

ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA

DISEÑO Y SIMULACIÓN DE DOS CONTROLADORES, PID NO LINEAL Y PID DUAL NO LINEAL, PARA EL SEGUIMIENTO DE TRAYECTORIA DEL ROBOT DIFERENCIAL UNICICLO PIONEER 3DX MEDIANTE SIMULINK Y COPPELIA-SIM (TOMO I)

DISEÑO Y SIMULACIÓN DE PID NO LINEAL PARA EL SEGUIMIENTO DE TRAYECTORIA DE ROBOT PIONEER 3DX

TRABAJO DE INTEGRACIÓN CURRICULAR PRESENTADO COMO REQUISITO PARA LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN ELECTRÓNICA Y AUTOMATIZACIÓN

DANIEL ENRIQUE MORALES MARTÍNEZ

daniel.morales@epn.edu.ec

DIRECTOR: ING. GEOVANNY DANILO CHÁVEZ GARCÍA PhD.

danilo.chavez@epn.edu.ec

DMQ, Febrero 2022

CERTIFICACIONES

Yo, DANIEL ENRIQUE MORALES MARTÍNEZ declaro que el trabajo de integración curricular aquí descrito es de mi autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

DANIEL MORALES

Certifico que el presente trabajo de integración curricular fue desarrollado por Daniel Enrique Morales Martínez, bajo mi supervisión.

ING. DANILO CHÁVEZ PhD.
DIRECTOR

DECLARACIÓN DE AUTORÍA

A través de la presente declaración, afirmamos que el trabajo de integración curricular aquí descrito, así como el (los) producto(s) resultante(s) del mismo, son públicos y estarán a disposición de la comunidad a través del repositorio institucional de la Escuela Politécnica Nacional; sin embargo, la titularidad de los derechos patrimoniales nos corresponde a los autores que hemos contribuido en el desarrollo del presente trabajo; observando para el efecto las disposiciones establecidas por el órgano competente en propiedad intelectual, la normativa interna y demás normas.

DANIEL ENRIQUE MORALES MARTÍNEZ

GEOVANNY DANILO CHAVEZ GARCIA

HANS JHANDRY LEÓN CHAMBA

DEDICATORIA

Dedico este trabajo a mis padres que han sido quienes me han permitido cursar y terminar mis estudios, además de llevarme a ser la persona que soy hoy en día.

AGRADECIMIENTO

A mis padres Nelson y Paulina, gracias por haber dado tanto por nuestra familia y por forjar quienes somos ahora, este logro les pertenece a ustedes.

A mis hermanos Adrián y Andrea por su apoyo incondicional que no se limita a esta etapa que termina.

A mi compadre Hans, con quien tuve el privilegio de haber cursado este camino escalando peldaño por peldaño hasta superarlo, felicidades a ti también, y gracias por tu amistad.

A mi familia que siempre fue mi motivación para avanzar y quienes me dieron la fuerza que necesitaba. También a mi Papi Valo que nos protege desde arriba, espero poderte haber enorgullecido, viejito.

Un especial agradecimiento al Dr. Chávez y el Dr. Camacho por sus valiosos aportes a nuestro aprendizaje en el desarrollo de este trabajo.

Finalmente agradecer a todos los amigos con los que compartí estos años y de quienes me llevo muy gratos recuerdos, una mención a los Unitarios cuya amistad se remonta a los primeros días de la carrera compartiendo risas y lágrimas y a quienes les deseo el mayor de los éxitos.

ÍNDICE DE CONTENIDO

CERTIFICACIONES	I
DECLARACIÓN DE AUTORÍA	II
DEDICATORIA	III
AGRADECIMIENTO	IV
ÍNDICE DE CONTENIDO.....	V
RESUMEN.....	VII
ABSTRACT.....	VIII
1.INTRODUCCIÓN	
1.1 Objetivo general	2
1.2 Objetivos específicos	2
1.3 Alcance	2
1.4 Marco teórico	3
1.4.1 Robótica móvil.....	3
1.4.2 Control de robots móviles	4
1.4.3 Controladores PID.....	5
1.4.4 Índices de rendimiento	6
2 METODOLOGÍA.....	7
2.1 Robots unicycle de tracción diferencial y robot Pioneer 3DX.....	8
2.2 Modelos matemáticos de robots tipo unicycle	9
2.3 Software utilizado para la simulación	13
2.4 Aproximación del sistema a modelo de primer orden	15
2.5 Controlador cinemático para seguimiento de trayectoria.....	19
2.6 Controlador PID no lineal	19
3 RESULTADOS, CONCLUSIONES Y RECOMENDACIONES	23
3.1 Resultados de simulación para trayectoria circular.....	23
3.2 Resultados de simulación para trayectoria lemniscata	29
3.3 Resultados de simulación para trayectoria rectangular	34
3.4 Comparación de resultados de PID no lineal frente PID tradicional.....	41
3.5 Simulación en CoppeliaSim	43
3.6 Conclusiones	44
3.7 Recomendaciones	45
4 REFERENCIAS BIBLIOGRÁFICAS.....	46
5 ANEXOS	49

ANEXO I.....	49
ANEXO II.....	50

RESUMEN

En el presente trabajo se desarrolla la implementación de un controlador tipo PID no lineal para el seguimiento de trayectoria del robot de tracción diferencial tipo unicycle Pioneer 3DX, con el objetivo de analizar las ventajas y desventajas que pueda presentar un controlador PID con valores variables en el tiempo frente a uno con valores constantes, implementado en trabajos previos [1].

Se implementará particularmente un PID no lineal con valores de K_p y K_i dependientes del error de la variable controlada, de modo que se pueda obtener una acción de control que se adapte al estado del sistema en un instante de tiempo en el que pueda requerir una acción de control más agresiva, o una más suave [2].

Para probar el controlador diseñado, se busca implementarlo en la plataforma Pioneer 3DX, en un esquema de seguimiento de trayectoria, cuyo principio de funcionamiento es que un robot "ideal" virtual con control cinemático genere las referencias necesarias para el robot real, de modo que el último siga al robot ideal [3]. El robot real será simulado en el programa Coppelia Sim que incluye el modelo dinámico del equipo, en un entorno 3D que además proporciona mediciones en tiempo real de posiciones, velocidades y orientación del robot, valores que serán realimentados hacia Simulink, en donde se implementarán los algoritmos de control y la generación de trayectorias. La interacción entre ambas plataformas se la realizará mediante comunicación síncrona en una estructura de Software in Loop.

PALABRAS CLAVE: Controlador PID, Constantes de control, error en estado transitorio, robot de tracción diferencial.

ABSTRACT

In the present document, the implementation of a non-linear PID type controller for the trajectory tracking of the Pioneer 3DX unicycle-type differential traction robot is developed, with the aim of analyzing the advantages and disadvantages that a PID controller with variable gains can present, versus one with constant ones, implemented in previous works [1].

In particular, a non-linear PID will be implemented with K_p and K_i values dependent on the error of the controlled variable, so that a control action can be obtained that adapts to the state of the system at an instant of time in which control action may be required [2].

To test the designed controller, we seek to implement it on the Pioneer 3DX platform, in a trajectory tracking scheme, whose operating principle is that a virtual "ideal" robot with kinematic control generates the necessary references for the real robot, so that the latter follows the ideal robot [3]. The real robot will be simulated in the Coppelia Sim program that includes the dynamic model of the equipment, in a 3D environment that also provides real-time measurements of the robot's positions, speeds and orientation, values that will be fed back to Simulink, where the parameters will be implemented. control algorithms and trajectory generation. The interaction between both platforms will be carried out through synchronous communication in a Software in Loop structure.

KEYWORDS: PID Controller, Control Constants, Transient state Error, Differential Traction.

1 INTRODUCCIÓN

En años recientes es cada vez más común la presencia de robots que cumplen con varias tareas específicas que anteriormente eran realizadas por humanos. Entre todas esas tareas se encuentran algunas que tienen como común denominador la locomoción del robot, es decir que requieren que un robot móvil realice cierto tipo de movimiento a fin de cumplir con su objetivo, como puede ser el caso de robots bodegueros de Amazon que llevan objetos dentro de la bodega, o incluso taxis automáticos que sean capaces de transportar personas dentro de una ciudad. En todos esos casos el robot debe ser capaz de circular por una trayectoria específica en un tiempo determinado. Se pueden ejecutar el seguimiento mediante varios algoritmos de control cuya efectividad depende del tipo de robot, sin embargo, los más comunes implican un controlador cinemático que genere referencias de velocidad lineal y angular que el robot pueda seguir. Para el correcto funcionamiento del sistema, es recomendable que las referencias de velocidad generadas en el controlador cinemático pasen por reguladores que se encarguen de que el robot siga las referencias de velocidad correspondientes.

En trabajos anteriores se han implementado controladores tradicionales de tipo PI con excelentes resultados en el seguimiento de trayectorias de distintos tipos, en este trabajo se propone una modificación del esquema de PID común, llamado PID no lineal, en el cual los parámetros sintonizados se convierten de un valor constante a uno variable dependiente del error que tenga el sistema en un instante determinado, de modo que el controlador sea capaz de generar una acción de control más o menos agresiva según el estado del sistema, es decir que durante el estado transitorio con magnitudes de error más elevadas, la acción de control tenderá a ser más agresiva, mientras que en estado estable que tiene errores más pequeños, la acción de control sea suave.

En el capítulo 2 se implementará un esquema de control de seguimiento de trayectoria completo, que contempla la generación de trayectoria, el controlador cinemático para robótica móvil, y controladores de velocidad basados en PID no lineales que garanticen que las velocidades reales del robot se ajusten a las referencias creadas por el controlador de seguimiento de trayectoria.

Los resultados de los experimentos realizados para varias trayectorias se presentarán en el capítulo 3 junto con su correspondiente análisis y la presentación de los índices de desempeño ISE que presente el controlador propuesto.

1.1 Objetivo general

Diseñar y simular dos controladores, PID No Lineal y PID Dual No Lineal para el seguimiento de trayectoria del robot diferencial unicycle Pioneer 3DX mediante Simulink y Coppelia-Sim.

1.2 Objetivos específicos

1. Realizar una recopilación bibliográfica de: los diferentes esquemas utilizados para el control de trayectoria en robots móviles, los fundamentos de control para PID y para sus variaciones propuestas.
2. Diseñar y sintonizar un esquema de control por PID no lineal y PID dual no lineal para el seguimiento de trayectorias de robots móviles.
3. Simular el seguimiento de trayectorias mediante el uso un robot móvil Pioneer 3DX mediante un esquema Software in Loop, utilizando los softwares CoppeliaSim y Simulink
4. Probar y comparar los esquemas de control propuestos frente a un controlador PID clásico, realizando una cuantificación por índices de desempeño, y estudiando el estado transitorio de las respuestas.
5. Desarrollar una interfaz gráfica en App Designer de Matlab para poder seleccionar el controlador, la trayectoria y visualizar los datos en pantalla.

1.3 Alcance

1. Se investigará sobre modelos cinemáticos del Pioneer 3DX aplicables al seguimiento de trayectoria, con el objetivo de obtener un modelo de primer orden con retardo.
2. Se estudiará la base teórica sobre los controladores PID no lineal y PID dual no lineal, para su aplicación en el control de trayectoria de un robot móvil.
3. Se diseñará un esquema de control por PID no lineal y otro por PID dual no lineal a partir del modelo de primer orden con retardo para su aplicación en el modelo del robot.

4. Se verificará el funcionamiento del sistema propuesto mediante simulación con diferentes trayectorias (cuadrada, circular, lemniscata) con los esquemas de control propuestos. Para lo cual se ocupará Simulink y Coppelia-Sim.
5. Se realizarán comparaciones entre los controladores propuestos mediante su cuantificación de índices de desempeño, y la respuesta en estado transitorio de cada controlador, observando máximo sobrepico y tiempo de establecimiento.
6. Se desarrollará una interfaz gráfica con el software App Designer para seleccionar el controlador, la trayectoria y observar resultados en pantalla, junto con los índices de rendimiento.

1.4 Marco teórico

El presente trabajo se enfoca en el diseño y simulación de un algoritmo de seguimiento de trayectoria para el robot Pioneer 3DX, que ocupe controladores de velocidad de tipo PID no lineal. En este capítulo se realizará la revisión de la base teórica en cuanto a conceptos sobre robótica móvil y control de procesos, temas que van a ser fundamentales durante el desarrollo del proyecto.

1.4.1 Robótica móvil

Se define a un robot móvil como un sistema de tipo electromecánico cuya principal característica es su capacidad de movilizarse de un lugar a otro sin necesidad de interacción humana, a través de un medio físico definido (tierra, agua, aire, etc.) mediante un mecanismo de locomoción específico, como pueden ser ruedas, patas, orugas, hélices, etc. [3]



Figura 1.1. Clases de robots según su mecanismo de locomoción

Debido a la naturaleza autónoma de los robots móviles, estos suelen incluir varios sensores destinados a diferentes acciones, como es el caso de los sensores infrarrojos que utilizan los robots seguidores de línea básicos; en casos más avanzados como el caso de robots de cultivo que son básicamente seguidores de línea incluyen sensores analógicos de color

[4] o inclusive cámaras con software de tratamiento de imagen de modo que puedan identificar la línea que se desea que sigan de forma automática.

Vehículos aéreos no tripulados, o UAV por sus siglas en inglés suelen incluir sensores más avanzados como velocímetros, giroscopios o sensores de fuerzas gravitacionales [5].

En varias aplicaciones de robots móviles, la inclusión de instrumentos de medición acordes a su medio se realiza con el fin de poder utilizar dichas mediciones para estimar la posición relativa del robot respecto a su punto de partida, estudio al que se lo llama odometría y que resulta de suma importancia para el seguimiento de trayectoria cuando no se tiene un camino claro [6].

1.4.2 Control de robots móviles

Se identifican dos tipos principales de control para un robot móvil, el primero de ellos es el control de posición, donde se le entrega al robot un punto al que debe movilizarse desde su posición inicial, sin importar el camino que siga ni su orientación final [7].

El segundo tipo de control es el de seguimiento de trayectoria, en el cual se desea que un robot se mueva entre dos puntos, pero siguiendo un camino deseado dentro de un límite de tiempo, este caso es una ampliación del control de posición ya que una trayectoria es en sí una serie de coordenadas en el tiempo que se van generando en intervalos regulares, y que el robot debe seguir conforme aparezcan, en este caso la orientación del robot es relevante para un óptimo seguimiento de la trayectoria propuesta [8].

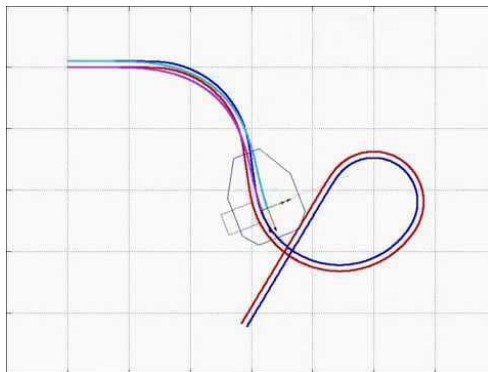


Figura 1.2. Ejemplo de seguimiento de trayectoria para un robot móvil terrestre

La generación de trayectorias es un componente importante en la parte de seguimiento de las mismas, pueden tratarse de trayectorias con formas preplanificadas que ocupan ecuaciones variantes en el tiempo que generan una serie de posiciones en el plano con una velocidad definida [9], generalmente se utilizan para realizar pruebas para los algoritmos de control del robot, como es el caso de este trabajo.

Existe una disciplina encargada de estudiar la capacidad de un robot para determinar automáticamente la ruta óptima entre un punto A y un punto B mediante varios algoritmos que incluyen tratamiento de datos de instrumentación e inteligencia artificial. A esta disciplina se la llama planeación de ruta o de caminos y es una ciencia bastante popular hoy en día sobre todo en el campo de inteligencia artificial y machine-learning [10].

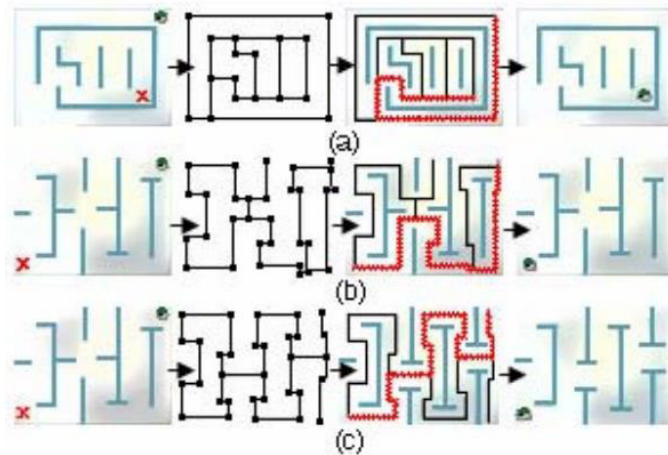


Figura 1.3. Ejemplo de planeación de caminos para laberintos de complejidad variada

La integración de las disciplinas de planeación de rutas y seguimiento de trayectoria es la que ha abierto las puertas a robots con autonomía mejorada, capaces de transportar carga o personas largas distancias y en entornos más complejos.

1.4.3 Controladores PID

Los esquemas de control vistos previamente para el seguimiento de trayectoria implican la realimentación de varias señales para poder garantizar que el robot se encuentre sobre la referencia, principalmente se requieren datos de posiciones (obtenidos por odometría) y velocidades (obtenidas por mediciones en el robot), con los que se opera a lazo cerrado. Es necesario un controlador que se encargue de que las variables realimentadas tengan errores cercanos a cero respecto a la referencia creada en la etapa de generación de trayectoria, que sea capaz de operar ante cambios bruscos de referencia y que sea robusto para rechazar perturbaciones en el sistema.

Los PID han demostrado ser controladores bastante confiables, robustos y relativamente simples de sintonizar; en el presente trabajo se centra en una variación de este tipo de controladores, por lo que es necesario conocer la base teórica al respecto.

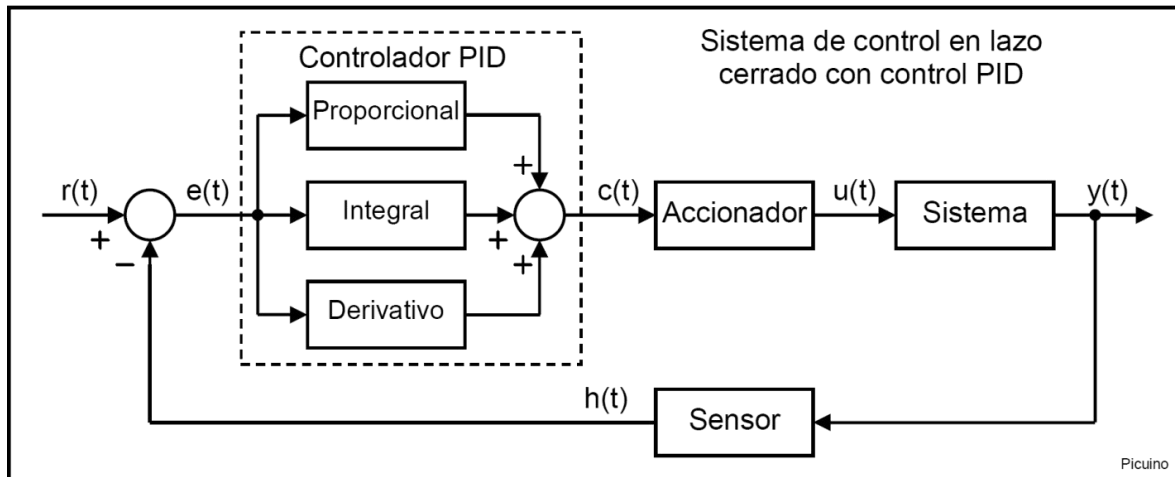


Figura 1.4. Esquema de operación de un sistema en lazo cerrado con controlador PID

El controlador PID se encuentra compuesto por tres componentes que proporcionan acciones de control, una proporcional que básicamente multiplica el error por una ganancia K_p e intenta minimizar al mismo, la acción proporcional se incrementa con el error.

La acción integral se calcula con la integral del error, que se define como la suma del error en el tiempo, de modo que la acumulación de la señal de error eleve la acción de control integral, hasta que se reduce el error en estado estable y aumenta la velocidad de la respuesta, pero ingresa inercia al sistema, por lo que se sacrifica la estabilidad del mismo.

La última acción del PID es la derivativa, que se calcula en base a la velocidad con la que el sistema se acerca a la referencia para poder frenar la respuesta en caso de ser demasiado rápida, ganando estabilidad a cambio de sacrificar su rapidez.

La popularidad del PID se debe también a la relativa simpleza con la que se lo puede sintonizar, existiendo varios métodos que van desde el tanteo de valores, hasta métodos lineales que se basan en la medición de parámetros de la curva de reacción [11].

1.4.4 Índices de rendimiento

Se define a los índices de desempeño como una medida cuantitativa que mide el rendimiento de un sistema en base a alguna variable del mismo, en pocas palabras viene a ser un número que permite evaluar la calidad de un sistema de control. En este proyecto se ocupa el índice ISE, o criterio de la integral del cuadrado del error (Integral Square Error) que tiene la siguiente ecuación:

$$ISE = \int e(t)^2 dt \quad (1.1)$$

El índice se ocupa con el fin de ajustar un controlador sintonizado por curva de reacción a fin de obtener la mejor respuesta posible, esto se realiza manipulando las ganancias y observando el valor del ISE, hasta que se logre llegar a su valor mínimo [12].

2 METODOLOGÍA

El presente trabajo experimental se ha desarrollado con el afán de observar respuestas del sistema ante diferentes entradas, analizar sus formas e índices cuantitativos de rendimiento para poder llegar a un resultado mediante método descriptivo.

