

Trabajo Fin de Grado Ingeniería Aeroespacial

Control de Trayectoria Basado en Álgebra Lineal

Autor: Diego Mesta Guisasola

Tutor: Francisco Rodriguez Rubio

**Dpto. Ingeniería de Sistemas y Automática
Escuela Técnica Superior de Ingeniería
Universidad de Sevilla**

Sevilla, 2021



Trabajo Fin de Grado
Ingeniería Aeroespacial

Control de Trayectoria Basado en Álgebra Lineal

Autor:

Diego Mesta Guisasola

Tutor:

Francisco Rodriguez Rubio

Profesor Titular

Dpto. Ingeniería de Sistemas y Automática
Escuela Técnica Superior de Ingeniería
Universidad de Sevilla

Sevilla, 2021

Trabajo Fin de Grado: Control de Trayectoria Basado en Álgebra Lineal

Autor: Diego Mesta Guisasola
Tutor: Francisco Rodriguez Rubio

El tribunal nombrado para juzgar el trabajo arriba indicado, compuesto por los siguientes profesores:

Presidente:

Vocal/es:

Secretario:

acuerdan otorgarle la calificación de:

El Secretario del Tribunal

Fecha:

Resumen

El problema de seguimiento de trayectoria es muy importante en la teoría de control. Su objetivo principal es garantizar que el sistema sigue, para una serie de variables, la evolución marcada por una referencia dada, esta referencia suelen obtenerse mediante procedimientos de optimización o mediante ordenes impuestas por usuarios, este sería el caso, por ejemplo de un robot que se encargue del transporte de herramientas a trabajadores. Otros ejemplos de implementaciones de soluciones para este problema son robots de transporte de carga, multirrotores que se encargan de analizar una estructura de forma autónoma y reactores químicos cuya evolución de reactivos deseada sea conocida de forma previa.

Este documento presenta una metodología para el diseño de controladores de seguimiento de trayectoria, convirtiendo el problema de diseño del controlador en un problema de resolución de un sistema de ecuaciones lineales. Este método busca dar un paradigma de resolución más sencillo a problemas de control complejos, lo que adicionalmente permite obtener conclusiones sobre el comportamiento del sistema bajo estudio, lo que facilita el reconocer las modificaciones necesarias para mejorar el rendimiento del sistema y del controlador.

Para resolver el problema de seguimiento de la trayectoria utilizando este método se requiere conocer la referencia que deben seguir todas las variables de estado del sistema, sin embargo, no todas pueden ser escogidas. Uno de los pasos más importantes para el diseño de un controlador utilizando este método es el análisis de cuales son las condiciones del sistema para tener una solución exacta, estas condiciones dan lugar a una serie de referencias puntuales para algunas de las variables de estado, cuya referencia no estaba definida de forma previa, es por ello que estas variables se denominan variables sacrificadas.

El objetivo principal del documento es definir el proceso de diseño de controladores utilizando esta metodología, nombrada LAB CD (Linear Algebra Based Controller Design), esto se consigue mediante la descripción del proceso general y su particularización para varios modelos de un robot móvil y un multirrotor. Adicionalmente, las simulaciones realizadas permiten comprobar el comportamiento de los controladores diseñados y las propiedades que este método confiere.

Abstract

The trajectory tracking problem is a very important one in control theory. Its main objective is to guarantee the system following, for a group of variables, the evolution given by a predefined reference, this reference is usually obtained through optimization procedures or through orders given by users, an example of this would be a transport robot that brings tools to workers. Other examples of the implementation of solutions for this problem would be load transporting robots, rotatory wing UAVs with the objective of autonomously analyzing an structure or chemical reactors whose reactive evolution needs to follow a predefined evolution.

This document presents a methodology used for the design of trajectory tracking controllers, converting the design problem in a matter of solving a linear equation system. This method tries to give a simpler point of view to more complex control problems, which additionally makes it easier to obtain conclusions about the behavior of the system under study, which in turn makes it easier to get to know the required modifications in order to improve its performance.

In order to solve the trajectory tracking problem using this methodology it is required a known reference for all of the state variables of the system involved in the model, although not all of them can be chosen. One of the most important parts of the controller design process using this method is the analysis of the conditions that allow the system to have an exact solution, this conditions define a series of temporary references for some of the state variables whose reference wasn't well defined previously, this is why this variables are called sacrificed variables.

The main objective of this document is to define the process of designing controllers using this methodology, which is called LAB CD (Linear Algebra Based Controller Design), this is accomplished through the definition of the general method and its application to various models of a mobile robot and rotatory wing UAV. Additionally, the simulations done with Simulink allow for checking the behavior of the designed controllers and the properties given to them by the method.

