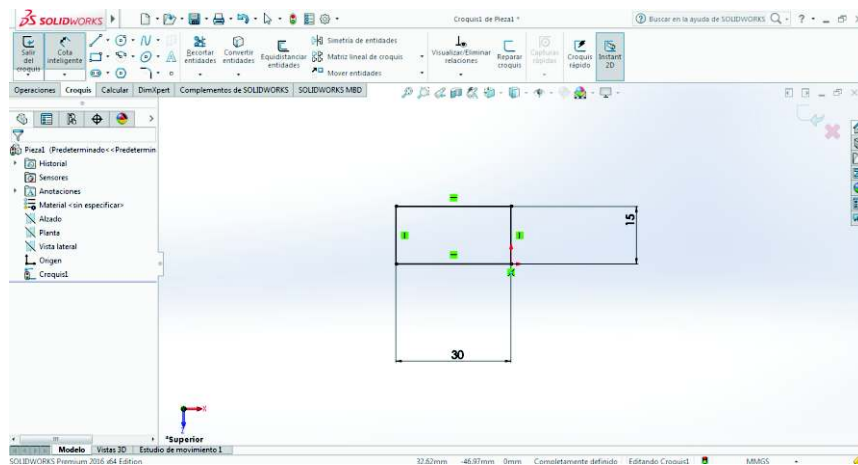


Figura B.6 Diseño de Pieza

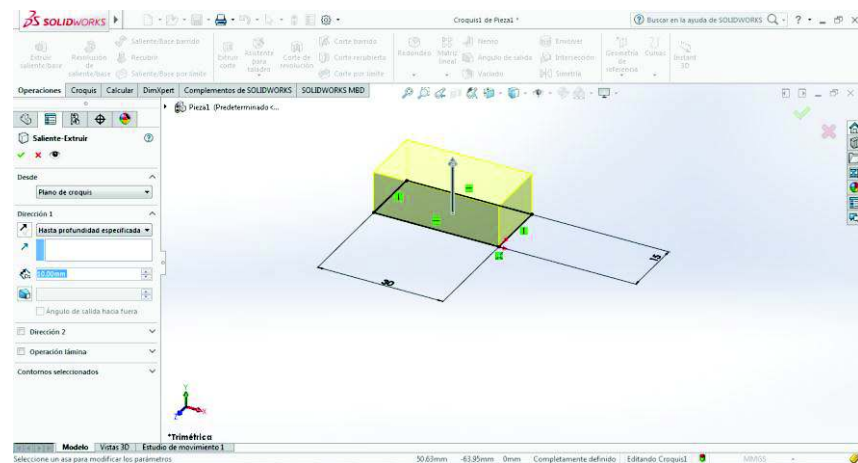


Figura B.7 Operación Extruir Saliente/ Base

Después se selecciona la opción Operación, la cual permite extruir saliente o Revolución Saliente para formar el sólido, esta operación permite dar una medida de profundidad.

El siguiente paso es guardar la pieza, cada pieza diseñada sigue los pasos anteriores, se opera en el primer paso es decir la forma que se quiera dar para crear al solido se lo hace en la opción de Pieza.

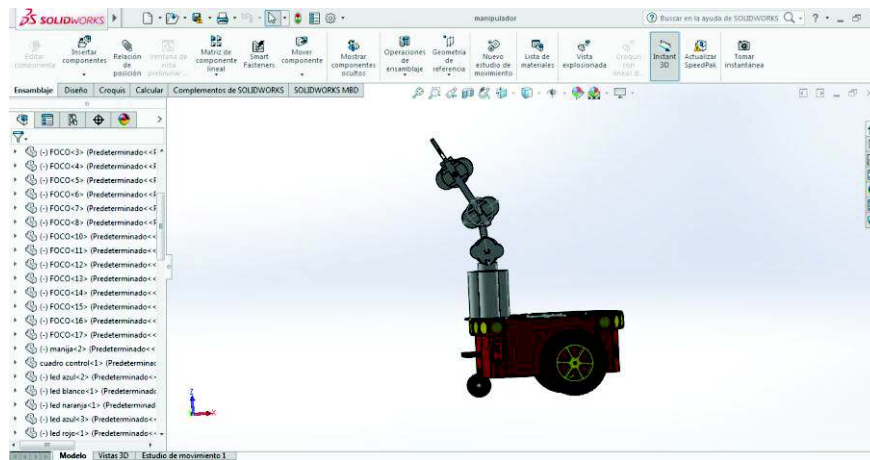


Figura B.8 Paso 3 Ensamblaje de Piezas.

Luego se crea un nuevo archivo donde se selecciona la opción ENSAMBLAJE ver Figura B.8, esta permite unir todas las piezas creadas para formar en una sola pieza el sólido deseado en este caso el Manipulador Móvil.

ANEXO C

MANUAL DE USUARIO DE LA INTERFAZ GRÁFICA

1. Instalar en su computador el programa Matlab R2015a o superior
2. Abrir la carpeta “Interfaz GUI Manipulador Móvil” y ejecutar el archivo “GENERAR.m”
3. Aparecerá la pantalla de bienvenida de la interfaz, como se muestra en la Figura C.1.