Para el desarrollo teórico se optó por utilizar información procedente de trabajos de titulación anteriores que hayan ahondado sobre todo en el tema de robótica móvil y diseño de controladores avanzados. También se realizó una recopilación bibliográfica sobre los temas principales en artículos publicados, revistas o libros de robótica y control de procesos industriales.

En el capítulo 1 se establecieron los objetivos y alcance del presente trabajo, así también se exploró la teoría básica sobre robótica móvil, tema que será profundizado a futuro en éste mismo capítulo.

Ahora que ya se ha revisado conceptos básicos respecto al tipo de robots al que va enfocado el trabajo de investigación, en el capítulo 2 se expondrá su modelado matemático, además que aparecerá el robot móvil Pioneer 3DX por primera vez, siendo el robot al que se busca controlar; éste robot posee ya varios modelos matemáticos desarrollados en base a repetidas pruebas sobre el mismo, con los que se puede estimar sus comportamientos físicos y que sirven para el desarrollo de controladores más avanzados que se encuentran fuera del alcance de éste trabajo. Debido a que se busca reducir la complejidad de los modelos dinámicos, se modelarán las respuestas del robot como sistemas de primer orden con retardo, que permitirán el diseño de controladores utilizando técnicas aplicables a sistemas lineales.

El programa CoppeliaSim incluye en sus librerías un modelo 3D del robot Pioneer 3DX que será útil a la hora de realizar pruebas para obtener modelos o validar controladores, debido a que se comporta como una planta normal, a la que ingresa las velocidades de los motores y entrega velocidades lineal y angular, posiciones en los ejes X y Y, y ángulo de orientación.

Para la implementación del generador de trayectoria, el algoritmo de seguimiento y el controlador planteado, además de la visualización de las variables medidas y la validación del modelo matemático, se utilizará el software Simulink de Matlab, que incluye varias

funciones propias para el control de procesos que son muy útiles para el control de robótica móvil también.

Se busca que ambos programas trabajen conjuntamente, para lo cual se los conectó mediante un esquema de Software in Loop, de modo que las mediciones realizadas en CoppeliaSim puedan ingresar a los algoritmos de control de Simulink, y las acciones de control de Simulink puedan llegar al robot en CoppeliaSim.

2.1 Robots unicycle de tracción diferencial y robot Pioneer 3DX

Un robot unicycle se define como un robot móvil que usa ruedas como medio de locomoción, generalmente cuentan con 2 ruedas fijas colocadas en un mismo eje, cuya velocidad de rotación puede ser controlada debido a que cada una se encuentra acoplada a un motor de DC. Se suele incluir una tercera rueda que generalmente se ubica en la parte posterior del robot, debido a que su única función es mejorar la estabilidad del robot, ésta rueda no es controlable [13].

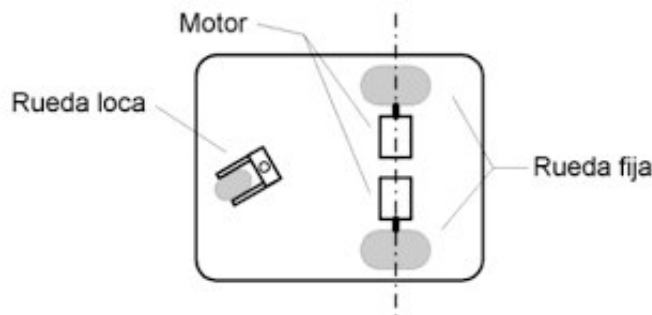


Figura 2.1. Modelo general de un robot móvil tipo unicycle de tracción diferencial

El robot Pioneer 3DX es un ejemplo de robot móvil de tracción diferencial tipo unicycle. Se trata de una plataforma desarrollada por Mobilerobots orientada a la educación e investigación debido a su relativa facilidad de uso y alta confiabilidad. Si bien es un tema que se tocará posteriormente, al ser un robot de uso tan extenso existen ya modelos 3D en las librerías de robótica móvil del software CoppeliaSim, que además permite simular el comportamiento dinámico del robot sin necesidad de ocupar su modelo matemático [14].



Figura 2.2. Robot Pioneer 3DX

El fabricante del robot incluye en su datasheet valores que resultarán ser útiles en secciones posteriores [15]:

Tabla 2.1. Valores proporcionados por el fabricante en datasheet del Pioneer 3DX

Parámetro	Valor
Diámetro de las ruedas	195 mm.
Longitud del eje de las ruedas	381 mm.
Velocidad lineal máxima	1 m/s
Velocidad angular máxima	1 rad/s

2.2 Modelos matemáticos de robots tipo unicycle

El primer modelo que se plantea del robot se trata del cinemático básico, que permite conocer, mediante ecuaciones matemáticas, las variables físicas que tienen que ver con el movimiento del robot, como lo son posiciones y velocidades, no se considera la inercia del robot o las fuerzas que producen su movimiento [16].

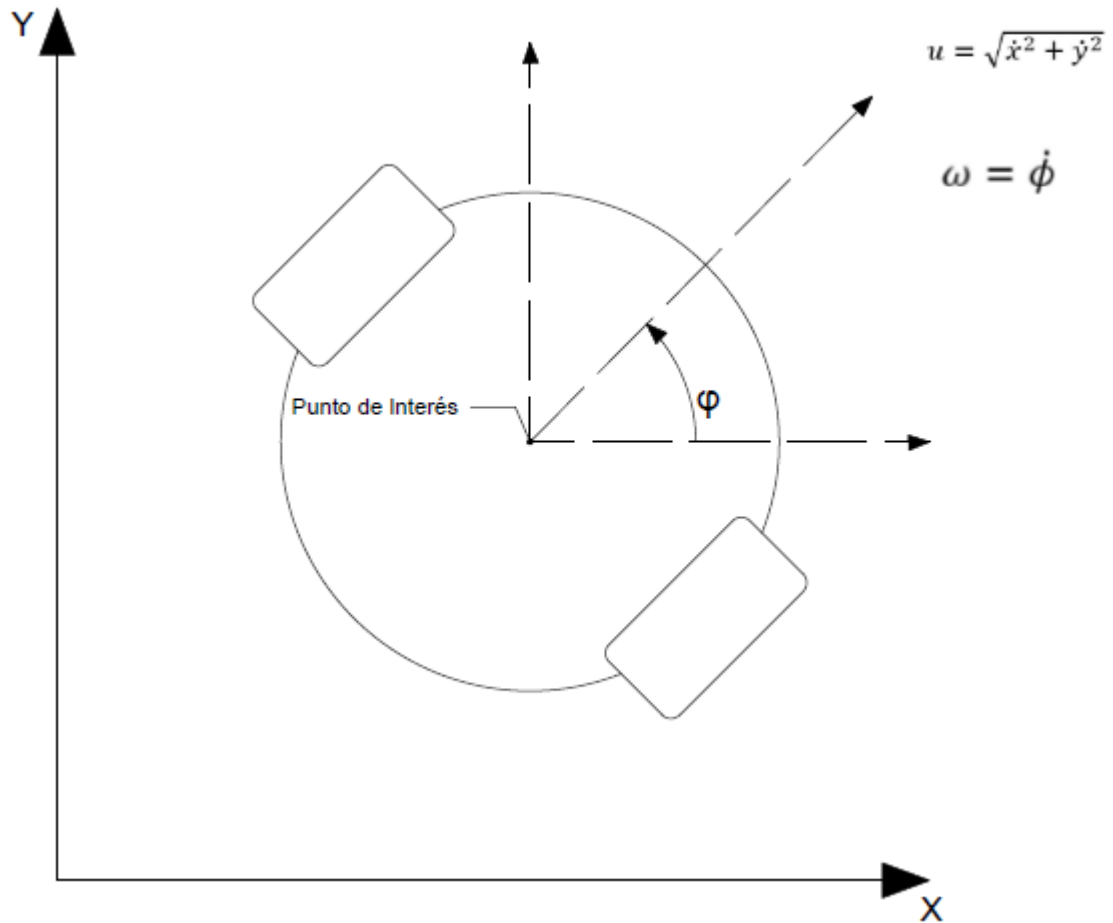


Figura 2.3. Modelo geométrico para un robot uniciclo con restricción no holonómica

El modelo cinemático contempla una velocidad lineal u compuesta por las velocidades lineales en X y Y del robot, y una angular ω que representa la variación del ángulo de orientación ϕ del robot respecto al eje X, generándose así las siguientes ecuaciones:

$$\dot{x} = u \cos(\phi) \quad (2.1)$$

$$\dot{y} = u \sin(\phi) \quad (2.2)$$

$$u = \sqrt{\dot{x}^2 + \dot{y}^2} \quad (2.3)$$

$$\omega = \dot{\phi} \quad (2.4)$$

El modelo cinemático se puede expresar en variables de estado de la siguiente forma:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} \cos(\phi) & 0 \\ \sin(\phi) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} u \\ \omega \end{bmatrix} \quad (2.5)$$

Con el modelo cinemático básico es posible conocer, mediante integración, los cambios de posición y orientación que experimenta el robot ante el ingreso de distintas velocidades lineales y angulares.

El modelo cinemático básico presenta varios inconvenientes, el primero de ellos es la restricción no holonómica propia del robot, la cual se define como la incapacidad del robot de ejecutar ciertos movimientos por la propia disposición física de sus componentes. En el caso del Pioneer 3DX, resulta imposible el movimiento lateral de forma perpendicular al eje de sus ruedas.

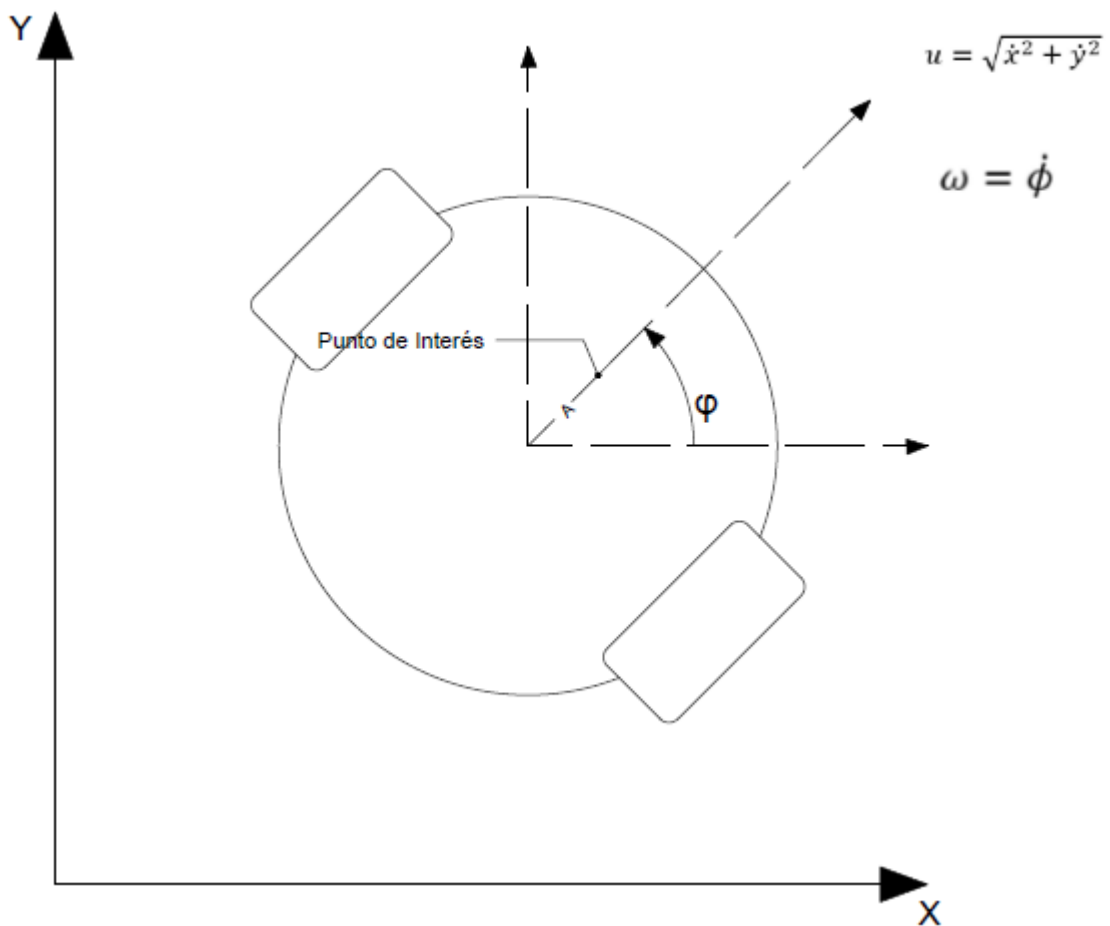


Figura 2.4. Modelo geométrico para un robot uniciclo con restricción no holonómica mejorada

Para solventar este inconveniente, se plantea el modelo cinemático con restricción no holonómica mejorada, el cual básicamente desplaza el punto de interés del robot desde su centro hacia un punto A, que puede estar delante o detrás del eje de las ruedas. El modelo cinemático queda descrito con las siguientes ecuaciones [17]:

$$\dot{x} = u \cos(\phi) - A\omega \sin(\phi) \quad (2.6)$$

$$\dot{y} = u \sin(\phi) + A\omega \cos(\phi) \quad (2.7)$$

$$u = \sqrt{\dot{x}^2 + \dot{y}^2} \quad (2.8)$$

$$\omega = \dot{\phi} \quad (2.9)$$

El modelo en variables de estado queda expresado de la siguiente manera:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} \cos(\phi) & -A \sin(\phi) \\ \sin(\phi) & A \cos(\phi) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} u \\ \omega \end{bmatrix} \quad (2.10)$$

A pesar de la mejora en las restricciones de movilidad del robot, el modelo cinemático no toma en cuenta variables físicas relacionadas con la masa del robot, las que pueden generar alteraciones en el movimiento debido a inercia o fricciones. Para ello se plantea el modelo dinámico del robot, que en sí es una mejora del modelo cinemático, que además de la mejora de restricción no holonómica, incluye la aceleración lineal y angular producida por fuerzas relacionadas con la masa del robot, las magnitudes de estas dependen de parámetros numéricos que se determinan de forma experimental y cuyos valores son propios de cada modelo de robot. En el caso del Pioneer 3DX, el modelo dinámico tiene la siguiente forma:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\phi} \\ \dot{u} \end{bmatrix} = \begin{bmatrix} u \cos(\phi) - A\omega \sin(\phi) \\ u \sin(\phi) + A\omega \cos(\phi) \\ \omega \\ \frac{\theta_3^0}{\theta_1^0} \omega^2 - \frac{\theta_4^0}{\theta_1^0} u \\ \frac{-\theta_5^0}{\theta_2^0} u \omega - \frac{\theta_6^0}{\theta_2^0} \omega \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 9 \\ 0 & 0 \\ \frac{1}{\theta_1^0} & 0 \\ 0 & \frac{1}{\theta_2^0} \end{bmatrix} \begin{bmatrix} u_{ref} \\ \omega_{ref} \end{bmatrix} \quad (2.11)$$

Los valores de θ en el modelo corresponden a funciones de algunos parámetros físicos del robot, como puede ser la masa, inercia, diámetro de las ruedas y parámetros de los motores del robot. Cabe recalcar que los parámetros pueden ser obtenidos matemáticamente o de forma experimental. Los valores de θ son los siguientes para el robot Pioneer 3DX:

$$\theta_1^0 = 0.24089$$

$$\theta_2^0 = 0.2424$$

$$\theta_3^0 = -0.00093603$$

$$\theta_4^0 = 0.99629$$

$$\theta_5^0 = -0.0037256$$

$$\theta_6^0 = 1.0915$$

2.3 Software utilizado para la simulación

Para realizar las pruebas necesarias en el robot mediante simulación, se utilizaron dos programas funcionando en un esquema de software in loop. El primero de ellos es el software CoppeliaSim, especializado en la simulación de robots en un entorno gráfico tridimensional, en donde el comportamiento del modelo se aproxima al de un robot real, incluyendo el comportamiento dinámico del mismo, por lo cual puede ser utilizado en lugar del modelo matemático en la ejecución de las simulaciones requeridas, tanto para modelado del sistema como para sintonización de controladores [18].

La programación para la generación de trayectorias, controlador cinemático y reguladores de velocidad lineal y angular se realizaron en el software Simulink, de Matlab, mediante programación en bloques [19]. La configuración de software in loop es necesaria debido a que los bloques de control en simulink se encuentran realimentados con el sistema simulado, por lo que se requiere realizar mediciones de algunas variables del robot en CoppeliaSim que serán ingresadas a los algoritmos de control para el cálculo de errores principalmente.

Bajo esta premisa, se puede decir que CoppeliaSim es el software donde se simula el robot Pioneer 3DX y se adquieren las mediciones correspondientes a posición, orientación y velocidades del robot, mientras que en Simulink se encuentran programados los algoritmos de generación y de seguimiento de trayectoria, que en base a los datos realimentados desde CoppeliaSim, generará las acciones de control que comandarán el movimiento del robot para la ejecución del seguimiento de trayectoria. El funcionamiento del sistema se registrará al siguiente esquema:

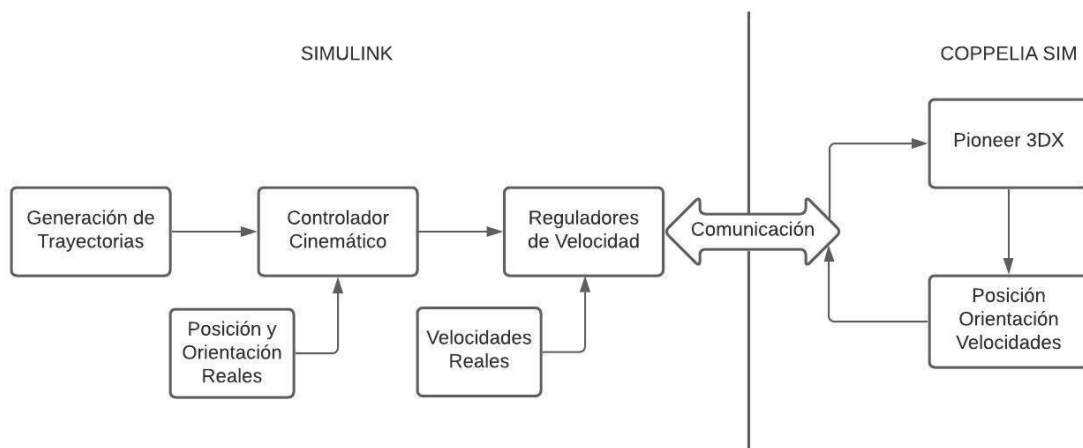


Figura 2.5. Diagrama esquemático de interacción entre funciones de Simulink y CoppeliaSim

Cuando se trabaja en simulación con dos programas diferentes, se puede dar el problema de que uno se ejecute con una base de tiempo diferente al del otro, es un caso que se da cuando se simulan sistemas de control en Simulink, que dependiendo de la carga computacional suele ralentizarse, mientras que CoppeliaSim, al únicamente contener el modelo 3D del robot, suele ejecutarse en tiempo real, esto provoca varios problemas con las mediciones de posiciones y velocidades para el seguimiento de trayectoria.

Para solucionar este problema se utiliza comunicación sincrónica entre Simulink y CoppeliaSim de modo que se garantice que las bases de tiempo con la que simulan ambos programas sean las mismas. CoppeliaSim se ejecuta a manera de “pasos” de 50ms por defecto, en el caso de Simulink depende del solucionador que se implemente, en el caso de uno de paso variable, la base de tiempo de Simulink no es constante por defecto, por lo que se decide ocupar un solucionador de paso fijo, con tiempo de 50ms al igual que CoppeliaSim. La comunicación síncrona involucra el envío de dos líneas importantes, el trigger desde Simulink, que alerta a CoppeliaSim que se va a realizar un envío de datos, a lo que debe ser respondido con una línea de confirmación que envía CoppeliaSim, indicando que se encuentra listo para realizar la comunicación; tras la recepción de la confirmación es Simulink quien inicia el envío de datos [20].

2.4 Aproximación del sistema a modelos de primer orden

El modelo dinámico del robot Pioneer 3DX presentaría una respuesta aproximada a la real, por lo que es útil para simulación, sin embargo, el diseño/sintonización de controladores para velocidad lineal y angular se complica debido a la complejidad del modelo y la presencia de elementos no lineales.

Para poder aplicar la teoría de sintonización de controladores PID en este caso, es necesario aproximar la respuesta del robot a un modelo de primer o segundo orden. Se reconocen como entradas del sistema las velocidades lineal y angular de referencia (las mismas se transforman en velocidades del motor izquierdo y derecho al momento de ingresarlas como comando en el robot) mientras que las salidas vienen a ser las velocidades lineal y angular reales o medidas del robot. Para simplificar la sintonización inicial de los controladores, se aproxima al sistema a uno de primer orden con retardo. La aproximación se realiza mediante la obtención de la curva de reacción del sistema, ingresando una entrada paso y observando la respuesta hasta que llegue a estado estable. A manera de preámbulo, el estado transitorio permitirá estimar la constante de tiempo del sistema, mientras que el estado estable servirá para estimar la ganancia del sistema, el retardo o tiempo muerto se obtendrá midiendo el intervalo de tiempo entre el ingreso de la referencia y la reacción del sistema.

Al momento de realizar las mediciones en el robot simulado, se deben tomar en cuenta ciertas consideraciones:

- La medición de la respuesta, tanto de velocidad lineal como de angular, no debería ingresar en el inicio de la simulación, debido a que se presenta un período de configuración de software que puede alterar la medición del tiempo muerto.
- La referencia inicial no debería ser nula debido a que el robot inicialmente se encuentra en reposo y presenta inercia que se resiste a su transición a movimiento, algo que puede alterar el estado transitorio de la respuesta.

Con las consideraciones anteriores tomadas en cuenta, se puede obtener la curva de reacción del sistema tanto para velocidad lineal como para velocidad angular.