Índice

<i>Resumen</i>	I
<i>Abstract</i>	III
1 Introducción al Seguimiento de Trayectorias	1
1.1 Problemas de Seguimiento de Trayectorias	1
1.1.1 Control en Adelanto	2
1.1.2 Control por Realimentación	2
1.1.3 Seguimiento de Trayectoria Iterativo	3
1.2 Modelo del Proceso	3
1.3 Procesos	4
1.4 Idea General del Documento	4
2 Método de Diseño del Controlador	5
2.1 Problema Objetivo	5
2.2 Diseño del Controlador	6
2.3 Metodología de Diseño de Controladores Basados en Álgebra Lineal (LAB CD)	7
2.4 Ecuación del Error de Seguimiento	8
2.5 LAB CD en Tiempo Discreto	8
3 Aplicación a un Robot Móvil	11
3.1 Control Cinemático en CT	11
3.1.1 Diseño del Controlador	11
3.1.2 Errores de Seguimiento	12
3.1.3 Resultados de Simulaciones	13
Recta en OX	13
Recta en OY	14
Circunferencia	15
Lemniscata	16
3.2 Control Cinemático en DT (Aprox. de Euler)	17
3.2.1 Diseño del Controlador	17
3.2.2 Resultados de la Simulación	18
Circunferencia	18
Lemniscata	19
3.3 Control Cinemático en DT (Aprox. por Trapecios)	20
3.3.1 Diseño del Controlador	20
3.3.2 Resultados de la Simulación	21
Circunferencia	21
Lemniscata	22
3.4 Modelo Dinámico en DT	23
3.4.1 Diseño del Controlador	24

3.4.2	Resultados de la Simulación	25
	Circunferencia	25
	Lemniscata	26
3.5	Comparación de Controladores	27
4	Aplicación a un Multirotor	31
4.1	Controlador Simplificado	32
4.1.1	Diseño del Controlador	32
4.1.2	Resultados de Simulación	34
	Recta en OX	34
	Recta en OY	35
	Ascenso Rectilíneo	36
	Lemniscata	37
	Ascenso Helicoidal	38
4.1.3	Mejoras del Controlador	39
4.2	Controlador "Completo"	40
4.2.1	Diseño del Controlador	40
4.2.2	Resultados de Simulación	42
	Lemniscata	43
	Ascenso Helicoidal	44
	Efecto del Giro	45
5	Conclusiones Generales	49
A	Diagramas Simulink	51
B	Bibliografía	57
	<i>Índice de Figuras</i>	59

1 Introducción al Seguimiento de Trayectorias

El objetivo principal de un sistema de control es proveer las señales de entrada apropiadas a la planta para que la planta controlada se comporte de la forma deseada. La planta debe poder comportarse de la forma requerida de forma que las entradas generadas deben encontrarse en todo momento en el conjunto de comportamientos admisibles.

Dentro de los comportamientos deseados se pueden distinguir dos necesidades distintas; mantenerse en un punto estable, el denominado *problema de regulación*, o seguir una trayectoria dada, el denominado *problema de seguimiento*, en ambos casos se debe mantener el control a pesar de las perturbaciones externas. A pesar de tener objetivos distintos, ambos problemas se solucionan de forma similar y tienen procedimientos de diseño del control similares.

Más formalmente se puede escribir como: dada una planta de la cual se mide información mediante las variables de salida $y(t)$, determine la señal de entrada $u(t)$ necesaria para completar el objetivo de control, a pesar de las posibles perturbaciones. De forma general las señales de entrada están sometidas a una serie de condiciones y se dispone de un modelo de la planta $y(t) = \mathbf{M}[u(t)]$.

La mayoría de los sistemas prácticos tienen un comportamiento no lineal y la solución del problema de control lleva a un controlador no lineal. Existen una enorme cantidad de enfoques para analizar el comportamiento y el diseño del control para sistemas no lineales y una existe un enfoque único que sea válido para cualquier tipo de sistema no lineal. Por el contrario, la teoría y los procedimientos de diseño del control para sistemas lineales están bien establecidos y aceptados. Esto lleva a la metodología que se explica a lo largo del documento y que tiene como objetivo la aplicación de teoría lineal para poder resolver problemas de control no lineales, esta, como se verá a continuación, es aplicable para el seguimiento de la trayectoria en la cual el objetivo es seguir una referencia para ciertas variables.

1.1 Problemas de Seguimiento de Trayectorias

En los últimos años ha habido un aumento significativo en el interés en el seguimiento de trayectorias, siendo una de las razones principales la aparición de vehículos autónomos, capaces de seguir una trayectoria sobre terreno desconocido y evitando obstáculos. De modo que el controlador diseñado deberá ser capaz de minimizar errores por perturbaciones y de tratar con condiciones iniciales desconocidas.

El control de sistemas autónomos puede considerarse en distintos niveles que varían desde el caso más simple de avisos frente a condiciones inesperadas hasta sistemas totalmente independientes donde el sistema de control debe incorporar mecanismos de toma de decisiones capaces de responder a cualquier cambio.

Se pueden considerar principalmente dos enfoques distintos para la estructura de control: tomar la acción de control sin tomar información de la planta controlada y su evolución, conocido como *control en adelanto*; o tomar la acción de control en función de los datos tomados de la planta en tiempo real, conocido como *control por realimentación*. Ambos sistemas no son excluyentes, sino por el contrario son complementarios y deben utilizarse en conjunto para diseñar la solución de control, un ejemplo claro donde se hace esto es en los problemas de seguimiento de trayectorias, donde la información sobre las trayectorias a seguir está disponible.