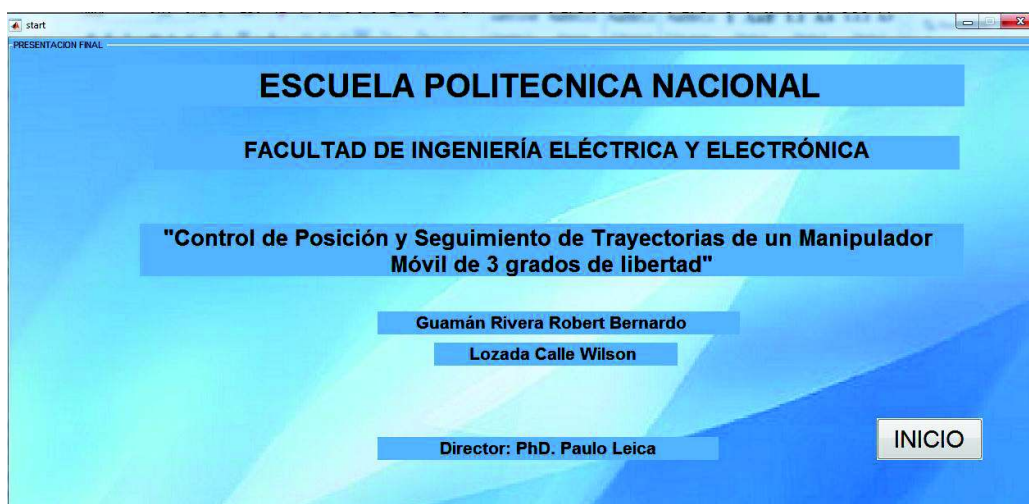


Figura C.1 Pantalla de bienvenida de la interfaz gráfica

4. Para empezar pulse el botón “INICIO” y aparecerá la pantalla de selección del modelo del manipulador móvil, como se muestra en la Figura C.2.

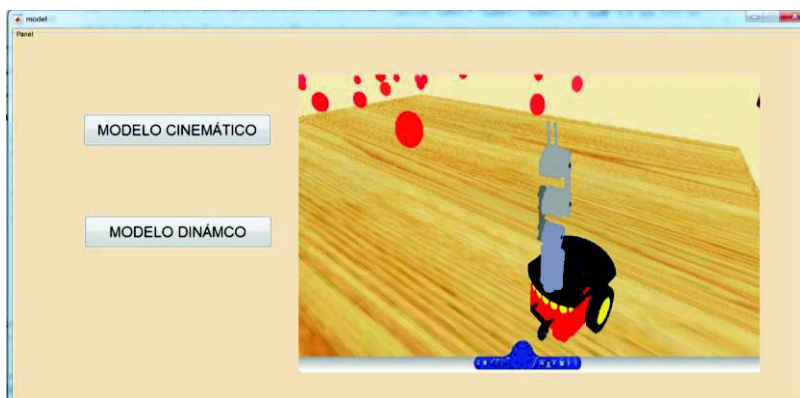


Figura C.2 Pantalla de selección del modelo

5. Seleccione una de las dos opciones y aparecerá la pantalla de selección entre control de posición o seguimiento de trayectorias, como lo muestra la Figura C.3.

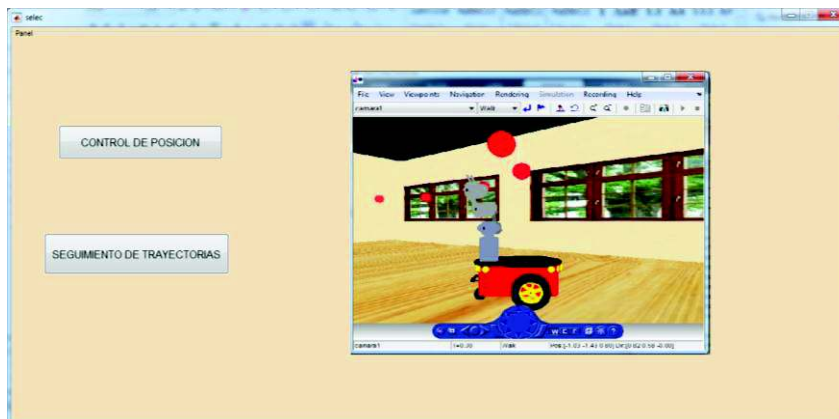


Figura C.3 Pantalla de selección del tipo de control

6. Seleccione una de las opciones y aparecerá la pantalla de simulación, si elige seguimiento de trayectorias aparecerá la pantalla mostrada en la figura C.4.



Figura C.4 Pantalla de simulación de seguimiento de trayectorias

7. En la pantalla de simulación puede elegir el tipo de controlador entre: PID, EULER, CMD-P y CMD-PI. De igual manera puede elegir entre cuatro trayectorias para el efector final: Seno, Cuadrada, Ocho y Circular.

8. Una vez elegido el controlador y la trayectoria deseada, presione el botón “GRAFICAR ERRORES” para obtener las gráficas de error en función del tiempo para los ejes X, Y, Z y la trayectoria en el plano XY. Presione el botón “REALIDAD VIRTUAL” para obtener la simulación en el entorno virtual. Presione el botón “COMPARAR” para obtener en una sola gráfica la evolución del error en función del tiempo de los cuatro controladores.
9. Si elige la opción control de posición se desplegará la pantalla de simulación mostrada en la Figura C.5. En esta pantalla puede ingresar las coordenadas del punto de referencia, si no ingresa ningún punto el programa simulará con el punto de referencia dado por defecto, al pulsar el botón “GRAFICAR ERROR” se presentará la gráfica de la evolución del error de posición en función del tiempo. Los otros botones cumplen las mismas funciones de la pantalla de simulación del seguimiento de trayectorias, explicado en C.8.



Figura C.5 Pantalla de simulación de control de posición