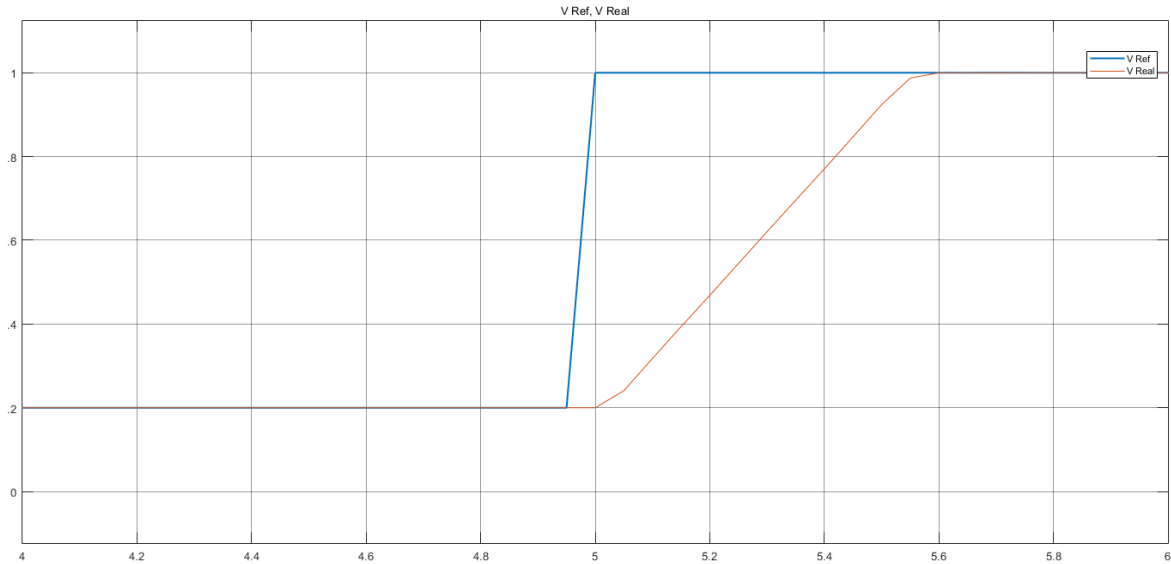


Figura 2.6. Curva de respuesta de velocidad lineal ante entrada paso

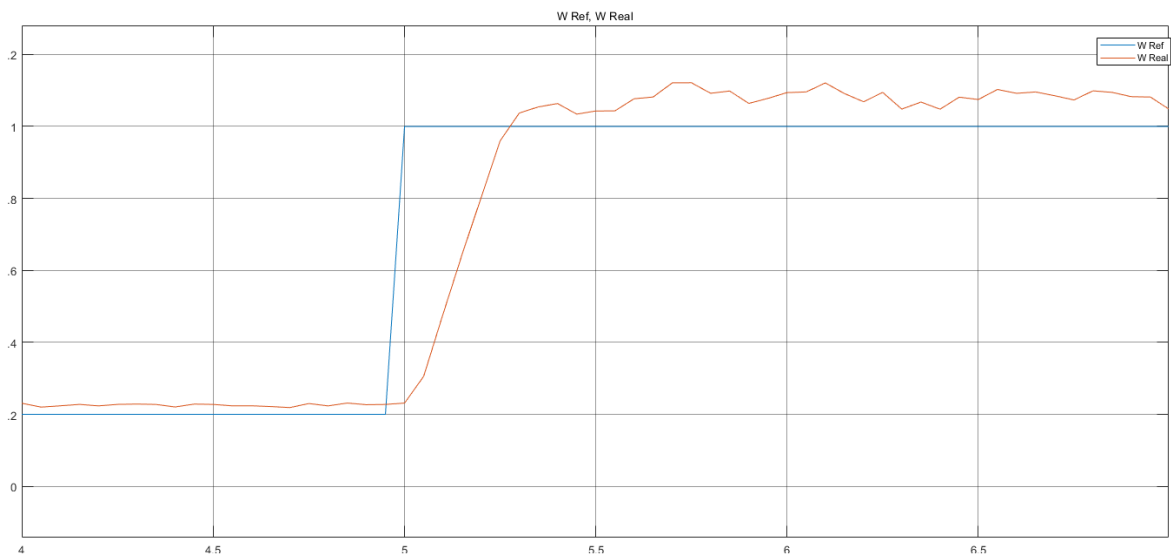


Figura 2.7. Curva de respuesta de velocidad angular ante entrada paso

Para aproximar el sistema a uno de primer orden con retardo se utilizan las ecuaciones de Alfaro, que se basa en obtener el tiempo que la respuesta alcanza el 25% y el 75% de su valor final, medidos desde el momento en que ingresa la referencia paso, para con esos valores calcular la constante de tiempo τ del sistema y su tiempo muerto t_m , con las siguientes ecuaciones [21]:

$$k = \frac{\Delta Y}{\Delta U} \quad (2.12)$$

$$\tau = a t_1 + b t_2 \quad (2.13)$$

$$t_m = c t_1 + d t_2 \quad (2.14)$$

Donde los valores de a, b, c, d son constantes propias del método de Alfaro:

$$a = -0.910$$

$$b = 0.910$$

$$c = 1.262$$

$$d = -0.262$$

Estos valores serán ocupados en la obtención del modelo de primer orden con retardo que tiene la siguiente forma:

$$G(s) = \frac{k_p}{\tau s + 1} e^{-t_m s} \quad (2.15)$$

Los valores de tiempo medidos para la velocidad lineal, junto con la constante de tiempo y tiempo muerto estimados son:

$$t_1 = 0.1540$$

$$t_2 = 0.4191$$

$$\tau = 0.2412$$

$$t_m = 0.0845$$

$$k = 1$$

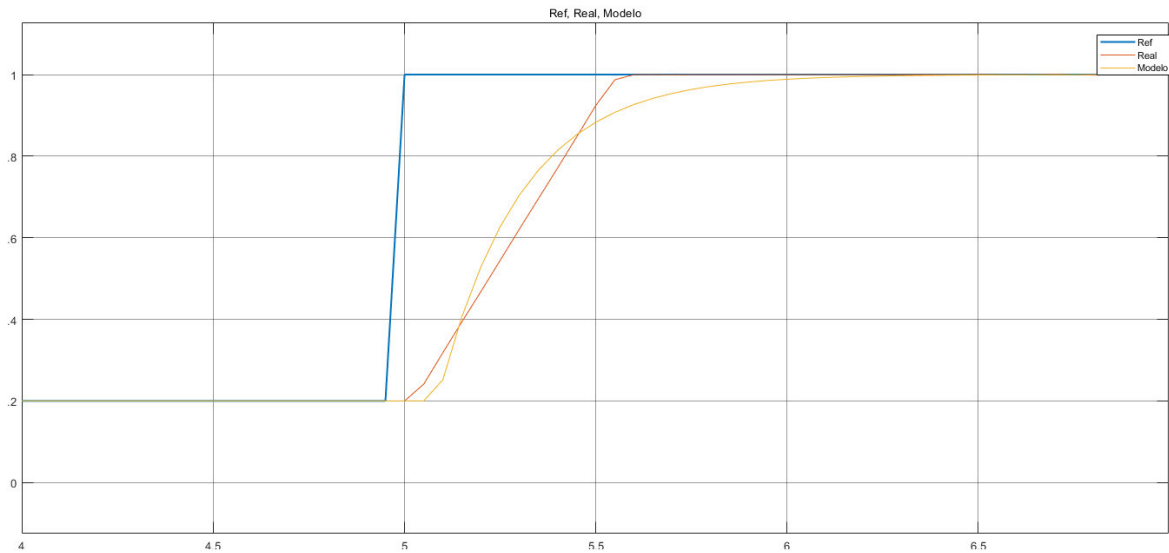


Figura 2.8. Validación de modelo aproximado de primer orden con retardo de la velocidad lineal

Los valores medidos y estimados para la velocidad angular son los siguientes:

$$t_1 = 0.0828$$

$$t_2 = 0.2162$$

$$\tau = 0.1214$$

$$t_m = 0.0478$$

$$k = 1.08$$

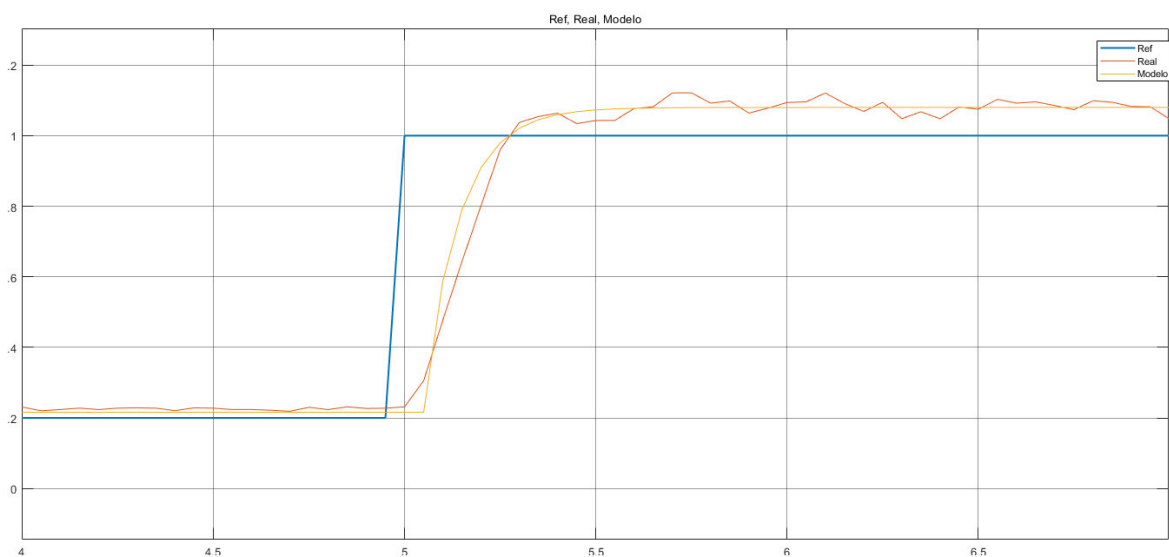


Figura 2.9. Validación de modelo aproximado de primer orden con retardo de la velocidad angular

Una vez se obtuvieron y validaron los modelos de primer orden con retardo, se puede proceder con la sintonización de los controladores PID para velocidad lineal y velocidad angular mediante los métodos simplificados.

2.5 Controlador cinemático para seguimiento de trayectoria

El esquema de control de seguimiento de trayectoria visto en la Figura 2.5 se compone de varios bloques en cascada. El primero de ellos es el generador de trayectorias, cuya tarea es calcular las coordenadas en el plano que el robot debe seguir en el tiempo para poder formar una trayectoria específica. Posteriormente se verá que un robot móvil necesita conocer también la serie de velocidades en el tiempo con las que el robot debe transitar la trayectoria para alcanzar a la serie de puntos, lo cual se puede calcular derivando la serie de posiciones respecto al tiempo utilizando bloques derivativos en Simulink.

El controlador cinemático es el encargado de transformar las referencias de posiciones y velocidades en X y Y que entrega el generador de trayectorias, a referencias de velocidad lineal y angular que serán transmitidas hacia el robot [22].

$$\begin{bmatrix} u_{ref} \\ \omega_{ref} \end{bmatrix} = \begin{bmatrix} \cos(\omega t) & \sin(\omega t) \\ -\frac{1}{A} \sin(\omega t) & \frac{1}{A} \cos(\omega t) \end{bmatrix} \begin{bmatrix} x_{ref} + l_x \tanh\left(\frac{k_x}{l_x} E_x\right) - x \\ y_{ref} + l_y \tanh\left(\frac{k_y}{l_y} E_y\right) - y \end{bmatrix} \quad (2.16)$$

2.6 Controlador PID no lineal

Un controlador PID no lineal se trata de una modificación del controlador tradicional PID en el cual se cambia el esquema de ganancias proporcional, integral y derivativa estáticas por unas que varían su valor dependiendo del estado del sistema en un instante de tiempo. Existen varias propuestas con las que se seleccionaría el valor de las ganancias del controlador, siendo la más aceptada la que utiliza el valor del error como base para realizar el cálculo de las ganancias, el sustento de esta propuesta es que el error proporciona información acerca del estado del sistema de forma implícita, por lo general se asume que un error elevado en un intervalo de tiempo pequeño implica un cambio de referencia o una

perturbación en el sistema, por lo que el controlador deberá generar la acción de control necesaria para compensar el error y alcanzar de nuevo la referencia.

Tomando esto en consideración, se propone el siguiente modelo para el cálculo de ganancias en función del error, tomando como base el valor de las ganancias obtenidos mediante la sintonización por métodos lineales, de modo que la ganancia estática del controlador PID se convierte en una ganancia variante en el tiempo [23].

$$K_p(t) = K_{p0} + K_{NL}|E| \quad (2.17)$$

Donde:

K_p es la ganancia proporcional variante en el tiempo

K_{p0} es la ganancia proporcional inicial, obtenida mediante reglas de sintonización

K_{NL} es el factor no lineal, una constante que se obtiene de forma experimental

E es el valor del error en un instante de tiempo

Si bien el modelo propuesto puede ser aplicado a las ganancias integral y derivativa también, por simplicidad se implementa el controlador PID en forma ideal, donde el valor de la ganancia proporcional influye en las demás ganancias [24], que tiene la siguiente fórmula:

$$m = K_p \left(E + \frac{1}{\tau_i} \int E dt + \tau_d \frac{dE}{dt} \right) \quad (2.18)$$

El controlador propuesto viene a ser una modificación del PID de ganancias estáticas, por lo que el proceso inicia con la obtención de los valores iniciales de las ganancias mediante reglas de sintonización lineales, utilizando los valores obtenidos en el apartado 2.4, donde se aproximó el sistema a uno de primer orden con retardo.

En lo que respecta al controlador que se va a sintonizar, en trabajos previos se ha visto el buen desempeño que tiene el controlador PI para sistemas de robótica móvil, por lo que se decide aplicar ese tipo de controlador. Las reglas de sintonización que se siguieron son las de Chien et al, que corresponden a las siguientes ecuaciones:

$$K_{p0} = \frac{0.6\tau}{k} \quad (2.19)$$

$$\tau_i = \tau \quad (2.19)$$

Una vez se hayan obtenido los valores de ganancia proporcional inicial tanto para el controlador de velocidad lineal como para el de velocidad angular, se procede a sintonizar el factor no lineal de ambos controladores de forma experimental, variando el factor no lineal y observando la respuesta a una entrada paso, para ello se utiliza el índice de rendimiento ISE, con cuyo valor se puede realizar el ajuste del factor, verificando que el índice disminuya con los cambios que se realicen.

Tabla 2.2. Sintonización de factor no lineal para controlador de velocidad angular

No. Prueba	K No lineal W	ISE
1	0.01	0.09259
2	0.05	0.09230
3	0.1	0.09207
4	0.5	0.08848
5	1	0.08497
6	2.12	0.07978

Tabla 2.3. Sintonización de factor no lineal para controlador de velocidad angular

No. Prueba	K No lineal U	ISE
1	0.01	0.08496
2	0.05	0.08426
3	0.1	0.08405
4	0.5	0.08241
5	1	0.07997
6	2.375	0.07493

Como se puede observar en las tablas 2.1 y 2.2, el incremento del factor no lineal disminuye el ISE en ambos casos, por lo que sería posible continuar elevando el valor hasta obtener el menor ISE posible, sin embargo, se debe considerar que el factor no lineal incrementa el valor de la ganancia proporcional e integral con el error, lo que produce también que la acción de control se eleve para el caso de tener errores grandes, por lo que se debe monitorear que este valor no supere los máximos recomendados por el fabricante, vistos en el apartado 2.1.

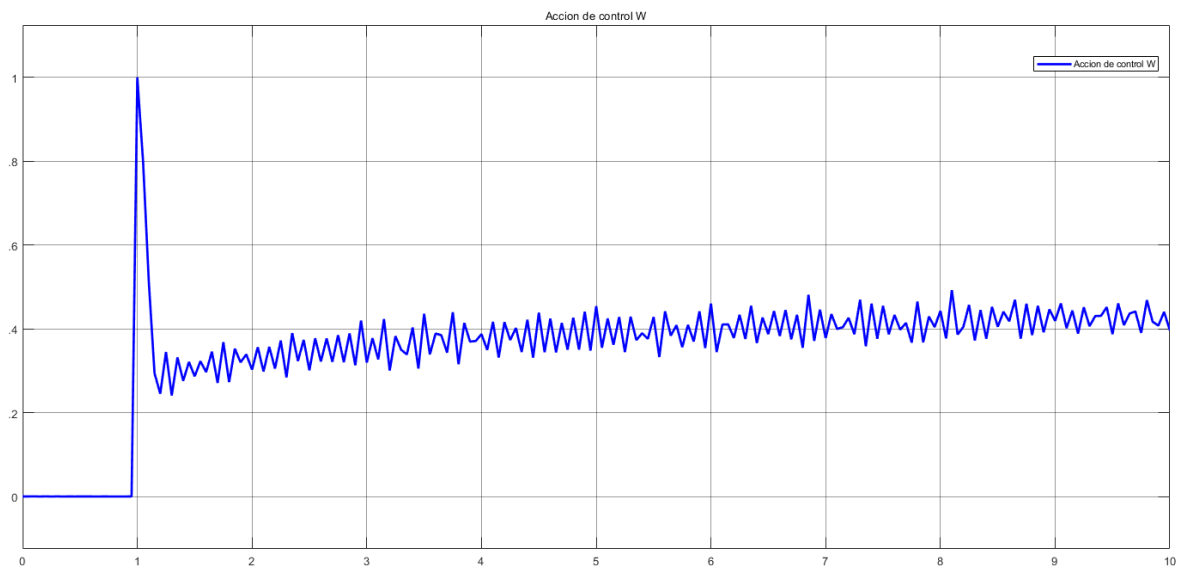


Figura 2.10. Acción de control de velocidad angular para el PI no lineal con factor de 2.12

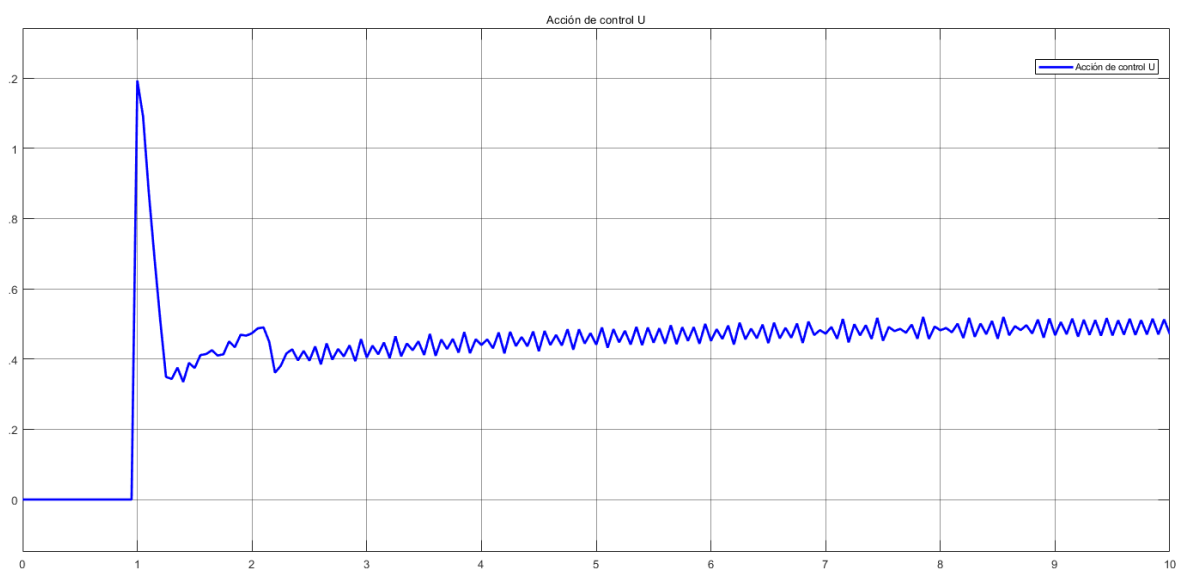


Figura 2.11. Acción de control de velocidad lineal para el PI no lineal con factor de 2.375

En las figuras 2.10 y 2.11 se puede ver como la acción de control generada por los controladores de velocidad lineal y angular alcanza los valores máximos recomendados por el fabricante para una entrada paso, lo que indica dos cosas, la primera es que el incremento del factor no lineal eleva la acción de control para cambios bruscos de referencia, por lo que se decide ocupar el último valor del factor en el que la acción de control alcanza los límites recomendados, siendo K_{NL} de 2.375 para la velocidad lineal, y de 2.12 para velocidad angular. La segunda cosa que nos indica el análisis de las acciones de control es que es necesario utilizar saturadores a la salida de ambos controladores, de

modo que cambios bruscos de referencia no lleguen a provocar una elevación de la acción de control que se salga de los valores máximos del Pioneer 3DX.

3 RESULTADOS, CONCLUSIONES Y RECOMENDACIONES

3.1 Resultados de simulación para trayectoria circular

En primer lugar, se implementa el controlador diseñado para una trayectoria circular, adecuada para observar el comportamiento del robot para referencias de velocidad lineal y angular constantes en la mayoría del trayecto. La posición inicial del robot es el origen del eje de coordenadas, mientras que el inicio de la trayectoria se encuentra en el punto $P_0(R,0)$, donde R es el radio del círculo, razón por la cual es necesario un período en el cual el robot alcance la referencia de trayectoria, en el cual las referencias de velocidad angular y lineal serán de amplitudes variables. Una vez terminado el estado transitorio, las velocidades lineal y angular serán constantes.