1.1.1 Control en Adelanto

Considérese la representación externa de un sistema lineal como el siguiente:

$$y(s) = G(s)u(s) \quad (1.1)$$

donde s es la variable compleja de Laplace y $G(s)$ es la matriz de transferencia. El control ideal para seguir una referencia $y_r(s)$ sería:

$$u(s) = G^{-1}(s)y_r(s) \quad (1.2)$$

Esto requeriría que $G^{-1}(s)$ fuese posible y estable. El seguimiento perfecto se conseguiría si no hubiese perturbaciones, el modelo y las condiciones iniciales fuesen exactamente las de la planta real.

Para un proceso no lineal, considérese la siguiente representación matemática:

$$y(t) = \mathbf{M}[u(t)] \quad (1.3)$$

donde $\mathbf{M}[\cdot]$ es el operador matemático que transforma la señal de entrada en la señal de salida. Si el modelo fuese perfecto y este operador invertible, la entrada de control necesaria para seguir la salida deseada $y_r(t)$ sería:

$$u(t) = \mathbf{M}^{-1}[y_r(t)] \quad (1.4)$$

De esta forma, el control en adelanto alcanzaría a seguir la trayectoria, sin embargo, existen varios motivos que hacen este control inadecuado e insuficiente: en primer lugar si el modelo no es exacto y/o existe incertidumbre en algunos parámetros del modelo, la salida no seguirá la trayectoria, lo mismo podría ocurrir si aparecen perturbaciones o dinámicas no modeladas, o podría incluso darse que el operador $\mathbf{M}^{-1}[\cdot]$ fuese inestable o irrealizable.

Debido a los motivos anteriores, el control en adelanto no puede aplicarse de esta forma en ningún caso, debe ser complementado con algún control por realimentación que se encargue de tratar las perturbaciones y la incertidumbre del modelo.

1.1.2 Control por Realimentación

Es bien conocido el hecho de que el control por realimentación es capaz de estabilizar plantas inestables y puede hacer que una planta sea robusta frente a perturbaciones y a incertidumbres en el modelo. Existen muchos enfoques distintos a la hora de diseñar un controlador por realimentación, sin embargo, como se ha comentado anteriormente, estos dependen en gran medida del tipo de no-linealidad que aparezca en el modelo.

Entre las diversas técnicas existentes, algunas son más comunes que el resto. La *linealización por realimentación* tiene como objetivo precontrolar el proceso para reducirlo a un modelo lineal. Requiere encontrar una transformación que convierta el sistema no lineal en un sistema lineal equivalente representado por una cascada de integradores mediante un cambio de variables y una señal de entrada apropiada. Esta técnica es muy útil pero no es siempre aplicable (Krener, 1999). En principio, el sistema no lineal debería ser afín al control.

El *control adaptativo* es una técnica muy útil que requiere un bucle doble. Existen diversos enfoques y una gran cantidad de literatura que describe todas estas opciones (Åström & Wittenmark, 2013; Landau, 1979). De forma general, el bucle interior es un control realimentado que suele estar diseñado para un modelo lineal local de la planta, y el bucle exterior es bucle identificativo que se intenta estimar la variación de los parámetros en el modelo de la planta o cambiar directamente los parámetros del controlador para adaptarlos a la situación. De esta forma, el control adaptativo se centra en controlar las plantas en entornos desconocidos intentando mejorar el modelo de la planta y con ello el control en línea.

Otro enfoque distinto para el mismo problema es el *control robusto*. En este caso, la incertidumbre en el modelo o en las perturbaciones externas deberán estar acotadas y el control es diseñado de forma que se tenga un rendimiento satisfactorio bajo todas las condiciones posibles (Morari & Zafiriou, 1989; Zhou & John, 1999). Este difiere del control adaptativo en que este último no tiene los cambios acotados ni conocidos de forma previa.

El *Control Predictivo del Modelo* permite tratar con modelos no lineales y provee una señal de control óptimo considerando un horizonte limitado en un entorno acotado. Normalmente implica una gran carga

computacional debido al enfoque en la optimización, aunque algunos algoritmos en tiempo real muy eficientes se han propuesto para su aplicación en línea (Camacho & Bordons, 2004; Rawlings, Meadows, & Muske, 1994).

Si existen una gran incertidumbre y el modelo no es bien conocido, el *control en modo deslizante* es una metodología apropiada de diseño. En este enfoque, el controlador aplica señales de control fuertes, que no dependen mucho del modelo, para guiar el estado de la planta a una superficie deslizante donde se aplica una señal de control linealizada (Utkin, 1993).