Las ecuaciones que describen la trayectoria circular son las siguientes:

$$x_{ref} = R\cos(\omega t) \quad (3.1)$$

$$y_{ref} = R\sin(\omega t) \quad (3.2)$$

$$\omega = \frac{V_{lineal}}{R} \quad (3.3)$$

Para la simulación realizada, la trayectoria tendrá un radio R de 3 metros, y una velocidad lineal de 0.3 m/s y una velocidad angular de 0.1 rad/s, en estado estable.

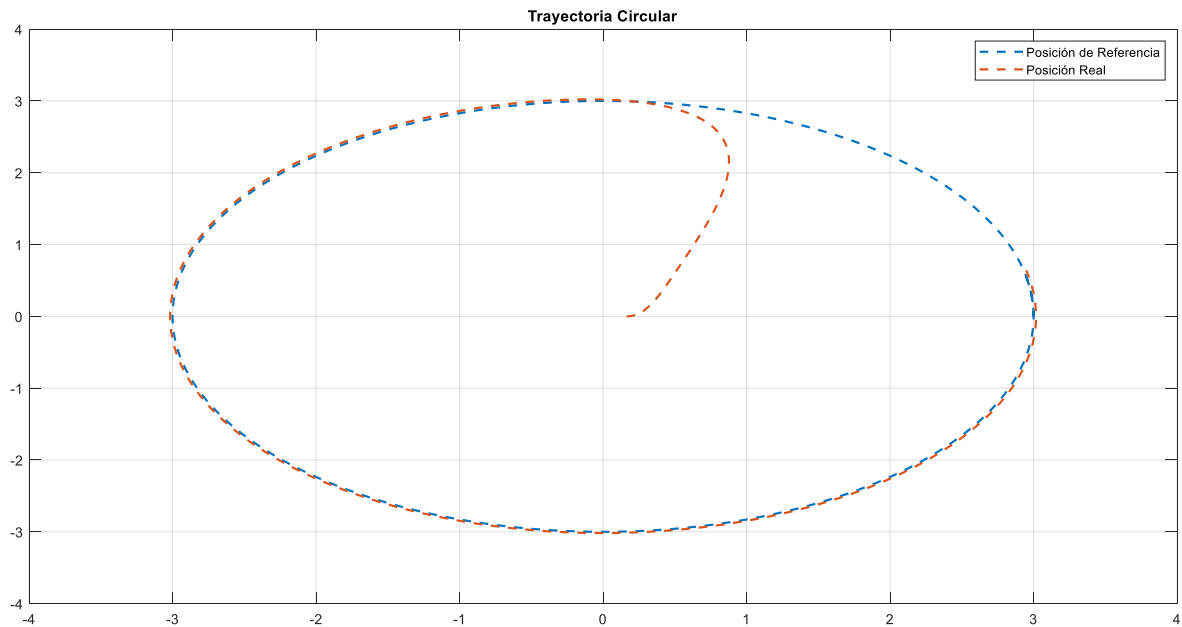


Figura 3.1. Seguimiento de trayectoria circular del robot Pioneer 3DX con controlador PI no lineal

Como se mencionó anteriormente, la simulación pasa por un estado transitorio en el que el robot alcanza la trayectoria desde su posición inicial, algo en lo que se profundizará posteriormente, sin embargo las primeras observaciones sugieren que en las etapas iniciales del recorrido, cuando el error de posición es el mayor, es en donde la parte no lineal del controlador PI tiene una mayor participación, presentando una respuesta más agresiva durante ese intervalo; mientras el robot se acerca más a la trayectoria la respuesta se suaviza debido a que el error disminuye y por ende el factor no lineal del controlador tiene valores más bajos.

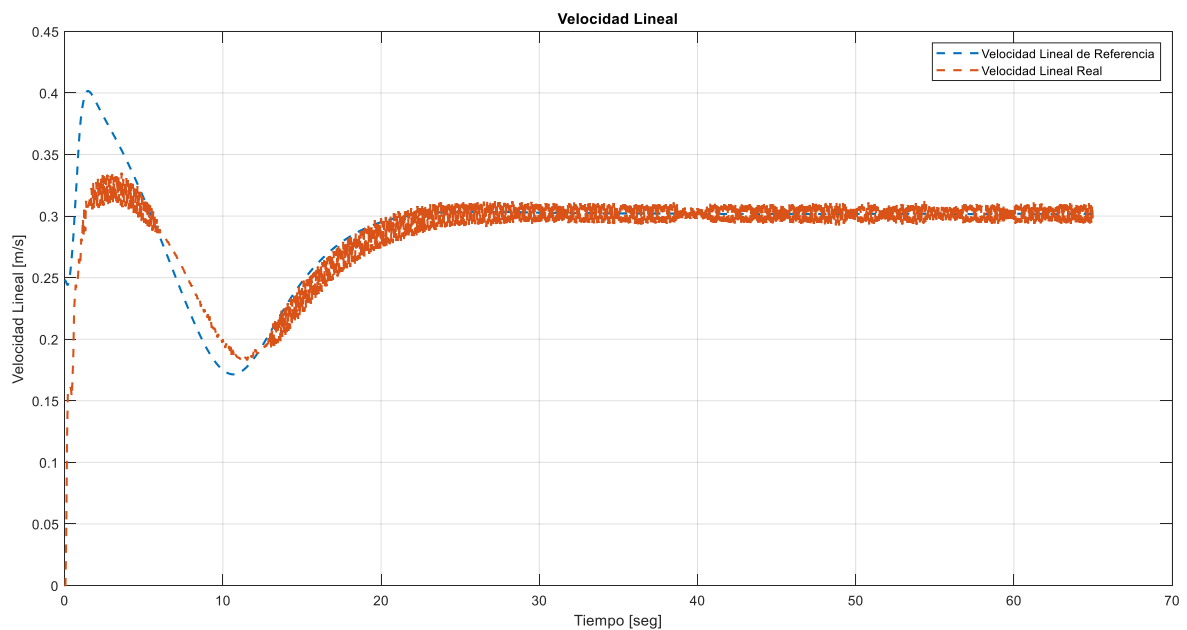


Figura 3.2. Referencia y respuesta de velocidad lineal durante la trayectoria circular

En la figura 3.2 se puede observar ciertas oscilaciones en la respuesta de velocidad lineal propias del controlador PI, las cuales tienen un valor máximo de 0.025 m/s. Es en esta parte donde se contemplan las variaciones de referencia en el estado transitorio de la simulación, en donde el robot tiene que alcanzar la trayectoria de referencia, los valores máximos pueden variar dependiendo del error de posición inicial (radio del círculo y posición inicial del robot), la agresividad del controlador cinemático visto previamente y la velocidad con la que se desea que se recorra la trayectoria. En el estado transitorio se pueden diferenciar dos escenarios, el primero ocurre entre los tiempos de 0 y 0.5 segundos, donde el error de velocidad lineal es elevado y el factor no lineal del controlador tiene un valor mayor, volviendo la respuesta más agresiva; y el segundo escenario se lo observa a partir del segundo 0.5, en donde el error ha disminuido significativamente y el factor no lineal tiene un menor peso en el controlador, suavizando la respuesta hasta que se alcanza el estado estable.

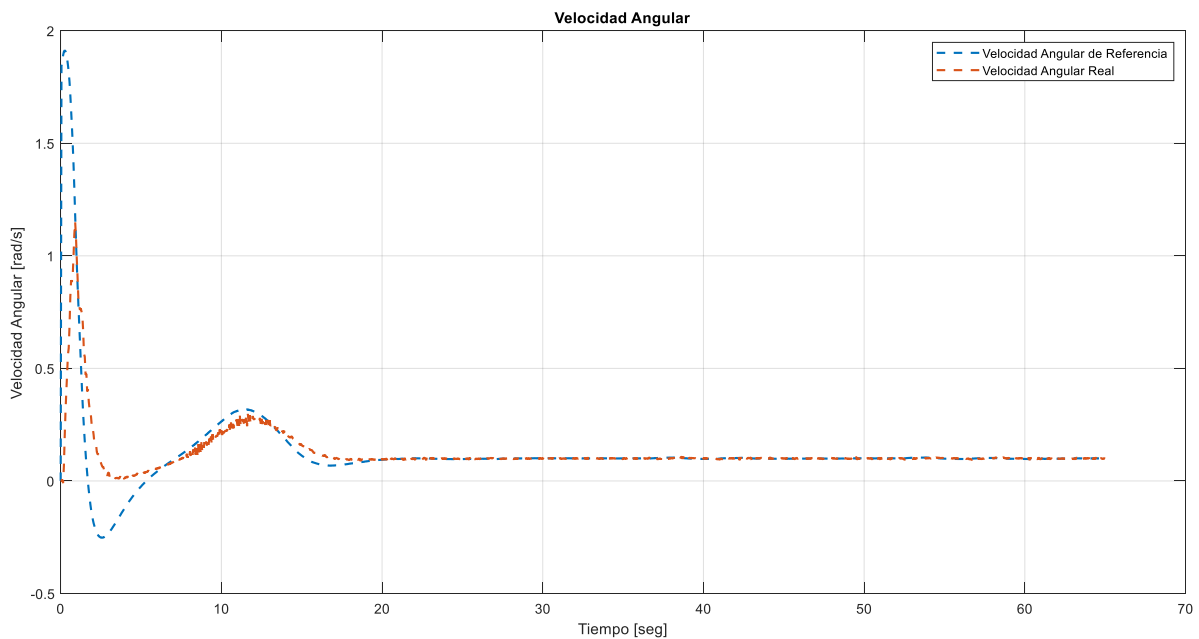


Figura 3.3. Referencia y respuesta de velocidad angular durante la trayectoria circular

En la figura 3.3 se observan resultados similares a los de la figura 3.2, en donde los intervalos de tiempo con mayor error presentan una respuesta más agresiva que los que se encuentran cercanos al estado estable, como es el caso del intervalo de tiempo comprendido en los 8 primeros segundos de simulación, que presentan respuestas más rápidas debido a los errores elevados. Cabe recalcar que en los primeros instantes de la simulación, cuando el error es el mayor, el controlador cinemático entrega una referencia de velocidad angular de 1.8rad/s, valor que se encuentra fuera de los límites constructivos del robot y que no debe ser superado, razón por la cual se utilizan saturadores para las acciones de control tanto en velocidad lineal como en velocidad angular; en este caso la velocidad angular se satura en el instante $t=2$ seg, momento en el cual el error disminuye y por ende la referencia entregada por el controlador cinemático baja a niveles normales de operación del robot.

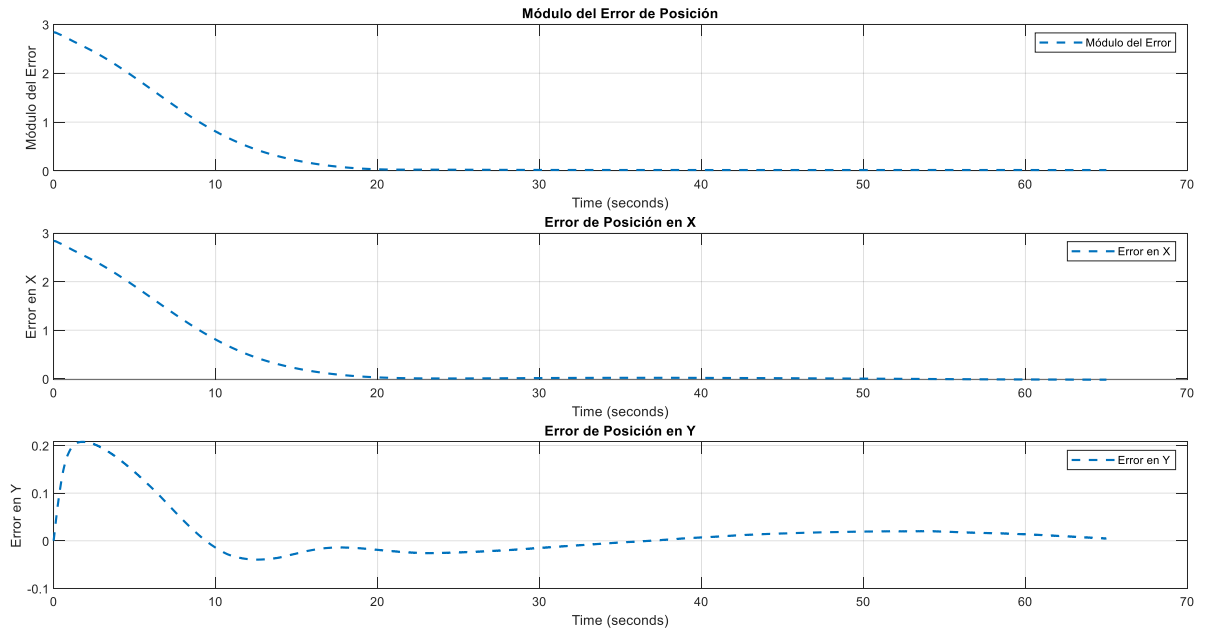


Figura 3.4. Errores de posición del Pioneer 3DX durante la trayectoria circular

En ésta parte se evidencia que existe una pequeña oscilación del error en el eje Y de un valor de 0.017 m. El controlador implementado tiene éxito en reducir los errores de los ejes X y Y a valores cercanos a 0 sin existir sobrepicos significativos. Finalmente, se observa el error inicial en el eje X en los primeros 8 segundos de la simulación, que es el intervalo de tiempo en el que el factor no lineal del controlador tendrá su mayor amplitud de todo el trayecto.

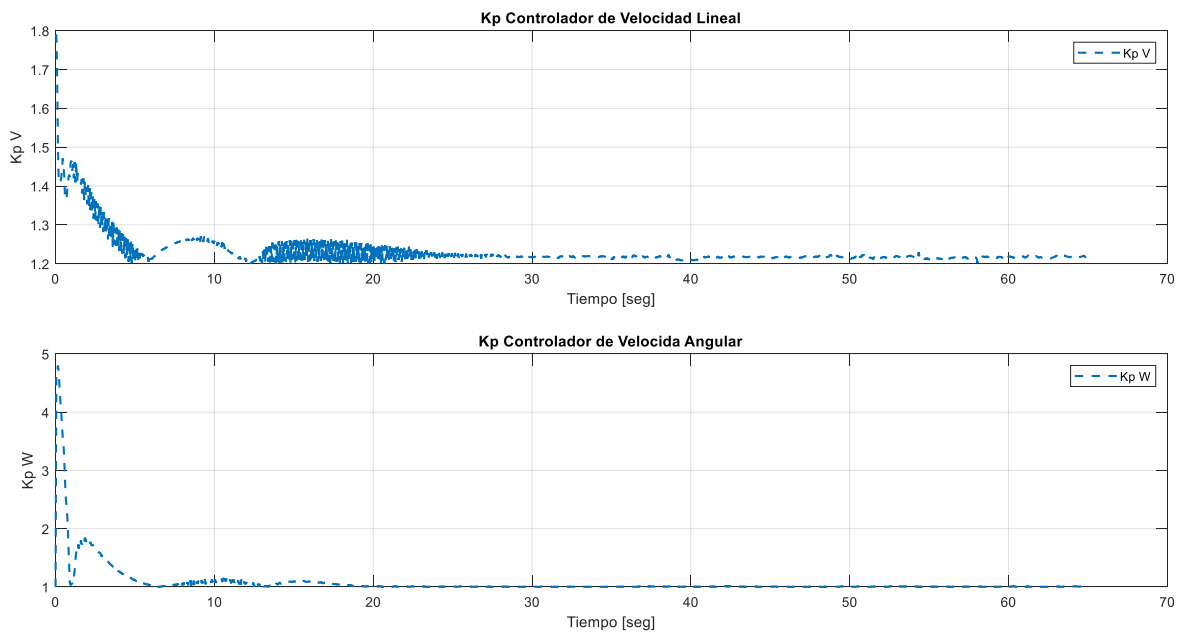


Figura 3.5. Evolución de los factores no lineales de los controladores de velocidad lineal y velocidad angular durante la trayectoria circular

En la figura 3.5 se presenta el valor que tiene la parte proporcional durante todo el trayecto, es aquí donde se evidencia que el estado transitorio con errores elevados donde el factor no lineal tiene mayor participación, en el caso del controlador de velocidad lineal llega a un valor de 1.78 en los intervalos iniciales, volviendo al valor mínimo de 1.2 cuando el sistema ha alcanzado el estado estable. De manera similar se comporta el controlador de velocidad angular, teniendo su parte proporcional un valor de 4.92 inicialmente para regresar a su valor mínimo en el estado estable.

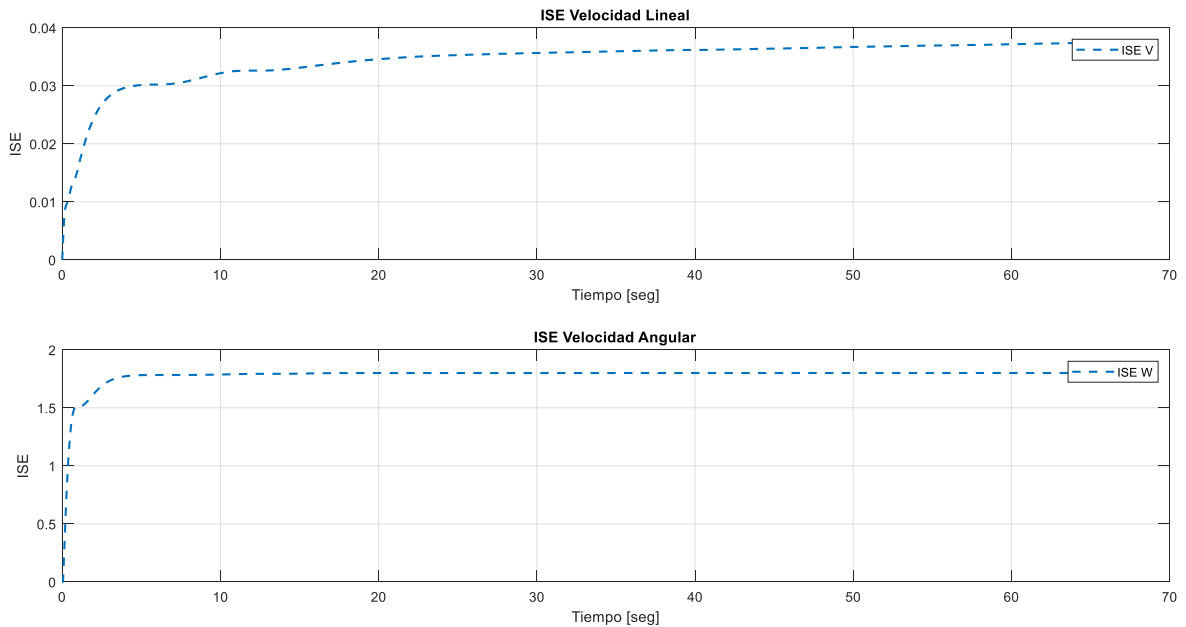


Figura 3.6. Evolución de factores de rendimiento ISE para velocidad lineal y angular en la trayectoria circular

La figura 3.6 presenta el índice de rendimiento ISE de ambos controladores implementados, el mayor crecimiento del error se evidencia en la etapa transitoria de la simulación. En el caso de la velocidad angular se muestra un crecimiento nulo del ISE cuando se alcanza el estado estable debido a un error muy cercano a 0, en la velocidad lineal en cambio existe un ligero crecimiento en el estado estable que se explica con las oscilaciones que presenta la respuesta.

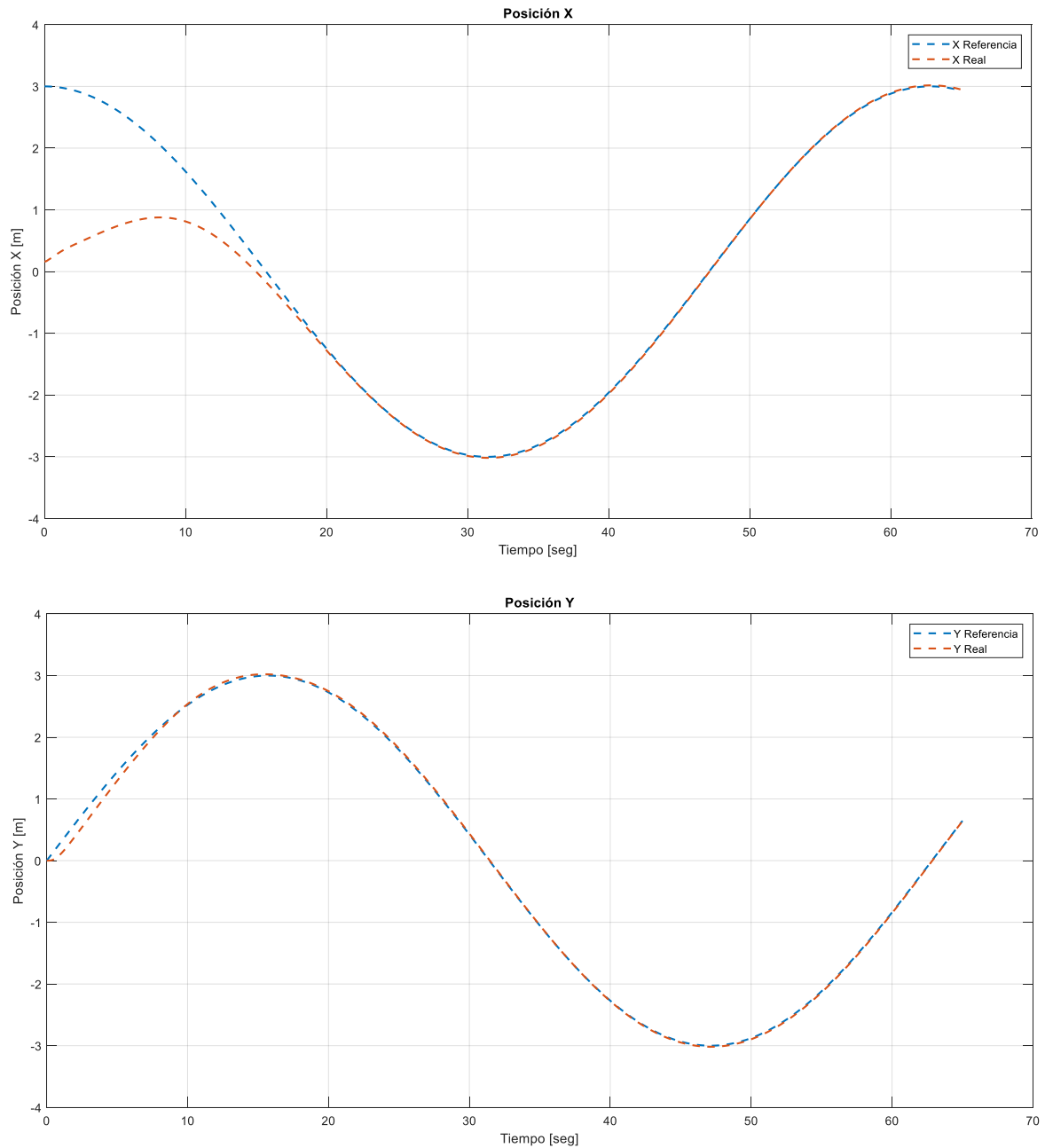


Figura 3.7. Posiciones reales y de referencia en X y Y para trayectoria circular

En la figura 3.7 se aprecia las evoluciones de la posición en X y Y tanto referenciales como reales, para la trayectoria circular de un radio de 3m, se puede observar de una forma más práctica los resultados de la figura 3.4 en donde el módulo del error se aproxima a 0 en el instante 17 seg. El mayor error se tiene en el eje X, por las razones expuestas previamente, sin embargo es allí donde se observa el trabajo del controlador cinemático que alcanza la referencia de forma satisfactoria. En cuanto al eje Y, si bien tanto la referencia como el robot inician en 0, el último se desvía brevemente debido a las acciones del controlador cinemático que compensa primero el eje X por su mayor error de posición, una vez el error en X se reduce, el robot vuelve a tomar la referencia en el eje Y. Finalmente, debido a la escala es casi imperceptible, sin embargo se comprueba que existe un pequeño error de posición en los puntos máximos y mínimos en la trayectoria del eje Y.

3.2 Resultados de simulación para trayectoria lemniscata

La segunda trayectoria implementada se trata de la Lemniscata de Bernoulli, en donde se busca poner a prueba los controladores diseñados ante cambios de orientación del robot tanto crecientes como decrecientes, manteniendo una velocidad lineal relativamente constante.

Las ecuaciones que describen la trayectoria Lemniscata son las siguientes:

$$x_{ref} = \frac{R \cos(\omega t)}{1 + \sin^2(\omega t)} - \frac{R}{2} \quad (3.4)$$

$$y_{ref} = \frac{R \sin(\omega t) \cos(\omega t)}{1 + \sin^2(\omega t)} \quad (3.5)$$

Los parámetros que definen a la trayectoria Lemniscata son similares a la circular, un radio de 3m, una velocidad lineal de 0.3m/s y velocidad angular de 0.1rad/s.

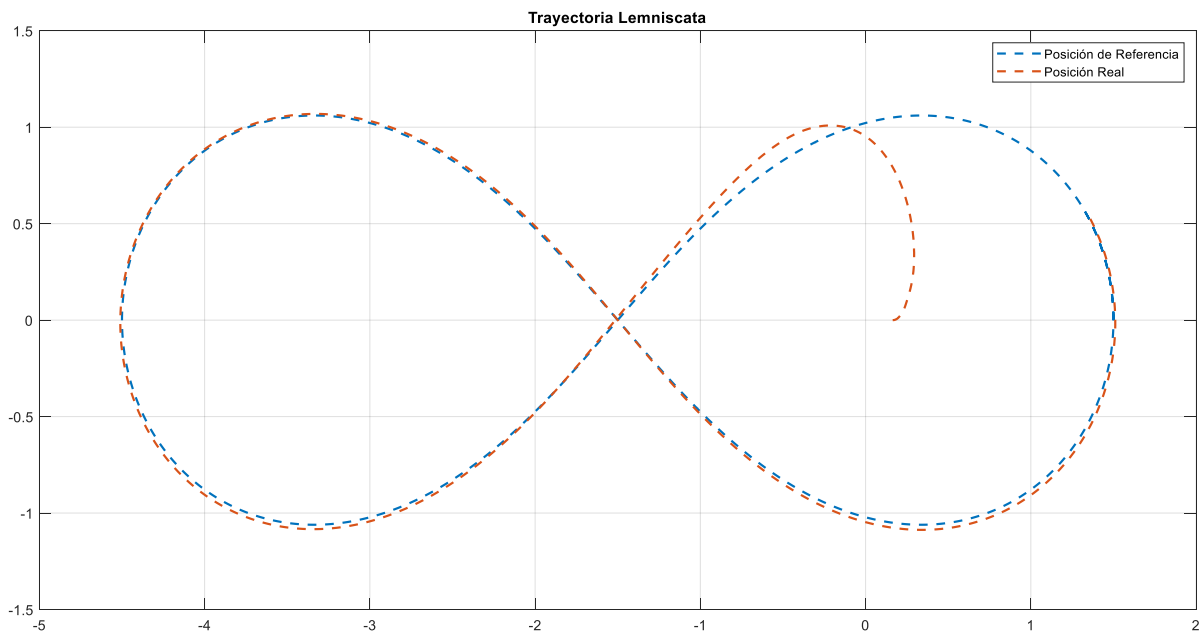


Figura 3.8. Seguimiento de trayectoria Lemniscata del robot Pioneer 3DX con controlador PI no lineal

La figura 3.8 muestra un efectivo seguimiento de la trayectoria Lemniscata, que inicia desde el centro de uno de los lóbulos de la misma. En éste caso se puede resaltar que inicialmente, la trayectoria tiene un cambio de orientación ligeramente más brusco que la trayectoria circular, lo que da como resultado el error de posición existente en el primer lóbulo.

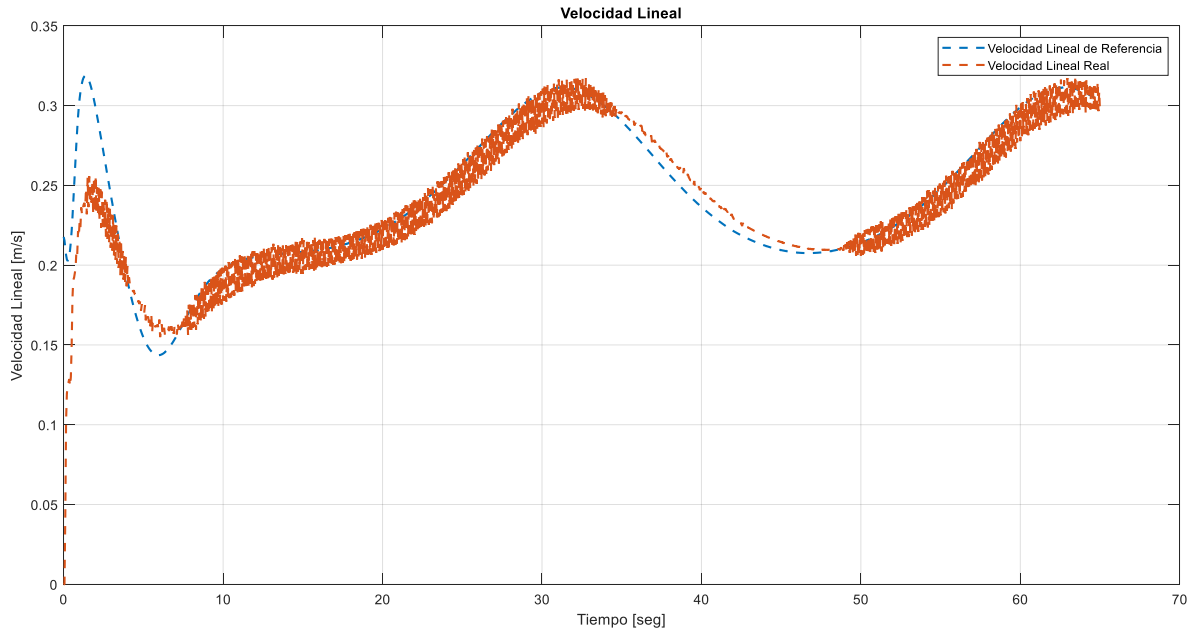


Figura 3.9. Referencia y respuesta de velocidad lineal durante la trayectoria lemniscata

En la figura 3.9 se aprecia que el controlador presenta oscilaciones de 0.025 m/s especialmente durante los intervalos crecientes de la velocidad lineal. Nuevamente se observa una respuesta más agresiva en los primeros instantes debido al elevado error entre las posiciones inicial del robot y el comienzo de la trayectoria, sin embargo en el caso de la Lemniscata el error inicial es menor que el de la circular (el radio de 3m se divide entre ambos lóbulos de la trayectoria) por lo que la acción de control que entrega el controlador cinemático es igualmente menor.

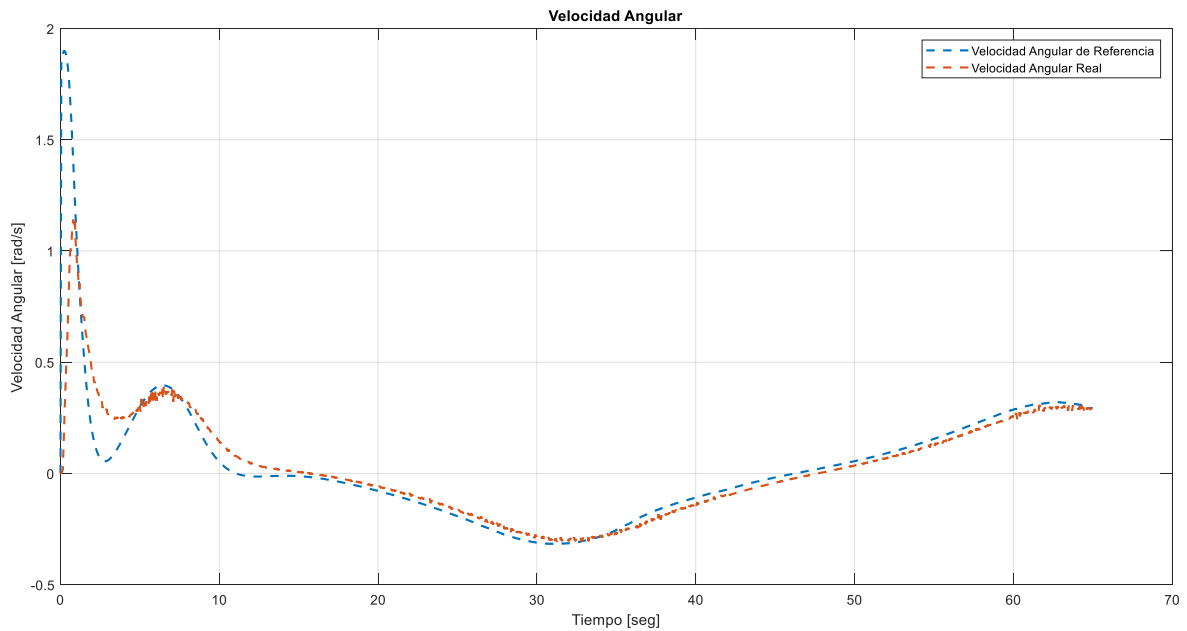


Figura 3.10. Referencia y respuesta de velocidad angular durante la trayectoria lemniscata

Los cambios de orientación en la trayectoria lemniscata se observan de mejor manera en la velocidad angular, tal y como lo muestra la figura 3.10 que posteriormente a los instantes iniciales de la simulación (el tiempo que tarda el robot en tomar la trayectoria) lleva intervalos regulares y simultáneos de velocidad angular creciente y decreciente, en este caso el controlador posee un error de velocidad de que da como resultado un error de posición en el eje Y que se muestra en figuras posteriores. El mismo que se puede explicar con el módulo de error de velocidad angular que en estado estable es bajo, por lo que el factor no lineal no tiene un peso relevante en el controlador.

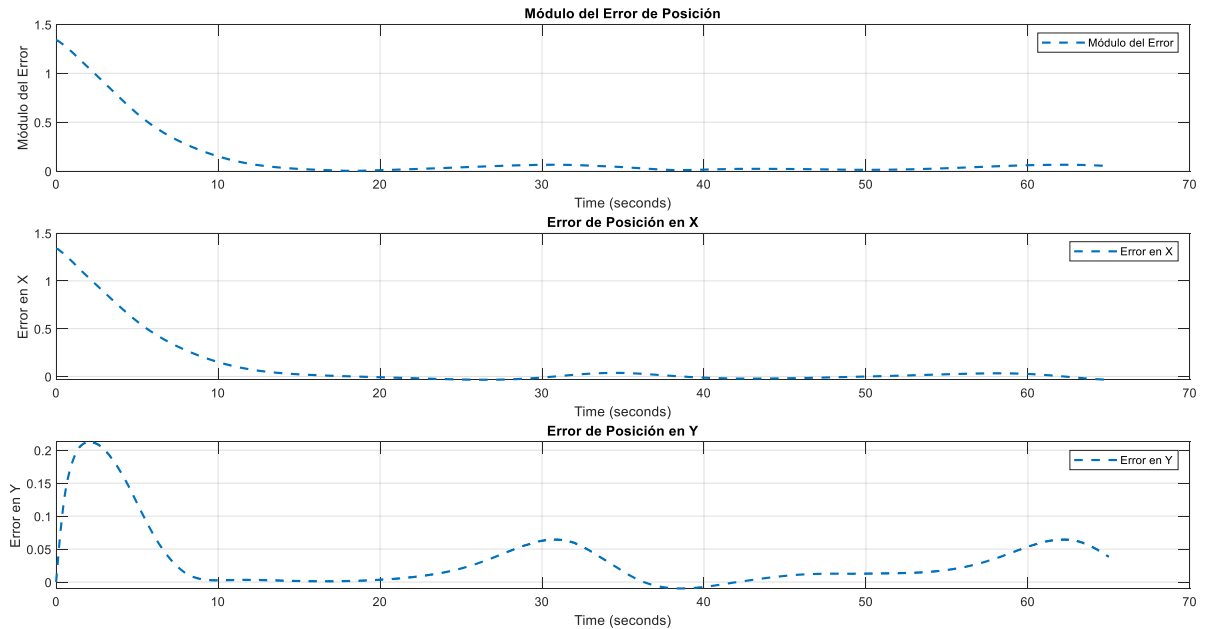


Figura 3.11. Errores de posición del Pioneer 3DX durante la trayectoria lemniscata

Los errores de posición durante la trayectoria Lemniscata se muestran en la figura 3.11, se aprecia cómo el robot alcanza la trayectoria en aproximadamente 15 segundos, momento en el que el módulo del error se aproxima a 0, sin embargo, también se puede ver el error de posición en el eje Y que se mencionó previamente, el cual alcanza su valor máximo alrededor del instante 30seg., repitiéndose de forma regular posteriormente. El crecimiento del error de posición en Y se debe al error de velocidad angular presente en la figura 3.10, sin embargo, el error se mantiene en valores aceptables, con un valor máximo de 0.06m.

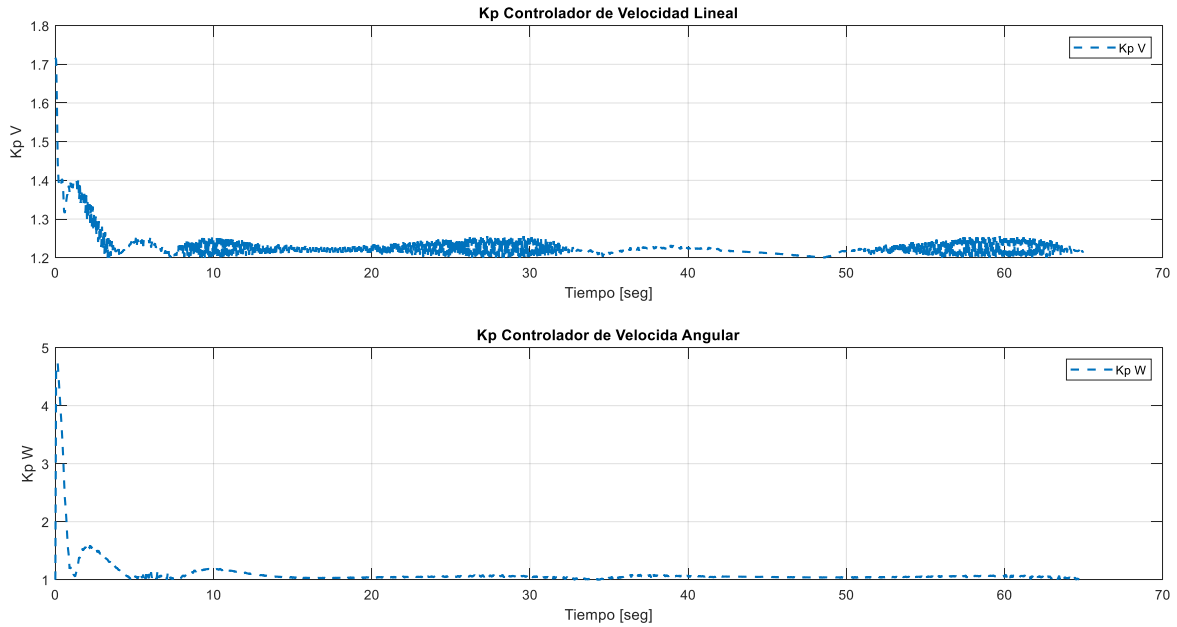


Figura 3.12. Evolución de los factores no lineales de los controladores de velocidad lineal y velocidad angular durante la trayectoria lemniscata

La figura 3.12 muestra los valores que toma el factor proporcional de los controladores de velocidad lineal y angular durante la trayectoria Lemniscata, con un comportamiento similar al presentado durante la trayectoria circular, en donde el factor no lineal tiene un valor elevado en el inicio de la simulación, cuando el error es elevado y por ende el controlador presenta una respuesta agresiva que se suaviza progresivamente mientras el robot alcanza las referencias y el error disminuye. En el factor del controlador de velocidad angular se observa que, a partir del instante 15seg., el error es bajo, por lo que las acciones de control no se elevan como en los primeros instantes, razón por la cual existe el error de velocidad angular.

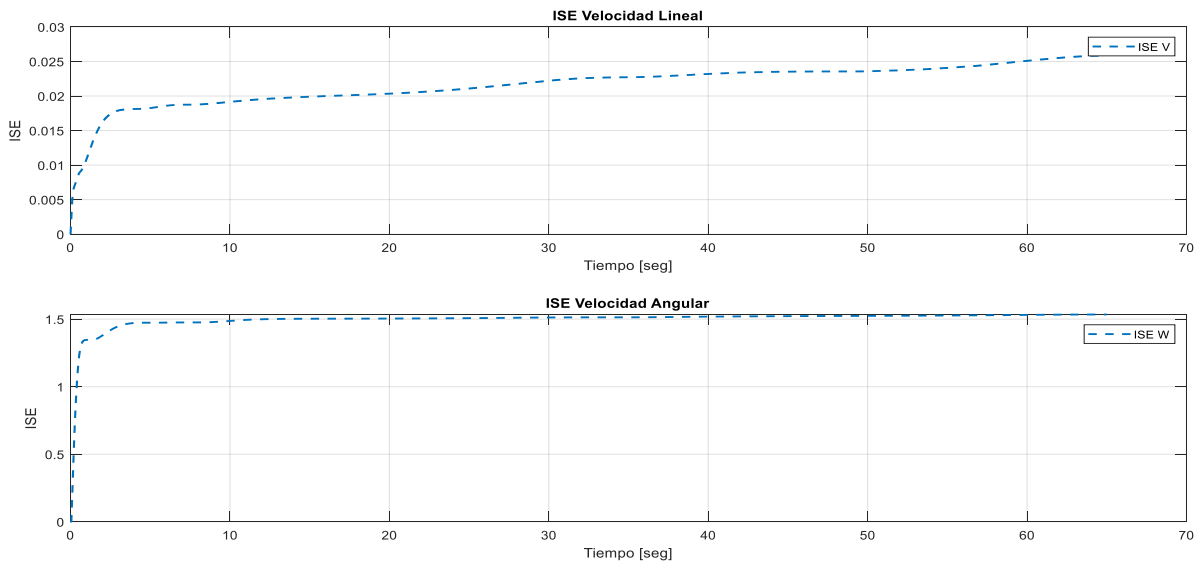


Figura 3.13. Evolución de factores de rendimiento ISE para velocidad lineal y angular en la trayectoria lemniscata

En lo que a los índices de rendimiento respectan, el ISE de la trayectoria Lemniscata tiene un comportamiento similar a la trayectoria circular en el que el mayor crecimiento se produce durante el inicio de la simulación, la principal diferencia con la figura 3.6 es un mayor crecimiento del ISE en intervalos regulares de aproximadamente 30 segundos, esto se debe al error que se produce tanto en velocidad lineal como angular en los períodos crecientes/decrecientes que caracterizan a la trayectoria Lemniscata, lo que produce que el ISE de velocidad lineal llegue al valor de 0.0266 y el de velocidad angular llegue a 1.52.