Una característica crítica a la hora de implementar el control por realimentación es la información disponible para generar el control. En este sentido, debe distinguirse entre *control por realimentación de salida* y *control por realimentación de estado*. En el segundo caso, se dispone del estado del proceso. Esto significa que toda la información sobre el proceso puede ser utilizada y, si el proceso es controlable, su dinámica puede ser controlada de forma arbitraria. Por el contrario, si solo se dispone de la señal de salida, el rendimiento del control se degradará, incluso si se diseña e implementa un observador de estado.

1.1.3 Seguimiento de Trayectoria Iterativo

En el control por realimentación se requiere de un modelo del proceso y el rendimiento del controlador depende en gran medida de la precisión del modelo. A veces, el comportamiento del proceso no es bien conocido o las incertidumbres del modelo son muy grandes y, por ello, el enfoque anterior falla. Los requisitos del modelo pueden ser reducidos si la actividad de seguimiento es repetitiva, es decir, si el objetivo es realizar *tareas de movimiento repetitivas*.

En esta estructura más elaborada el control se compone de dos partes. Un control por realimentación básico, el cual es independiente del modelo, como un controlador Proporcional-Derivativo (PD), y un controlador en adelante que sea actualizado en cada iteración y computado para minimizar los errores de seguimiento en la iteración previa y que sea reinicializado al comienzo de cada prueba. De esta forma, se introduce una capacidad de aprendizaje en la etapa de diseño del controlador, reduciendo los errores de seguimiento (Owens & Hättönen, 2005).

En general, el diseño de un controlador de seguimiento de trayectoria algunos de los desarrollos de cada enfoque y, como conclusión general, estará condicionado en gran medida por el conocimiento disponible sobre la planta, es decir, sobre el modelo de la misma.

1.2 Modelo del Proceso

En los siguientes capítulos se va a presentar una metodología para procesos no lineales basada en álgebra lineal. Como se mostrará, el diseño del controlador se enfocará a través de la inversión del modelo, pero de forma suave. Esto implica que el modelo debe ser afín al control.

Una de las ideas principales es que en el seguimiento de trayectoria, no todas las variables del proceso tienen que seguir una referencia predefinida. De esta forma solo las variables a seguir tienen que ser forzadas a seguir las referencias, y el resto de variables, denotadas como *variables sacrificadas*, deberán seguir unas referencias ad hoc, que permitan un comportamiento suave de las variables a seguir. La referencia se asumirá conocida, así como sus derivadas, en caso de ser requeridas.

Utilizando una representación interna, el modelo será el siguiente:

$$\dot{x}(t) = F(x(t), u(t), d(t), t) \quad (1.5)$$

donde $x \in \mathfrak{R}^r$ denota el estado del sistema, $u \in \mathfrak{R}^m$ denota la entrada, $t \in \mathfrak{R}$ denota el tiempo continuo y $d \in \mathfrak{R}^r$ denota las perturbaciones externas. Si el modelo es invariante en el tiempo, t no será una entrada de la función F .

Normalmente, el número de variables a seguir será igual o menor que el número de entradas, esto quiere decir, que existirá un número máximo de $r - m$ variables sacrificadas.

En el enfoque a desarrollar, el modelo del proceso se asume afín al control, y las perturbaciones se suponen aditivas. Esto implica que (1.5) puede ser reescrito como:

$$\dot{x}(t) = f(x(t)) + g(x(t))u(t) + d(t) \quad (1.6)$$

donde $f(\cdot)$ es un vector de dimensión r y $g(\cdot)$ es una matriz $r \times m$.

La mayoría de las veces el controlador se implementará de forma digital, por tanto, el controlador del modelo deberá ser dado en tiempo discreto siendo el resultado de una discretización de un controlador en tiempo continuo o del diseño directo para un modelo discretizado de la planta.

Un problema crítico del control del tiempo discreto es la selección del tiempo de muestreo, el cual debe ser un compromiso entre los requisitos de la carga computacional y la precisión de seguimiento.

1.3 Procesos

El control de trayectoria aparece tradicionalmente en el diseño de *servosistemas*. Esto es muy común en plantas mecánicas donde el objetivo principal del control es garantizar que la señal motorizada sigue la referencia provista por la señal del área de información. Un ejemplo sería la regulación del ángulo de los alerones de una aeronave.

1.4 Idea General del Documento

El documento trata sobre un método de diseño de controladores de seguimiento de trayectoria para un conjunto variado de procesos. Este método se basa en el modelo matemático de la planta y su desarrollo implica conceptos de teoría de sistemas y álgebra lineal.

En el capítulo 2, la metodología, denominada Diseño de Control Basado en Álgebra Lineal (LAB CD), será descrita para el diseño de un controlador.

En el capítulo 3, la metodología LAB CD se aplicará a control de un robot móvil terrestre, utilizando una serie de modelos distintos y realizando simulaciones para comparar su funcionamiento.

En el capítulo 4, se aplica la metodología LAB CD para el control de un multirrotor utilizando dos modelos distintos, uno más complejo y otro simplificado y se comparan sus resultados mediante simulación.

2 Método de Diseño del Controlador

En este documento se trata el problema del seguimiento de trayectoria para sistemas no lineales. El sistema a considerar puede ser un vehículo autónomo, ya sea terrestre, aéreo o marino, o cualquier proceso en el que se deba seguir una referencia, por ejemplo, en un reactor químico.