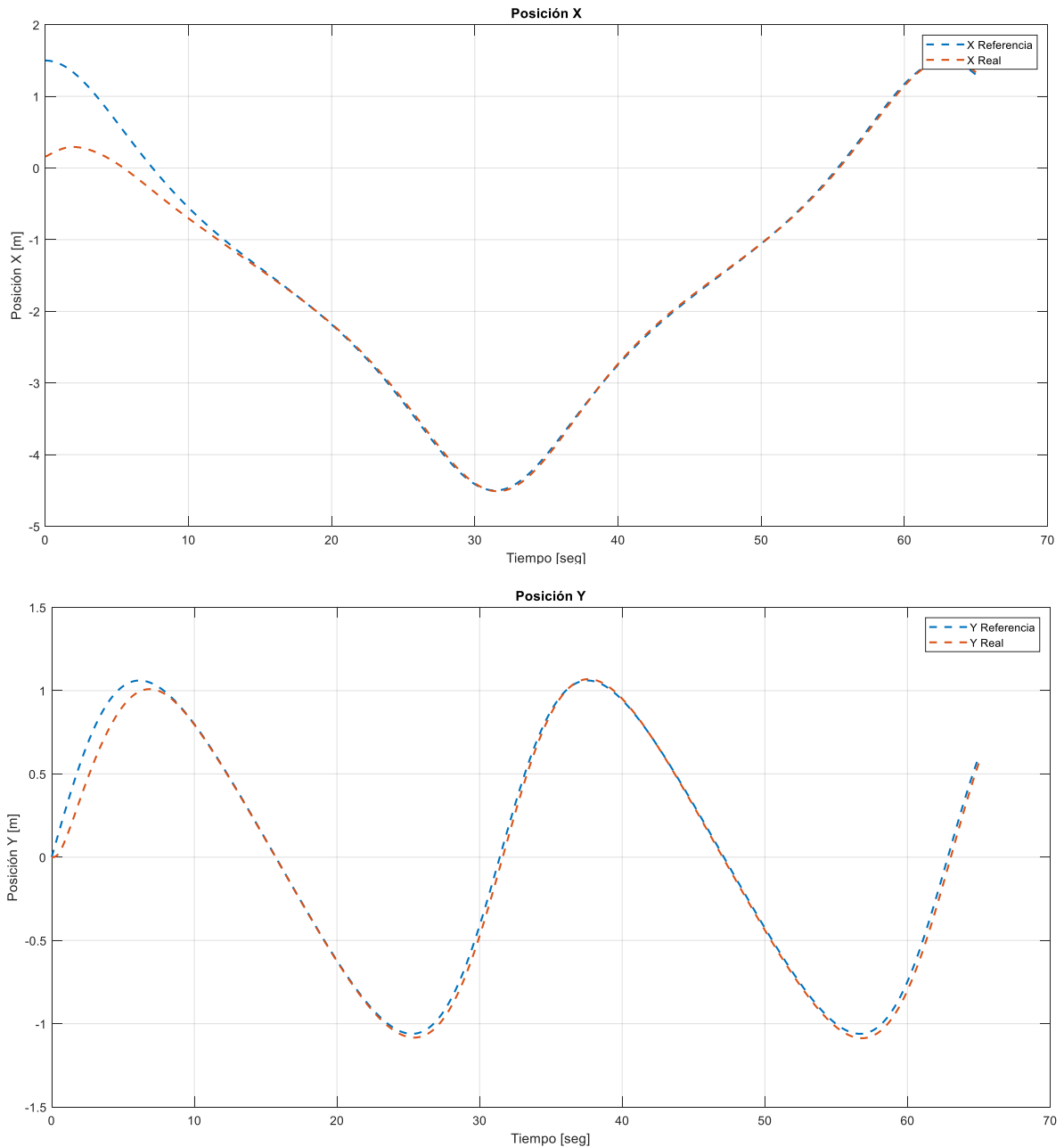


Figura 3.14. Posiciones reales y de referencia en X y Y para trayectoria lemniscata

Finalmente en la figura 3.14 se muestra el seguimiento de las posiciones de X y Y en el tiempo, se puede apreciar que el error inicial en el eje X es efectivamente la mitad del radio de la Lemniscata, lo que corresponde al radio de uno de los lóbulos. De forma similar a la trayectoria circular, ocurre que en el eje Y se produce un error de posición inicial debido a

las acciones del controlador cinemático para tomar la referencia en X, ambos errores se reducen significativamente tras 15 segundos de simulación, que es el momento en el que el robot alcanza la trayectoria en ambos ejes. En el eje Y se puede observar el error de posición mencionado previamente, sobre todo en los intervalos de decrecimiento de la posición Y.

3.3 Resultados de simulación para trayectoria rectangular

La trayectoria cuadrada permite probar los controladores ante cambios bruscos de la orientación del robot, los que se observan en las esquinas de la trayectoria.

Las ecuaciones que describen la trayectoria Lemniscata son las siguientes:

Para el primer intervalo:

$$x_{ref\ 1} = V_{lineal}t \quad (3.6)$$

$$y_{ref\ 1} = 0 \quad (3.7)$$

Para el segundo intervalo:

$$x_{ref\ 2} = l \quad (3.8)$$

$$y_{ref\ 2} = -l + V_{lineal}t \quad (3.9)$$

Para el tercer intervalo:

$$x_{ref\ 3} = 3l - V_{lineal}t \quad (3.10)$$

$$y_{ref\ 3} = l \quad (3.11)$$

Para el cuarto intervalo:

$$x_{ref\ 4} = 0 \quad (3.12)$$

$$y_{ref\ 4} = 4l - V_{lineal}t \quad (3.13)$$

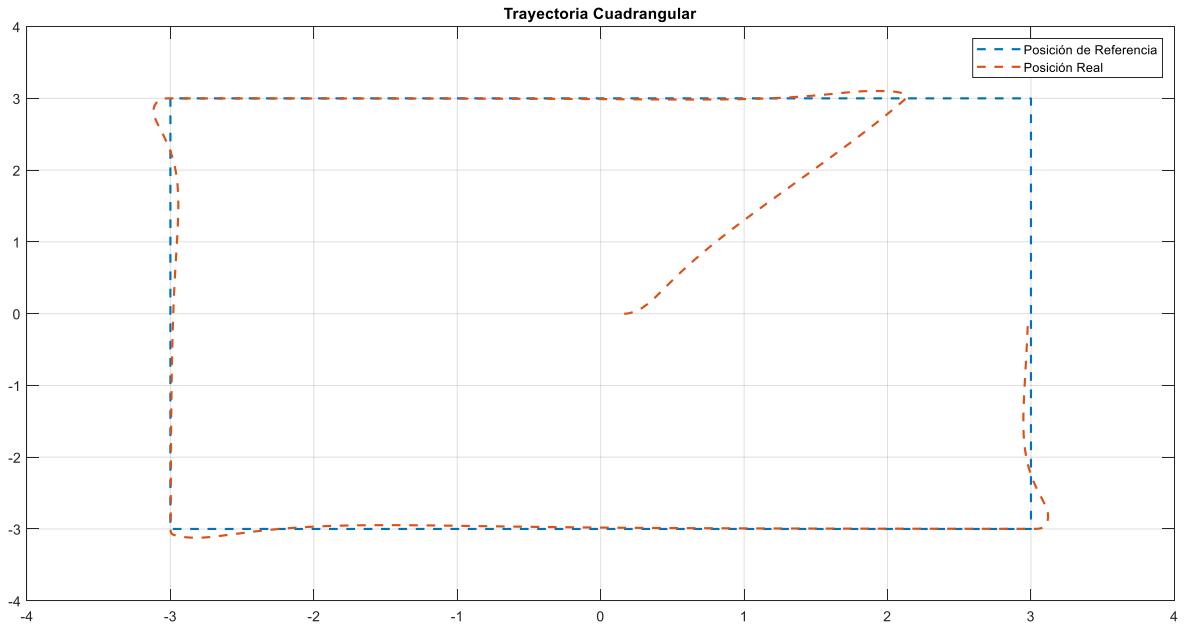


Figura 3.15. Seguimiento de trayectoria cuadrada del robot Pioneer 3DX con controlador PI no lineal

En la figura 3.15 se observan los resultados del controlador implementado para la trayectoria cuadrada. Lo primero que salta a la vista es que esta trayectoria presenta errores de posición mucho más marcados que las dos anteriores, pues la trayectoria cuadrangular presenta cambios bruscos en la orientación del robot en las esquinas.

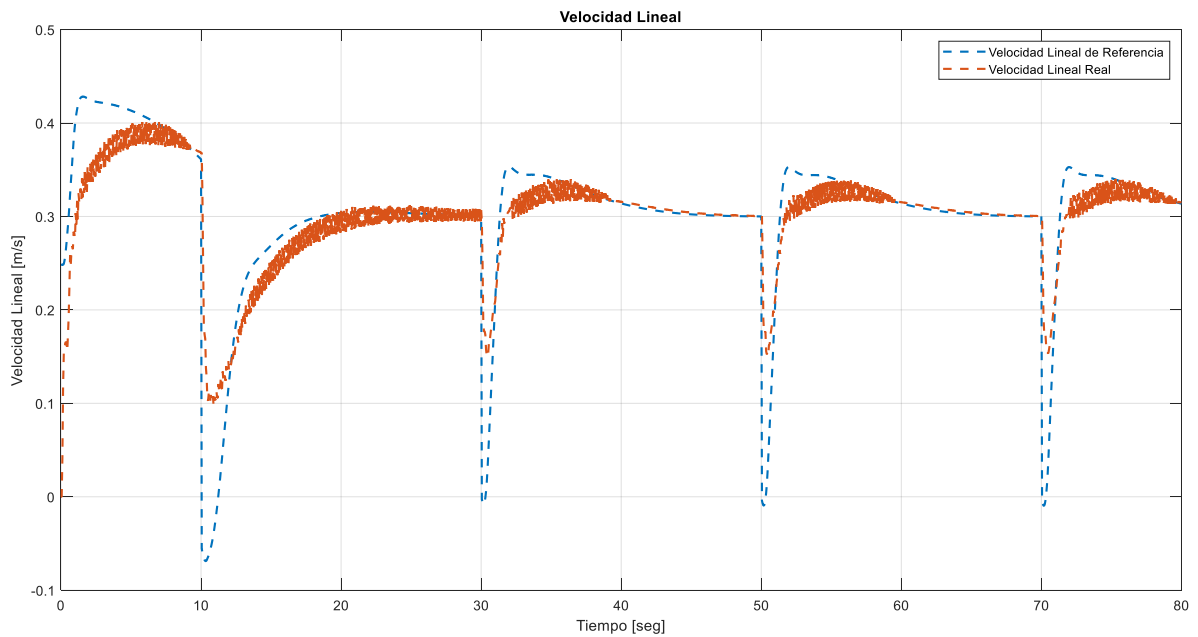


Figura 3.16. Referencia y respuesta de velocidad lineal durante la trayectoria cuadrada

En la gráfica 3.16 se muestra la referencia y respuesta de velocidad lineal a lo largo de la trayectoria. Se puede ver como la velocidad lineal de referencia decae bruscamente en los tiempos 10, 30, 50 y 70 seg., que corresponden a los tiempos en los que el robot alcanza las esquinas del cuadrado. La velocidad lineal de referencia disminuye su valor debido a que se busca que el robot desacelere durante un instante para poder realizar el cambio de orientación de la esquina. Cuando el cambio se efectúa en su mayor parte, el robot retoma el trayecto con su velocidad lineal de 0.3m/s, produciéndose errores de posición en las esquinas del cuadrado.

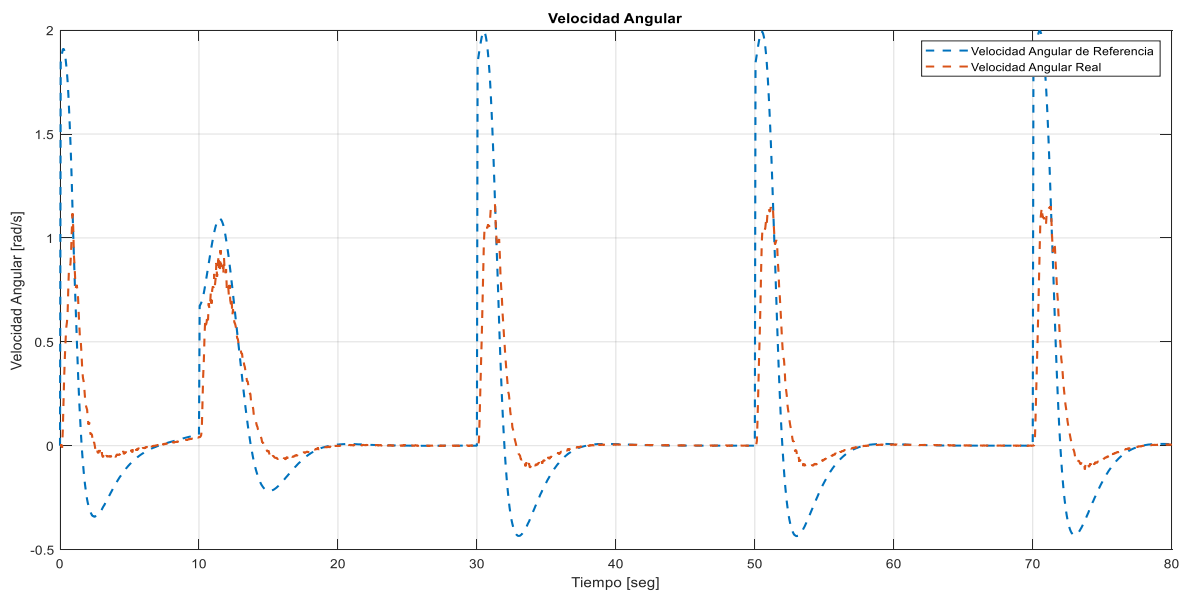


Figura 3.17. Referencia y respuesta de velocidad angular durante la trayectoria cuadrada

En la figura 3.17 aparece la velocidad angular, tanto de referencia como real del robot durante todo el trayecto. Se puede observar que, al igual que en la velocidad lineal, existen cambios bruscos en los tiempos 10, 30, 50, 70 seg. En este caso se denotan crecimientos bruscos, los cuales se explican que, cuando el robot alcanza las esquinas del cuadrado, existe un cambio brusco de orientación del robot, el cual se logra con una velocidad angular elevada hasta que el robot alcance la orientación deseada. Las referencias de velocidad angular negativas se producen cuando el robot completa el cambio de orientación en las esquinas del cuadrado, debido a lo abrupto del cambio existe un error de orientación de que debe ser compensado orientando al robot hacia el lado opuesto, lo cual se logra con un intervalo de velocidad angular negativo.

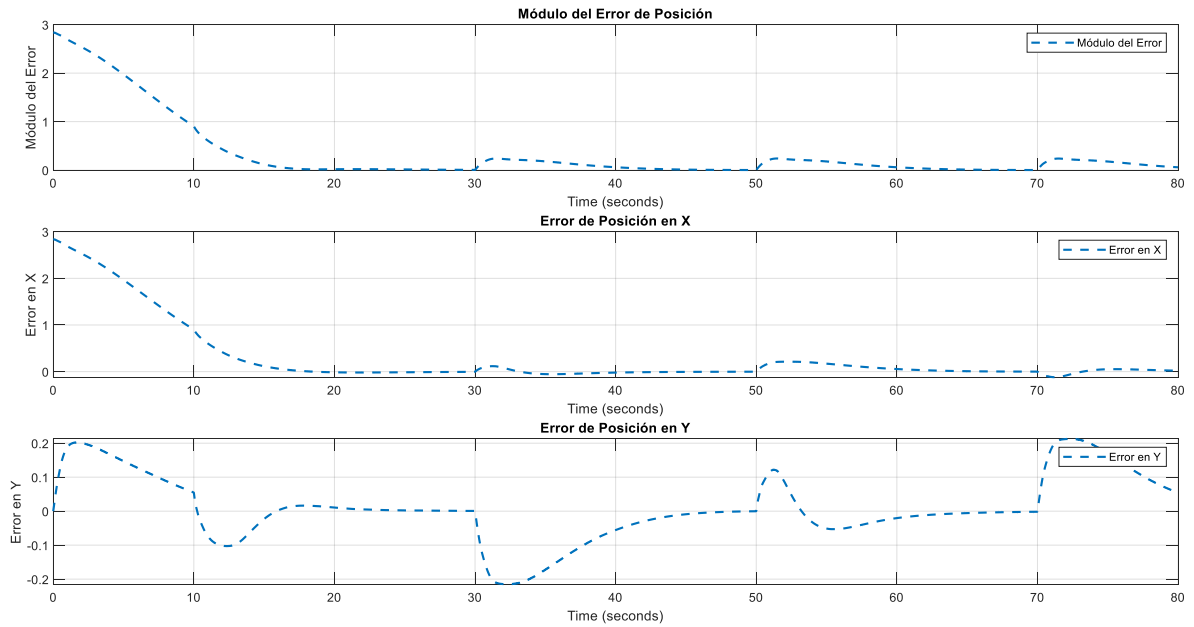


Figura 3.18. Errores de posición del Pioneer 3DX durante la trayectoria cuadrada

En la figura 3.18 se aprecia la evolución del error de posición a lo largo de la trayectoria cuadrangular, nuevamente el mayor error aparece al inicio del recorrido, cuando el módulo del mismo es de la mitad del lado del cuadrado, el módulo inicial del error disminuye progresivamente conforme el robot alcanza la trayectoria hasta que alcanza un valor cercano al nulo, tal y como ocurría en las dos trayectorias anteriores. La mayor diferencia de la trayectoria cuadrada aparece cuando el robot alcanza las esquinas y tiene que disminuir su velocidad lineal para realizar el cambio de orientación que exige la trayectoria, lo que genera un aumento en el módulo del error de un valor de 0.125 m. Cuando se completa el cambio de orientación de las esquinas, el robot nuevamente acelera linealmente hasta alcanzar a la referencia, reduciéndose el error de posición nuevamente. El ciclo se repite cada vez que el robot llegue a una esquina del cuadrado.

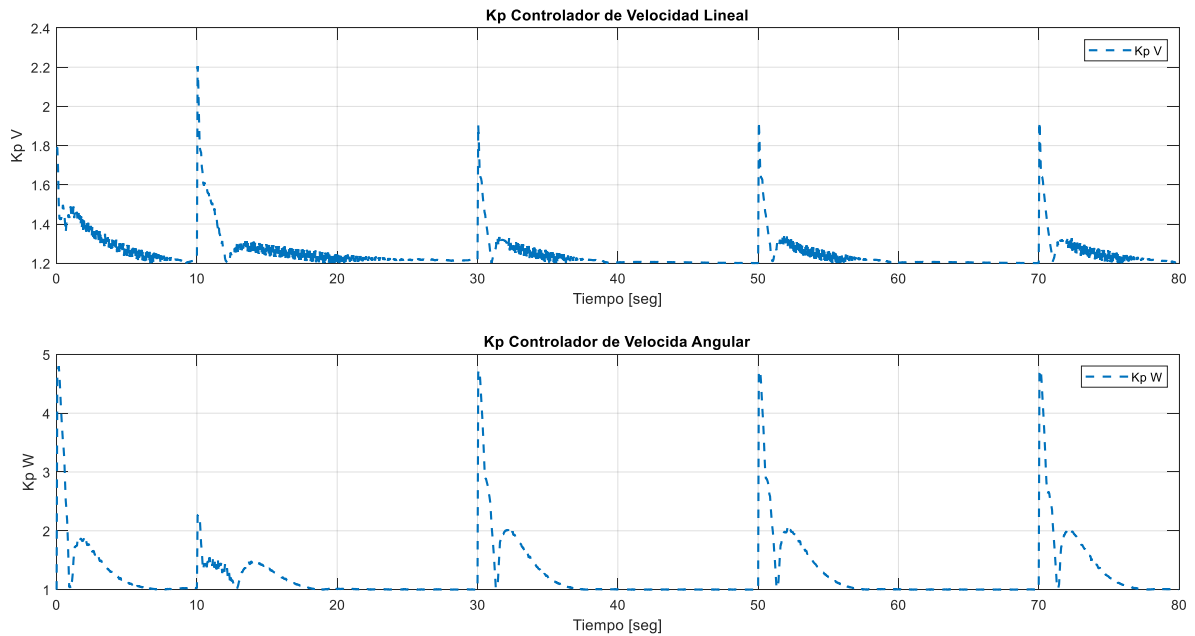


Figura 3.19. Evolución de los factores no lineales de los controladores de velocidad lineal y velocidad angular durante la trayectoria cuadrada

En la figura 3.19 se aprecia la evolución del factor no lineal del controlador de velocidad lineal y de velocidad angular a lo largo del trayecto. De forma similar a las anteriores trayectorias, el factor no lineal tiene un mayor valor en los primeros instantes, cuando el error es el mayor, alcanzando un valor de 2.2 para velocidad lineal y 4.89 para velocidad angular, a medida que disminuye el error de posición, el factor vuelve a su valor inicial. La diferencia radica en las esquinas del cuadrado, cuando los errores de velocidad lineal y angular se elevan nuevamente, produciendo un aumento en ambos factores no lineales hasta que el robot alcance nuevamente las referencias de velocidad.