El controlador de seguimiento de trayectorias debe proveer la dirección y la velocidad de los cambios en las variables de la planta para guiarla a lo largo de un camino predefinido. El camino deseado es el resultante de un sistema de decisión de más alto nivel, que se base en alguna estrategia de planeamiento de camino.

A la hora del seguimiento de trayectoria, se pueden considerar tres escenarios diferentes: (1) guiar la planta desde un punto inicial a un punto final (el objetivo), esto se denomina movimiento *punto-a-punto*, y en este caso la trayectoria y el tiempo que toma no es tan relevante; (2) guiar la planta a través de una trayectoria geométrica independientemente del tiempo, esto se conoce como *seguimiento de camino*, y el objetivo principal es seguir la trayectoria deseada sin inquirir en grandes errores; (3) guiar la planta desde un punto inicial a lo largo de una trayectoria requerida, siguiendo un tiempo preescrito, esto es conocido como seguimiento de ruta y es el control de trayectoria más completo, ya que implica el control de la evolución geométrica de la planta así como su velocidad.

La metodología se centra en la resolución del problema de seguimiento de ruta y en este capítulo se presentan sus principales características y se explica el método utilizado para el diseño de controladores siguiendo esta metodología.

2.1 Problema Objetivo

Se asumen conocidos un modelo de la planta, así como de la trayectoria a seguir. El modelo básico general puede expresarse de la siguiente forma:

$$\begin{aligned}\dot{x}(t) &= F(x(t), u(t), d(t), t) \\ y(t) &= H(x(t), u(t), t)\end{aligned}\tag{2.1}$$

donde $x \in \mathfrak{R}^r$ denota el estado del sistema, $u \in \mathfrak{R}^m$ denota la entrada, $y \in \mathfrak{R}^p$ denota la salida del sistema y $d \in \mathfrak{R}^r$ denota la perturbación externa. El estudio de este problema de control tiene una gran cantidad de literatura de investigación y en la solución óptima está basada, en la mayor parte de los casos en un enfoque ad hoc para la situación concreta.

En el estudio realizado se asumen una serie de simplificaciones:

1. El modelo es afín en control.
2. El modelo se encuentra en fase mínima.
3. El modelo es invariante en el tiempo.
4. El estado del sistema es mensurable.
5. El modelo es exacto y no existen perturbaciones.

Es decir, el modelo inicial viene dado por:

$$\begin{aligned}\dot{x}(t) &= f(x(t)) + g(x(t))u(t) \\ y(t) &= x(t)\end{aligned}\tag{2.2}$$

La quinta hipótesis se relajará más adelante y se considerarán algunas incertidumbres y perturbaciones. Además, la cuarta hipótesis puede ser relajada si se añaden observadores o realimentación dinámica de

la salida. La tercera hipótesis también puede ser eliminada si la ley de variación es conocida, añadiendo complejidad a la computación del control. La primera hipótesis no es muy restrictiva ya que la mayoría de los procesos a considerar presentan esta estructura de control. Por último, la hipótesis 2 es necesaria debido a una cancelación del modelo en la solución formal.

Para describir por completo el problema de seguimiento, algunas trayectorias de referencia deben ser provistas. Se asumirá en todo momento que el planificador de movimientos provee una trayectoria posible, es decir, una trayectoria que puede ser seguida mediante la entrada adecuada. Por tanto, se tiene una hipótesis adicional:

6. Tanto la referencia como sus derivadas son conocidas.

Sin embargo, en muchos casos la trayectoria solo está definida para un conjunto de las variables de estado y la trayectoria del resto de variables de estado es opcional, permitiendo algo de libertad en la implementación del control.

2.2 Diseño del Controlador

Se procede a dividir el vector de variables de estado en dos partes: las variables a seguir, $\xi(t) \in \mathfrak{R}^{r_1}$ y las variables restantes, $z(t) \in \mathfrak{R}^{r-r_1}$. Estas últimas son las denominadas variables sacrificadas comentadas anteriormente. Normalmente, el número de variables a seguir (r_1) es igual al número de acciones de control independientes (m), y el número de variables sacrificadas depende del modelo de la planta. Pudiéndose escribir el modelo de la planta de la siguiente forma:

$$\begin{bmatrix} \dot{\xi}(t) \\ \dot{z}(t) \end{bmatrix} = \begin{bmatrix} f_{\xi}(\xi(t), z(t)) \\ f_z(\xi(t), z(t)) \end{bmatrix} + \begin{bmatrix} g_{\xi}(\xi(t), z(t)) \\ g_z(\xi(t), z(t)) \end{bmatrix} u(t) \quad (2.3)$$