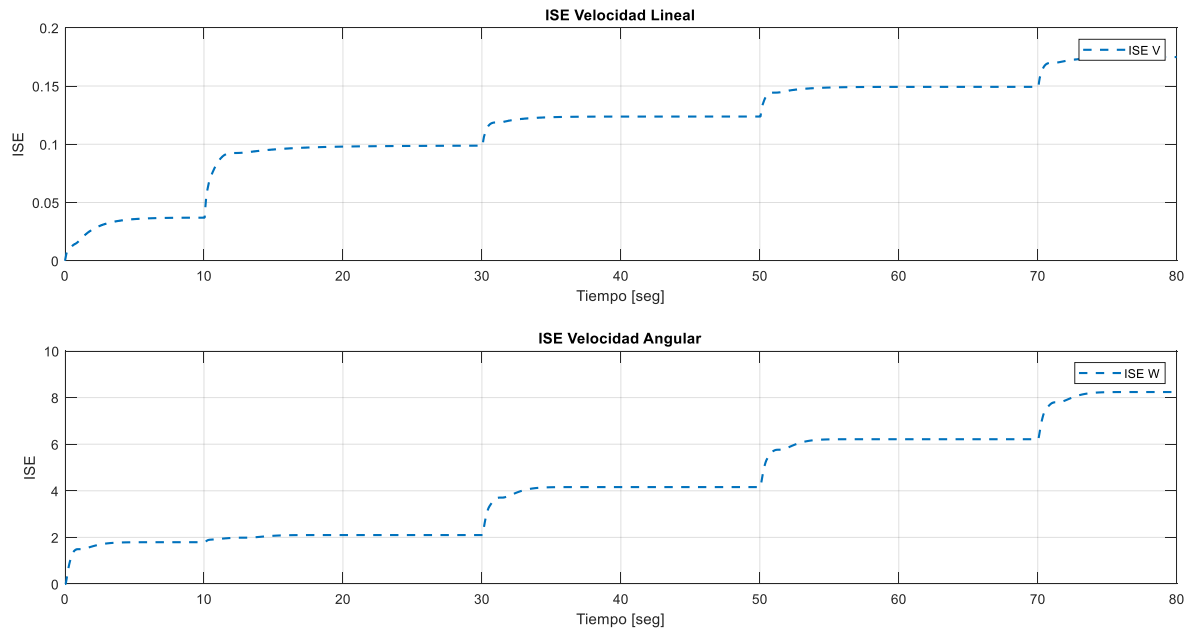


Figura 3.20. Evolución de factores de rendimiento ISE para velocidad lineal y angular en la trayectoria cuadrada

La evolución de los factores de rendimiento ISE se muestra en la figura 3.20, en donde se aprecian elevaciones del valor del error de forma periódica, las cuales ocurren tanto en el inicio del recorrido, como en las esquinas del cuadrado, que son los puntos en los que los errores se elevan, sin embargo, se observa como el ISE se estabiliza aproximadamente 17 seg. después de alcanzar la esquina del cuadrado, lo que es el tiempo que tarda el robot en alcanzar de nuevo la referencia.

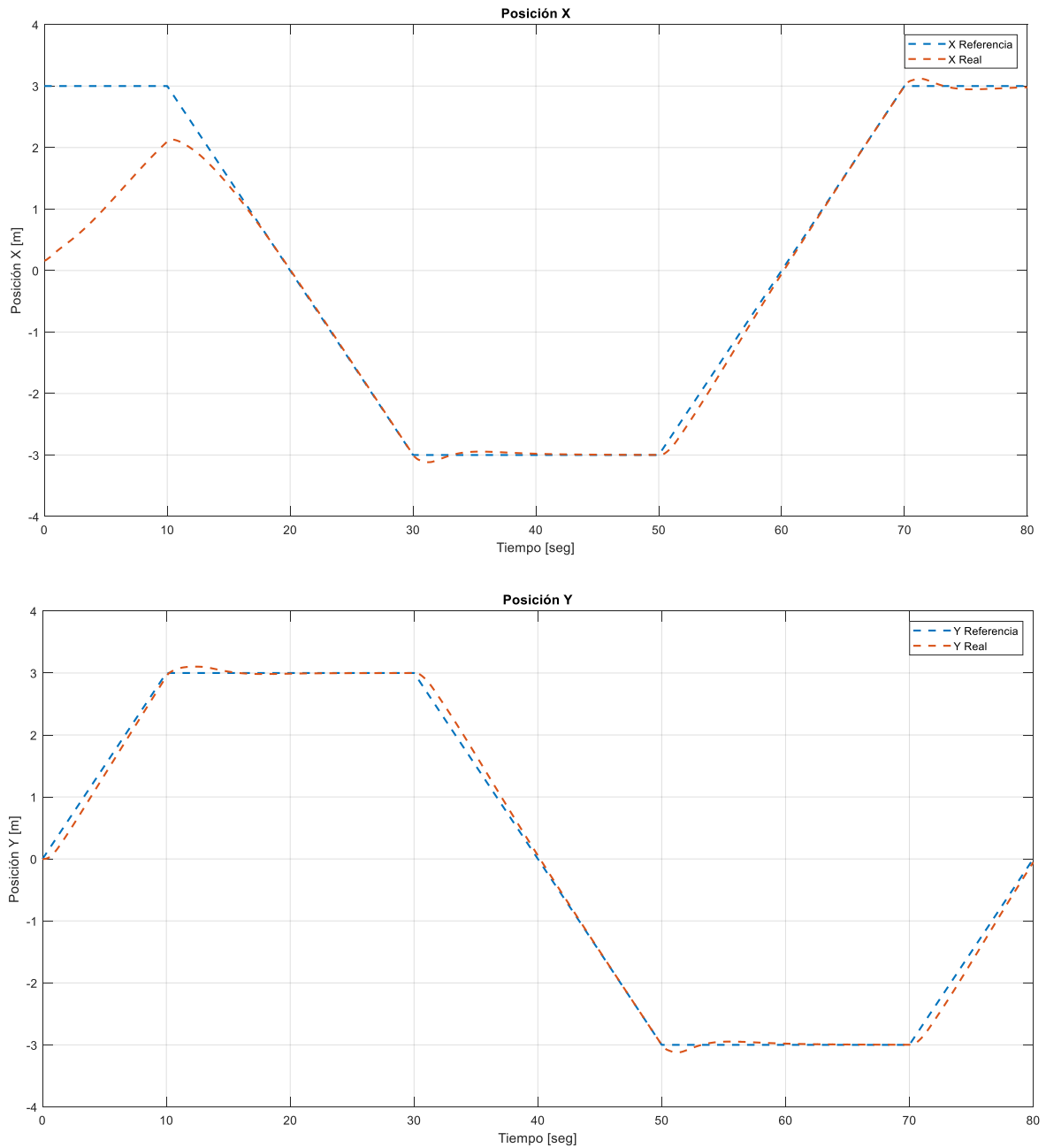


Figura 3.21. Posiciones reales y de referencia en X y Y para trayectoria cuadrada

Finalmente, en la figura 3.21 se aprecian las posiciones en X y Y para la trayectoria cuadrada, tanto la referencia como la real del robot, se pueden ver los cambios bruscos de orientación en las partes crecientes y decrecientes de las referencias en X y Y. En este caso se observan sobrepicos de aproximadamente 0.125 m. que se producen en las esquinas del cuadrado. Lo que ocurre es que, la referencia de velocidad lineal disminuye de forma abrupta, pero el robot no frena inmediatamente, sino que disminuye su velocidad, lo que produce que el robot sobrepase la referencia hasta que cambie su orientación, produciéndose así el sobrepico en las esquinas del cuadrado.

3.4 Comparación de resultados de PID no lineal frente a PID tradicional

Como se pudo determinar en los ítems anteriores del capítulo 3, un controlador PID no lineal incrementa sus factores de control cuando se tiene un error mayor, por lo general en el estado transitorio, mientras que en el estado estable regresa sus factores aproximadamente a los que tendría un PID tradicional.

Para realizar una comparación entre ambos controladores, se considera la trayectoria cuadrada que es la que presenta una mayor cantidad de cambios bruscos en la orientación del robot, que implican mayor cantidad de estados transitorios con errores iniciales relativamente elevados, y que por ende muestra de mejor manera las diferencias que tiene un PID no lineal frente a uno tradicional.

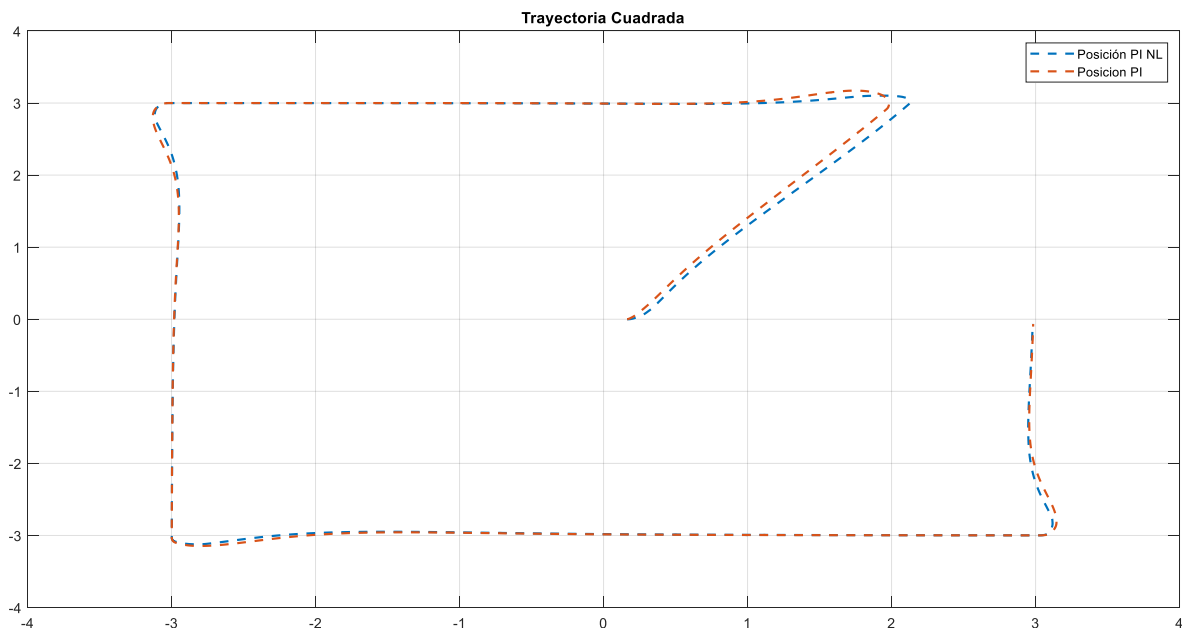


Figura 3.22. Comparación de seguimiento de trayectoria de PI No Lineal y PI Tradicional

En la figura 3.22 se evidencia que el controlador PI no lineal tiene una respuesta más agresiva en el estado transitorio que el PI tradicional, razón por la cual alcanza la trayectoria antes. Una vez se encuentra en estado estable sobre la trayectoria, los factores del PI no lineal regresan a sus valores iniciales (iguales a los del PI tradicional) por lo que la respuesta de ambos controladores es prácticamente la misma. En los estados transitorios de las esquinas del cuadrado ocurre algo similar, pero en menor medida; los factores del PI no lineal se incrementan con el error, provocando una respuesta más agresiva que provoca que el robot alcance la trayectoria antes que el robot controlado por el PI tradicional, sin embargo en este caso la diferencia es menos notoria debido a que el error

que se produce en las esquinas es significativamente menor al que se produce en el estado transitorio inicial.

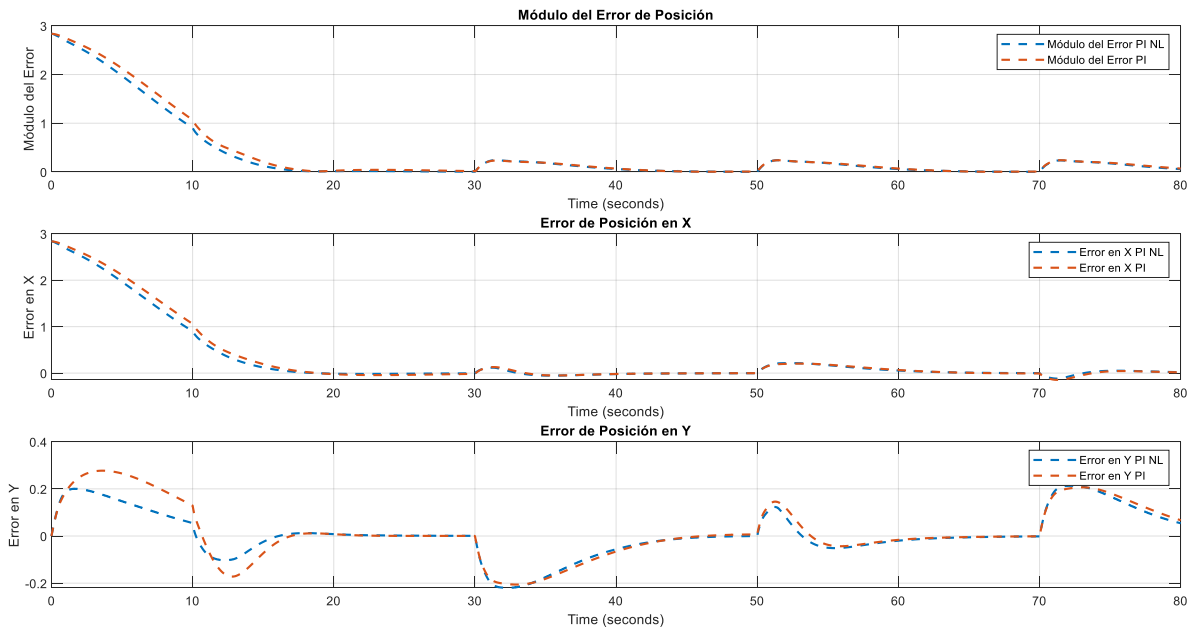


Figura 3.23. Comparación de errores de posición para PI No Lineal y PI Tradicional

En la Figura 3.23 se puede apreciar que el robot controlado por el PI no lineal tiene mejores resultados ante cambios de referencia bruscos que el PI tradicional, pues el módulo del error del primero disminuye a mayor velocidad que en el caso del segundo.

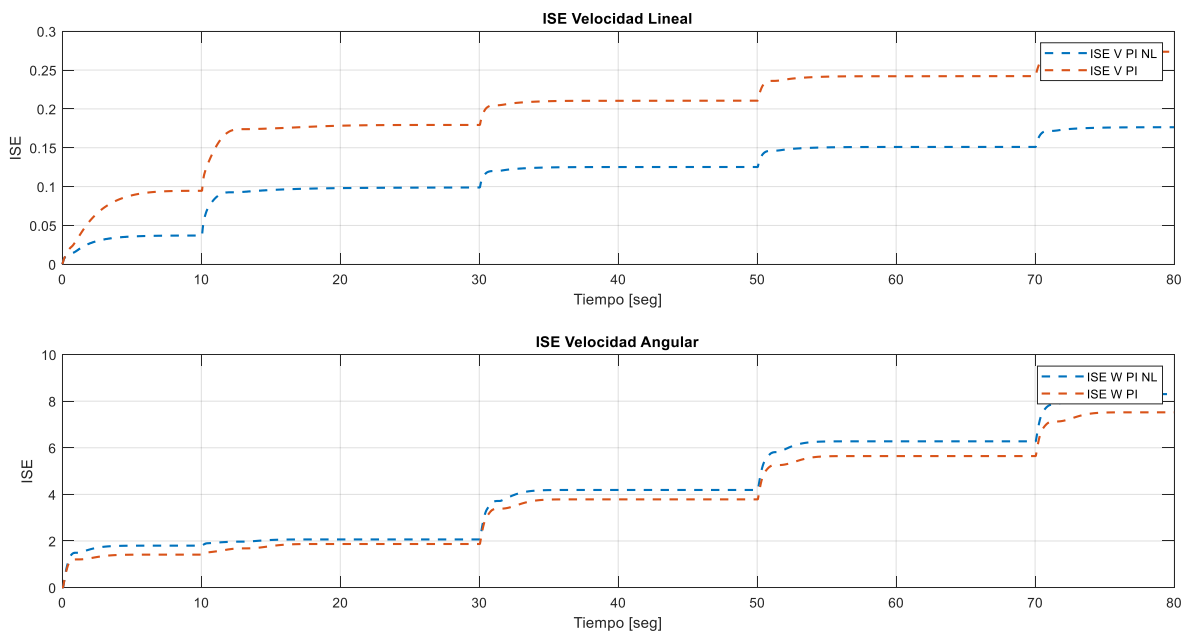


Figura 3.24. Comparación de ISE para PI No Lineal y PI Tradicional

Finalmente se comparan los índices de rendimiento de ambos controladores, en la Figura 3.24 se observa que el ISE de la velocidad lineal es menor para el PI no lineal, lo que confirma que el controlador es capaz de alcanzar la referencia de velocidad lineal antes que el PI tradicional, sin embargo, la respuesta más agresiva en velocidad lineal implica sacrificar la respuesta de velocidad angular, la cual tiene un ISE ligeramente mayor en el PID no lineal.

Cabe recalcar que, tal y como se explicó en el capítulo 2, la respuesta del controlador no lineal mejora si se sigue aumentando su factor característico, sin embargo, esto produce que la acción de control incremente demasiado, sacrificando la vida útil de los actuadores en el acto (razón por la cual se colocan saturadores). En el caso del Pioneer 3DX las velocidades lineal y angular máximas limitan el dimensionamiento del factor no lineal, por lo que su respuesta tiende a asemejarse al de un PI tradicional, aunque como se pudo ver en el apartado de comparaciones, las diferencias entre ambos esquemas son significativas en los estados transitorios.

3.5 Simulación en CoppeliaSim

Durante la simulación se pueden observar el seguimiento de trayectoria en tiempo real en la escena de CoppeliaSim mediante el uso de un objeto de dibujo que marca la ruta que sigue el robot, tal y como se muestra en la gráfica 3.25 que corresponden a simulaciones realizadas para las tres trayectorias vistos desde la escena de CoppeliaSim. Cabe recalcar que las trayectorias mostradas en este apartado corresponden exactamente a las figuras 3.1, 3.8 y 3.22 debido a que son las mismas mediciones de posición que se realimentan a Simulink.

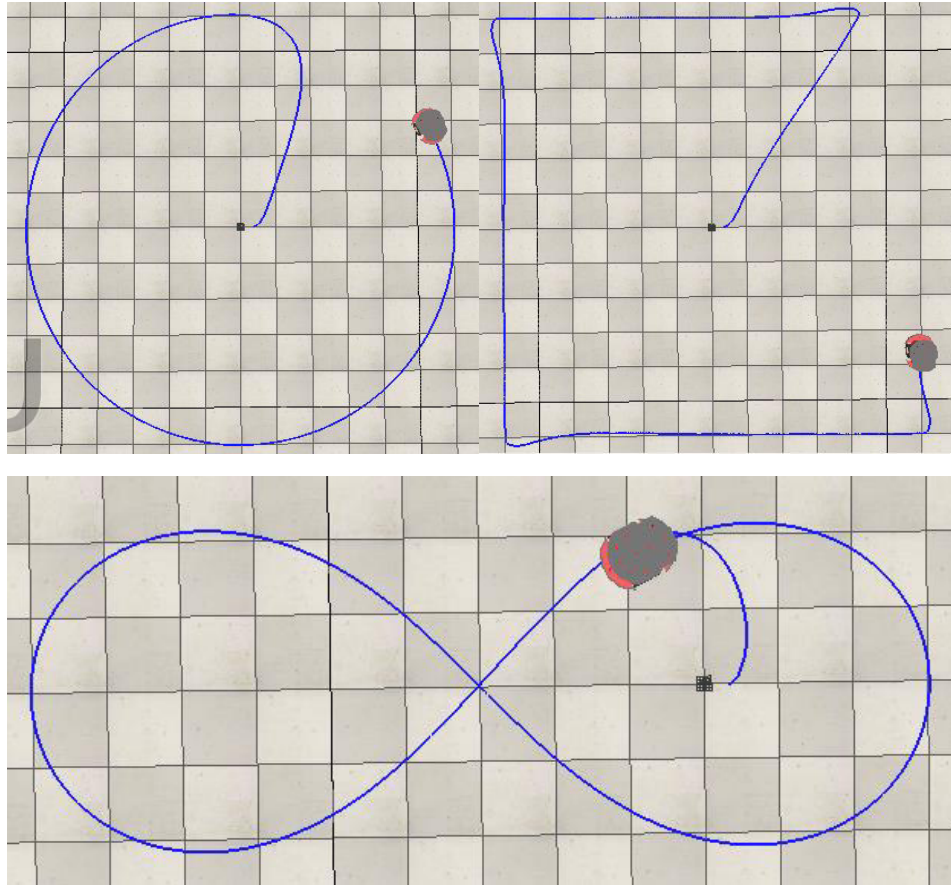


Figura 3.25. Seguimiento de las tres trayectorias en tiempo real en CoppeliaSim

3.6 Conclusiones

- Se realizó una recopilación bibliográfica de los diferentes esquemas utilizados para el control de trayectoria en robots móviles, los fundamentos de control para PID y sus variaciones propuestas.
- Se realizó el diseño y sintonización de un esquema de control por PID no lineal para el seguimiento de trayectorias de robots móviles.
- Se simuló el seguimiento de trayectorias mediante el uso un robot móvil Pioneer 3DX mediante un esquema Software in Loop, utilizando los softwares CoppeliaSim y Simulink
- Se realizaron las comparaciones del controlador PID no lineal propuesto frente a un controlador PID clásico, cuantificando por índices de desempeño, y estudiando el estado transitorio de las respuestas.

- Se desarrolló una interfaz gráfica en App Designer de Matlab para poder seleccionar el controlador, la trayectoria y visualizar los datos en pantalla.
- Se completó satisfactoriamente el diseño del controlador PI No Lineal, cumpliendo el objetivo de mejorar el desempeño del controlador PI tradicional, en base a las mediciones realizadas de los índices de desempeño ISE.
- La sintonización del controlador base se realizó utilizando los valores obtenidos en la estimación del sistema a modelo de primer orden, la cual se realizó en las primeras etapas del desarrollo del trabajo.
- Se pudo evidenciar que el controlador no lineal presenta una mejor respuesta en general que un controlador PI de ganancias fijas cuando el sistema se encuentra en estado transitorio, debido a que la magnitud de las constantes del controlador depende del valor del error en un instante de tiempo, pero una vez se alcanza el estado estable, el comportamiento de ambos controladores es similar debido a que las ganancias del controlador no lineal tienen aproximadamente sus valores iniciales.

3.7 Recomendaciones

- Al momento de realizar la sintonización de un esquema de control que pueda presentar acciones agresivas, se debe tener en cuenta los límites constructivos del robot para las variables manipuladas.
- Se pueden utilizar diferentes métodos para la aproximación del sistema a uno de primer orden con retardo que pueden presentar resultados diferentes a los que se obtuvieron en este trabajo, en ese caso se recomienda realizar la aproximación con otros métodos y comparar sus resultados para poder elegir el que se apegue más al sistema real.
- En el caso de la simulación, el uso de un tiempo de muestreo menor al que viene por defecto (50ms) incrementa la carga computacional del software utilizado y realentiza la simulación en general, por lo que se recomienda seleccionar un tiempo adecuado a las características del computador disponible.

4 REFERENCIAS BIBLIOGRÁFICAS

- [1] B. Sarabia, Diseño y Simulación de tres Técnicas de Control Clásicas y Robustas Aplicadas al Seguimiento de Trayectorias ante la Presencia de Retardos Fijos para la Plataforma Robótica Pioneer 3DC, Escuela Politécnica Nacional, Quito, Ecuador, Noviembre 2017.
- [2] O. Camacho, A new approach to design and tune sliding mode controllers for chemical processes, University of Florida, Tampa, FL, US, 1996.
- [3] J. M. Holland, Designing Autonomous Mobile Robots, Elsevier, 2004.
- [4] D. M. Anton, S. E. Szabo, C. A. Mociar, A. F. Iuhas, Line follower mobile robots, prototypes of robots and functional of mobile robots, Faculty of Managerial and Technological Engineering, University of Oradea, Oradea, Romania, 2021.
- [5] A. J. Alshaer, Making a drone and drone designing road map, Beykent University, Istanbul, Turkey, 2021.
- [6] C. Eising, L. Pereira, J. Horgan, A. Selvaraju, J. McDonald, P. Moran, 2.5D vehicle odometry estimation, Department of Electronic and Computer Engineering, University of Limerick, Limerick, Ireland, 2021.
- [7] G. Hernández, L. Ríos, M. Bueno, Implementación de un controlador de posición y movimiento de un robot móvil diferencial, Universidad de Preira, Pereira, Colombia, 2016.
- [8] J. Cortabitarte, P. Novarini, M. Naiouf, A. De Giusti, Seguimiento y estimación de trayectorias, Instituto de Investigación en Informática, UNLP, Buenos Aires, Argentina.
- [9] D. Bravo, C. Rengifo, Generación de trayectorias de marcha para un robot humanoide a partir de captura de movimiento, Ciencia en Desarrollo, Vol.12 (núm. 2) pp. 31-42, Julio-Diciembre 2021.

- [10] B. Garro, H. Sossa, R. Vázquez, Evolving ant colony system for optimizing path planning in mobile robots, Centro de Investigación en Computación, IPN, Mexico City, 2007.
- [11] Visioli Antonio, "Basics of PID Control", in *Practical PID Control*, 1st ed. Brescia, Italy: Springer Science+Business Media. 2006, pp. 3-7.
- [12] F. G. Shinskey, "Integrated Error Criteria", in *Process Control Systems*, 4th ed. New York, NY, USA: McGraw-Hill. 1996, pp. 16-18.
- [13] L. Asqui, "Planificación y seguimiento de caminos de manera autónoma para robots móviles tipo unicycle", Escuela Superior Politécnica de Chimborazo, Riobamba, Ecuador, Noviembre 2017.
- [14] Adept Mobilerobots, "Pioneer 3-DX", Adept Technology, 2011.
- [15] Adept Mobilerobots, "Pioneer 3 Operations Manual", Adept Technology, 2006.
- [16] G. Andaluz, Modelación, Identificación y Control de robots móviles, Escuela Politécnica Nacional, Quito, Ecuador, Octubre 2011.
- [17] O. Gonzáles, J. Rosero, A. Rosales, Control Predictivo Aplicado a un Robot Móvil, Escuela Politécnica Nacional, Quito, Ecuador, Junio 2014.
- [18] CoppeliaSim, "CoppeliaSim User Manual", CoppeliaRobotics, January 2022.
- [19] Matlab, "Simulink Documentation", Mathworks, 2021.
- [20] A. Morales, V-Rep como herramienta de simulación de sistemas dinámicos, Escuela Técnica Superior de Ingeniería, Universidad de Sevilla, Sevilla, España, 2019.
- [21] O. Camacho, A. Rosales, F. Rivas, "Parámetros característicos de un proceso", en *Control de Procesos*, 1era ed. Quito, Ecuador: EPN Editorial. 2020, pp. 14-16.
- [22] F. Martins, W. Celeste, R. Carelli, T. Batos-Filho, M. Sarcinelli-Filho, Controlador Dinámico Adaptable para Seguimiento de Trayectoria con Robots Móviles, Universidad Federal do Espírito Santo, Vitória, Brazil, Octubre 2007.
- [23] Y. Tian, A Non Linear PID Controller with Applications, Zhejiang University, Hangzhou, China, 1999.

[24] P. Anbumalar, J. Janani, Siriam, Uwise, Non Linear PID Control of pH Process, Abdur Rahman University, Chennai, India, 2015.

5 ANEXOS

ANEXO I

En el Anexo I se muestra el código utilizado para la obtención de los parámetros necesarios para la linealización de la planta por diferentes métodos, de forma automática. Estos valores fueron usados posteriormente para la sintonización de los controladores clásicos.

Cálculos para la velocidad lineal:

Modelo de primer orden con retardo (Método de Alfaro)

```
a=-0.91;  
b=0.910;  
c=1.262;  
d=-0.262;  
t1=0.154  
t2=0.4191  
Tao=a*t1+b*t2  
tm=c*t1+d*t2
```

Controlador PI (Método de Chien-Hrones-Reswick)

```
K=0.5/0.5  
Kp=(0.6*Tao)/(K*tm)  
Ti=Tao  
Ki=Kp/Ti
```

Controlador PID (Método de Chien-Hrones-Reswick)

```
Kc=(1/K)*(1.35*(Tao/tm)+0.25)  
Ti=Tao*((2.5*(tm/Tao)+0.46*(tm/Tao)^2)/(1+0.61*(tm/Tao)))  
Td=(0.37*tm)/(1+0.2*(tm/Tao))  
Kd=Td*Kc  
Ki=Kc/Ti
```

Cálculos para la velocidad angular:

Modelo de primer orden con retardo (Método de Alfaro)

```
a=-0.91;  
b=0.910;  
c=1.262;  
d=-0.262;  
t1=0.0828  
t2=0.2162  
Tao=a*t1+b*t2  
tm=c*t1+d*t2
```

Controlador PI (Método de Chien-Hrones-Reswick)

```
K=0.871/0.8  
Kp=(0.6*Tao)/(K*tm)  
Ti=Tao  
Ki=Kp/Ti
```

Controlador PID (Método de Chien-Hrones-Reswick)

```
Kc=(1/K)*(1.35*(Tao/tm)+0.25)  
Ti=Tao*((2.5*(tm/Tao)+0.46*(tm/Tao)^2)/(1+0.61*(tm/Tao)))  
Td=(0.37*tm)/(1+0.2*(tm/Tao))  
Kd=Td*Kc  
Ki=Kc/T
```

ANEXO II

En el Anexo II se incluye una guía rápida para el manejo del componente práctico del trabajo, empezando con la interfaz gráfica compuesta por una ventana de presentación en donde se puede seleccionar el controlador a utilizar (de entre los dos propuestos en el trabajo) y la trayectoria que va a seguir el robot.

Inicialmente en la carpeta anexada como componente práctico, se debe ejecutar el archivo llamado Ventana Principal desarrollado en Matlab App Designer.

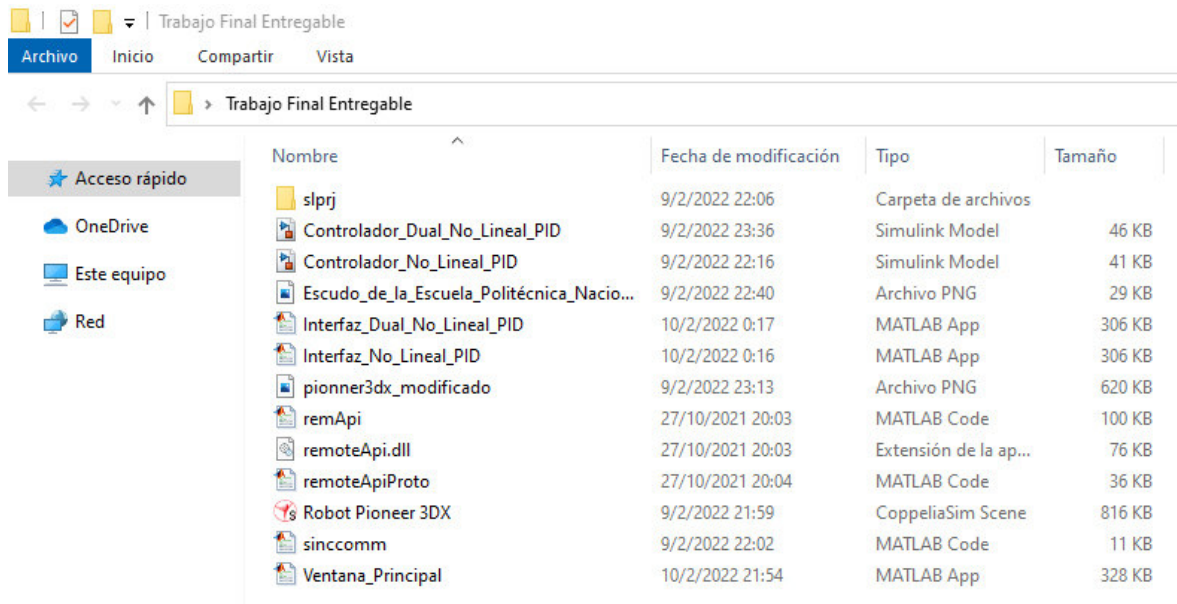


Figura 5.1. Contenido de la carpeta anexada como componente práctico

La ventana principal de la interfaz muestra una presentación sobre el trabajo y sus integrantes, además se puede elegir el controlador con el que se desea realizar la simulación.



Figura 5.2. Ventana principal de la interfaz

Una vez se selecciona el controlador que se desea simular va a aparecer la interfaz de simulación con la que se puede elegir la trayectoria que va a seguir el robot y los resultados que se van a mostrar. Adicionalmente, se va a abrir el archivo de simulink correspondiente al controlador seleccionado de forma automática. En el archivo de simulink, tal y como se expresó previamente incluye el generador de trayectorias, controlador cinemático y controladores de velocidad lineal y angular, junto con los bloques que realizan el cálculo del factor no lineal dependiendo del error. El bloque que realiza la comunicación con CoppeliaSim también se encuentra presente en el archivo, representado como una caja negra a la que ingresan las señales de velocidad lineal y angular, y salen las mediciones como posición, velocidad y orientación.

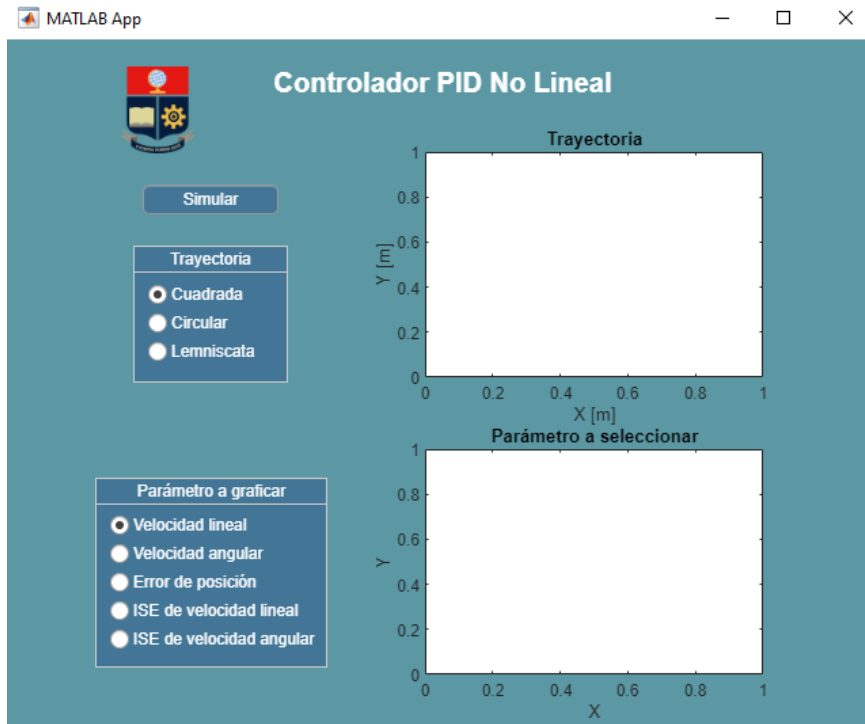


Figura 5.3. Interfaz gráfica para control y visualización de resultados

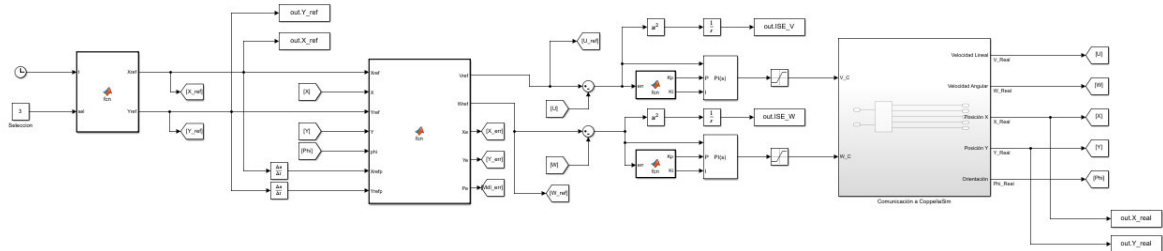


Figura 5.4. Archivo de simulink del controlador PID no lineal

Para proceder a la simulación se debe primero abrir la escena de CoppeliaSim que contiene el Pioneer 3DX, el archivo se llama Robot Pioneer 3DX y es incluido en la carpeta con el componente práctico. La escena únicamente incluye el robot en un entorno abierto y sin obstáculos, tal y como se ve en la figura 5.5.

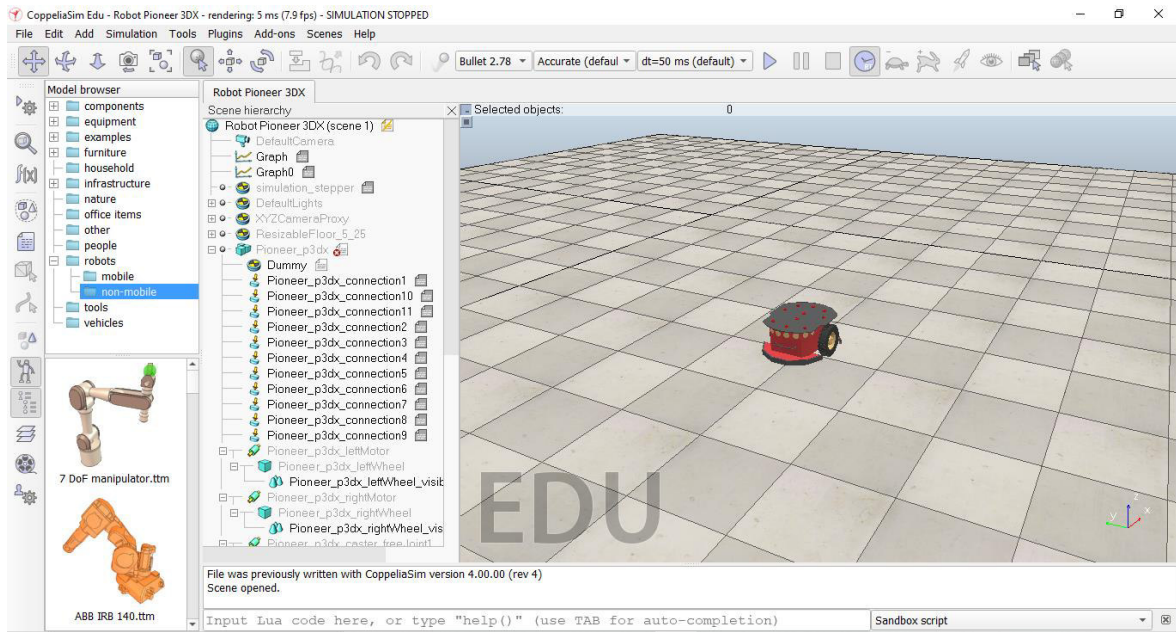


Figura 5.5. Escena de CoppeliaSim con el robot Pioneer 3DX

Con la escena abierta se puede proceder a iniciar la simulación, las mediciones obtenidas durante la misma se mostrarán al final en la pantalla, mientras que el seguimiento de trayectoria se puede observar en la escena de CoppeliaSim en tiempo real.

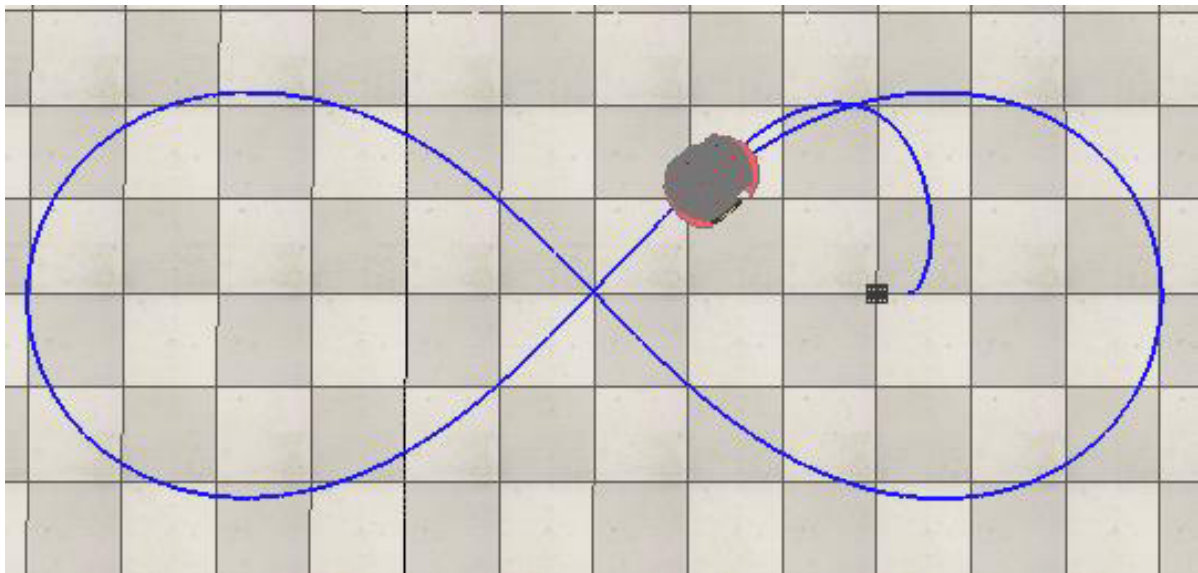


Figura 5.6. Demostración de seguimiento de trayectoria en tiempo real

Una vez finalizada la simulación se mostrarán los resultados en la interfaz gráfica, en la parte superior se muestra la trayectoria de referencia frente a la real, mientras que en el inferior se encuentran las mediciones más relevantes realizadas en todo el trayecto, correspondientes a la Velocidad Lineal, Angular, Errores de posición e Índices de rendimiento.

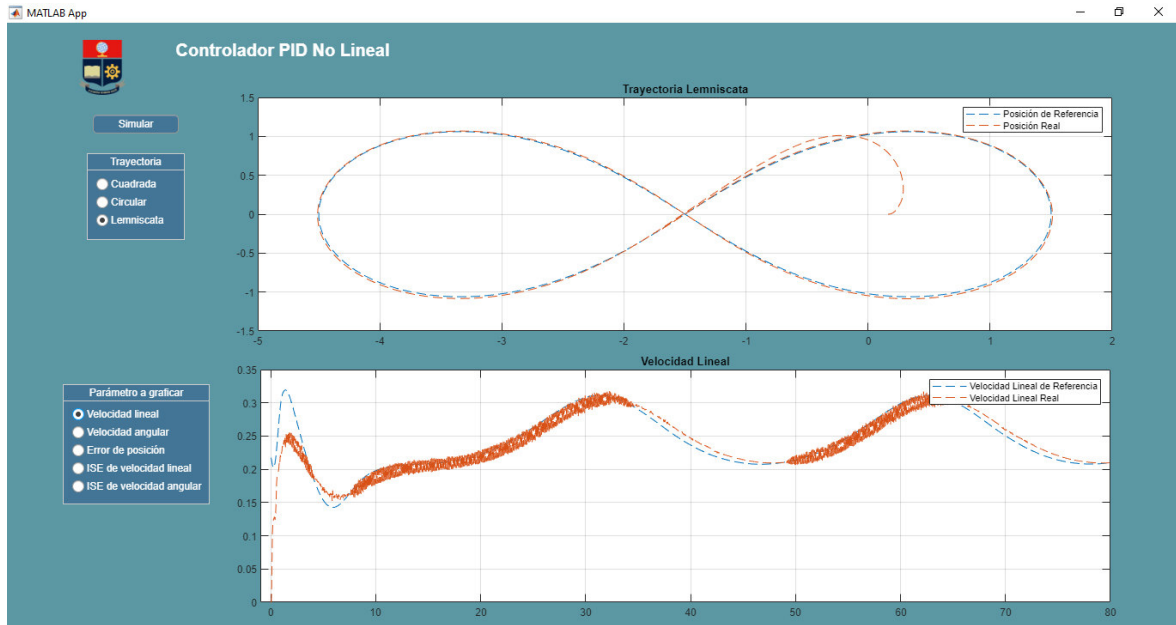


Figura 5.7. Demostración de resultados obtenidos para trayectoria Lemniscata