Si $\xi_{ref}(t)$ es provisto por el planificador de movimiento y z_{ref} se definirá más adelante; el problema de control puede ser escrito como: dado el modelo (2.3) y las referencias $\xi_{ref}(t)$, encontrar la entrada de control $u(t)$ forzando el subvector de estado $\xi(t)$ a seguir la referencia $\xi_{ref}(t)$ satisfaciendo el modelo. La derivada del vector de variables a seguir, $\xi(t)$, se reemplazan inicialmente por las de las variables de referencia, asumiendo un acercamiento suave, por ejemplo, proporcional al error. En la definición del problema no se hace referencia a las variables sacrificadas, a pesar de esto, las derivadas de algunas variables sacrificadas, en caso de ser requeridas, se definirán más adelante, considerándose el mismo enfoque. Este es:

$$\begin{bmatrix} \dot{\xi}(t) \\ \dot{z}(t) \end{bmatrix} = \begin{bmatrix} \dot{\xi}_{ref}(t) + k_{\xi}[\xi_{ref}(t) - \xi(t)] \\ \dot{z}_{ref}(t) + k_z[z_{ref}(t) - z(t)] \end{bmatrix} \quad (2.4)$$

donde k_{ξ} , k_z son dos matrices diagonales (de dimensiones r_1 , $r - r_1$, respectivamente), las cuales son los parámetros de control. Combinando (2.3) y (2.4), se obtiene el siguiente modelo para la planta controlada.

$$\begin{bmatrix} \dot{\xi}_{ref}(t) + k_{\xi}[\xi_{ref}(t) - \xi(t)] - f_{\xi}(\xi(t), z(t)) \\ \dot{z}_{ref}(t) + k_z[z_{ref}(t) - z(t)] - f_z(\xi(t), z(t)) \end{bmatrix} = \begin{bmatrix} g_{\xi}(\xi(t), z(t)) \\ g_z(\xi(t), z(t)) \end{bmatrix} u(t) \quad (2.5)$$

De forma que las propiedades del LAB CD depende de la selección de los parámetros (o aquellos derivados de otros enfoques más complejos que el de (2.3)) y de las búsqueda de una solución a (2.5) para derivar la acción de control, $u(t)$.

La ecuación (2.5) puede ser reescrita como:

$$b(t) = A(t)u(t) \quad (2.6)$$

donde A es una matriz conocida con dimensiones $r \times m$ y b es un vector de dimensión r , donde algunos de los datos son conocidos de forma parcial ($z_{ref}(t), \dot{z}_{ref}(t)$).

Para poder encontrar la solución de $u(t)$ en (2.5), es un requisito que b sea una combinación de los vectores columna de A . Esto determinará el valor posible para la referencia de algunas de las variables sacrificadas, $z_{ref}(t)$, además de modificar la primera fila de (2.4) de forma que la primera fila de (2.5) cumple esta condición. Una vez que se han definido b y A , el control se obtiene resolviendo (2.6) utilizando la solución en mínimos cuadrados.

$$u(t) = A^{\dagger}(t)b(t) \quad (2.7)$$

donde $A^{\dagger}(t)$ es la matriz pseudoinversa de la matriz $A(t)$.

2.3 Metodología de Diseño de Controladores Basados en Álgebra Lineal (LAB CD)

Como se ha comentado anteriormente, la metodología LAB CD (Linear Algebra Based Controller Design) se utiliza principalmente para diseñar el seguimiento de trayectoria de una planta no lineal cuyo modelo es afín al control. Inicialmente, las hipótesis detalladas anteriormente también son consideradas:

1. El modelo es de fase mínima. 2. El modelo de la planta es invariante en el tiempo. 3. No existen perturbaciones internas ni externas. 4. El estado es accesible y medible. 5. Las referencias a seguir, así como sus derivadas (por lo menos la de primer y segundo orden) también son accesibles.

Algunas de estas hipótesis serán relajadas más adelante.

Para tener un proceso más formal, se definen los siguientes pasos:

Paso 1. Obtener una representación interna del modelo de la planta.

$$\dot{x}(t) = f(x(t)) + g(x(t))u(t) \quad (2.8)$$

Paso 2. Dividir el vector de estado en dos subvectores, uno con las variables a seguir, $\xi(t)$, cuya referencia es conocida, y el resto de las variables de estado, $z(t)$, denominadas como variables sacrificadas, cuya referencia será determinada. El nuevo modelo será:

$$\begin{bmatrix} \dot{\xi}(t) \\ \dot{z}(t) \end{bmatrix} = \begin{bmatrix} f_{\xi}(\xi(t), z(t)) \\ f_z(\xi(t), z(t)) \end{bmatrix} + \begin{bmatrix} g_{\xi}(\xi(t), z(t)) \\ g_z(\xi(t), z(t)) \end{bmatrix} u(t) \quad (2.9)$$

Paso 3. Definir las derivadas de las variables de estado como una aproximación de sus referencias. Esta aproximación es la etapa de diseño del control. Para cualquiera de las variable de estado, la diferencia entre sus derivadas y las correspondientes referencias es una función del error actual entre el estado y la referencia correspondiente. En el más simple de los casos, esta función no es más que un factor proporcional, como se expresa a continuación:

$$\begin{bmatrix} \dot{\xi}(t) \\ \dot{z}(t) \end{bmatrix} = \begin{bmatrix} \dot{\xi}_{ref}(t) + k_{\xi}[\xi_{ref}(t) - \xi(t)] \\ \dot{z}_{ref}(t) + k_z[z_{ref}(t) - z(t)] \end{bmatrix} \quad (2.10)$$

Es importante recordar que no todas las variables sacrificadas requieren tener una referencia, ya que sus dinámicas pueden ser determinadas por las acciones de control computadas. La selección de los parámetros a si como el balance de los errores definirá la ley de control.

En este paso, se puede definir el control asumiendo enfoques más complejos que el anterior, pero el coste computacional suele sobrepasar los posibles beneficios.

Paso 4. Se combinan las dos últimas ecuaciones ara llegar a un modelo de la planta controlada.

$$\begin{bmatrix} \dot{\xi}_{ref}(t) + k_{\xi}[\xi_{ref}(t) - \xi(t)] - f_{\xi}(\xi(t), z(t)) \\ \dot{z}_{ref}(t) + k_z[z_{ref}(t) - z(t)] - f_z(\xi(t), z(t)) \end{bmatrix} = \begin{bmatrix} g_{\xi}(\xi(t), z(t)) \\ g_z(\xi(t), z(t)) \end{bmatrix} u(t) \quad (2.11)$$

que puede ser resumido como

$$b(t) = A(t)u(t) \quad (2.12)$$

La matriz A es conocida de (2.3). El valor de referencia de las variables sacrificadas, así como sus derivadas que aparecen en el vector b , no están definidas. Para tener una solución exacta para $u(t)$, b debe ser combinación lineal de los vectores columna de A . Las referencias para las variables sacrificadas, así como sus derivadas, se definen de forma que esta condición se cumpla.

Nota: La computación puede requerir en algunos casos aproximaciones que se discutirán más adelante cuando se trate con distintas aplicaciones

Paso 5. Se resuelve la ecuación anterior de forma óptima, utilizando el método de mínimos cuadrados, computándose la acción de control de la forma siguiente:

$$u(t) = A^{\dagger}(t)b(t) \quad (2.13)$$

Nota: Como se discutía anteriormente, los parámetros de control embebidos en el vector $b(t)$, y su sintonización pueden darse por soluciones ad hoc, o utilizando enfoques generales de optimización.

2.4 Ecuación del Error de Seguimiento

En el caso no lineal, el error de seguimiento está muy involucrado, pero puede derivarse de la siguiente forma: asumiendo que la referencia para todas las variables de control son requeridas. En el paso 4, para que una solución de (2.5) sea posible, la referencia debe ser tal que

$$\begin{bmatrix} \dot{\xi}_{ref}(t) + k_{\xi}[\xi_{ref}(t) - \xi(t)] - f_{\xi}(\xi(t), z_{ref}(t)) \\ \dot{z}_{ref}(t) + k_z[z_{ref}(t) - z(t)] - f_z(\xi(t), z(t)) \end{bmatrix} = \begin{bmatrix} g_{\xi}(\xi(t), z_{ref}(t)) \\ g_z(\xi(t), z(t)) \end{bmatrix} u(t) \quad (2.14)$$

tenga una solución única. De esta forma, substrayendo (2.3), se obtiene

$$\begin{aligned} & \begin{bmatrix} \dot{\xi}_{ref}(t) - \dot{\xi}(t) + k_{\xi}(\xi_{ref}(t) - \xi(t)) - [f_{\xi}(\xi(t), z_{ref}(t)) - f_{\xi}(\xi(t), z(t))] \\ \dot{z}_{ref}(t) - \dot{z}(t) + k_z(z_{ref}(t) - z(t)) \end{bmatrix} \\ & = \begin{bmatrix} g_{\xi}(\xi(t), z_{ref}(t)) - g_{\xi}(\xi(t), z(t)) \\ 0 \end{bmatrix} u(t) \end{aligned} \quad (2.15)$$

Esto se reescribe como,

$$\begin{bmatrix} \dot{e}_{\xi} \\ \dot{e}_z \end{bmatrix} = \begin{bmatrix} -k_{\xi} e_{\xi} \\ -k_z e_z \end{bmatrix} + \begin{bmatrix} f_{\xi}(\xi, z_{ref}) - f_{\xi}(\xi, z) \\ 0 \end{bmatrix} - \begin{bmatrix} g_{\xi}(\xi, z_{ref}) - g_{\xi}(\xi, z) \\ 0 \end{bmatrix} u(t) \quad (2.16)$$

$$\begin{bmatrix} \dot{e}_{\xi} \\ \dot{e}_z \end{bmatrix} = \begin{bmatrix} -k_{\xi} e_{\xi} \\ -k_z e_z \end{bmatrix} + \begin{bmatrix} h_{\xi}(\xi, z_{ref}, z, u) \\ 0 \end{bmatrix} \quad (2.17)$$

Estos términos adicionales determinarán la evolución de los errores. En las aplicaciones desarrolladas en los siguientes capítulos se mostrará que bajo unas condiciones bien definidas estos términos se anularán, y la estabilidad de la planta controlada se asegurará mientras que los parámetros del controlador k_{ξ} y k_z sean positivos. Además, si no todas las variables sacrificadas necesitan seguir una referencia predeterminada, la segunda fila en (2.14)-(2.17) serán diferentes.

Una característica interesante de LAB CD en el enfoque mostrado en (2.17) es que, primero, el error de seguimiento de las variables sacrificadas se guía a cero, y, posteriormente, los errores de seguimiento de trayectoria también se guían a cero.

2.5 LAB CD en Tiempo Discreto

Hoy en día, todos los controladores se implementan en sistemas digitales, de forma que el controlador debe ser expresado en DT. Una forma es discretizar la acción de control previamente computada, pero se pueden obtener mejores resultados si el controlador se deriva de un modelo en DT de la planta. Además el modelo inicial de la planta puede ser expresado en DT, y, por tanto, el controlador se puede diseñar directamente en este entorno.

Se asume un modelo en planta como el dado por (2.2). Para obtener las medidas, se emplea un muestreador de periodo T y se utiliza un dispositivo de retención con el mismo periodo para aplicar la acción de control. El primer paso es discretizar el modelo. El enfoque más simple es la aproximación de Euler

$$\frac{d[x(t)]}{dt} \Big|_{t=nT} \simeq \frac{x((n+1)T) - x(nT)}{T} = \frac{x_{n+1} - x_n}{T} \quad (2.18)$$

donde n es el instante de muestreo. Muchas otras aproximaciones pueden ser utilizadas, pero si el tiempo de muestreo es lo suficientemente corto, la aproximación de Euler es aceptable, y puede ser aplicada con facilidad a plantas no lineales.

De forma que, en el entorno DT, los pasos del diseño del control serán similares a los destacados anteriormente.

Paso 1. Obtener una representación interna de la planta en DT. Si el modelo inicial es el dado por (2.2), el modelo DT será

$$x_{n+1} = x_n + T[f(x_n) + g(x_n)u_n] \quad (2.19)$$

Paso 2. Dividir el vector de estado en dos subvectores, uno con las variables a seguir, ξ_n , cuya referencia es conocida, y las variables sacrificadas, z_n , cuya referencia será determinada. El nuevo modelo es

$$\begin{bmatrix} \xi_{n+1} \\ z_{n+1} \end{bmatrix} = \begin{bmatrix} \xi_n \\ z_n \end{bmatrix} + T \left(\begin{bmatrix} f_\xi(\xi_n, z_n) \\ f_z(\xi_n, z_n) \end{bmatrix} + \begin{bmatrix} g_\xi(\xi_n, z_n) \\ g_z(\xi_n, z_n) \end{bmatrix} u_n \right) \quad (2.20)$$

Paso 3. Define el próximo estado de las variables de estado como una aproximación de sus referencias. Esta aproximación es la etapa de diseño del control. Para cualquier variable de estado, su incremento, así como su correspondiente referencia es una función del error actual entre su estado actual y su referencia. En el caso más simple, esta función es simplemente un factor proporcional, como se expresa en (2.4). Ahora, en DT

$$\begin{bmatrix} \xi_{n+1} - \xi_n \\ z_{n+1} - z_n \end{bmatrix} = \begin{bmatrix} \xi_{ref,n+1} - \xi_{ref,n} + T \kappa_\xi [\xi_{ref,n} - \xi_n] \\ z_{ref,n+1} - z_{ref,n} + T \kappa_z [z_{ref,n} - z_n] \end{bmatrix} \quad (2.21)$$

donde, como antes, κ_ξ y κ_z son matrices diagonales cuyos elementos ponderan el error de seguimiento en las siguientes referencias. De ahora en adelante, se denota por $k_\xi = I - T \kappa_\xi$ y $k_z = I - T \kappa_z$. Esto permite reescribir (2.21) como

$$\begin{bmatrix} \xi_{n+1} \\ z_{n+1} \end{bmatrix} = \begin{bmatrix} \xi_{ref,n+1} \\ z_{ref,n+1} \end{bmatrix} - \begin{bmatrix} k_\xi (\xi_{ref,n} - \xi_n) \\ k_z (z_{ref,n} - z_n) \end{bmatrix} \quad (2.22)$$

En esta ecuación se ve claramente que para $k_\xi = k_z = I$ el error de seguimiento se mantiene constante, mientras que para $k_\xi = k_z = 0$ el error se cancela en un solo periodo de muestreo. Valores intermedios de los coeficientes proveerán una evolución más suave, reduciendo el progresivamente. Estos son los parámetros de control a sintonizar.

Paso 4. Combinando (2.20) y (2.22), se obtiene un modelo de la planta controlada

$$\begin{bmatrix} \xi_{ref,n+1} - k_\xi (\xi_{ref,n} - \xi_n) - \xi_n - T f_\xi(\xi_n, z_n) \\ z_{ref,n+1} - k_z (z_{ref,n} - z_n) - z_n - T f_z(\xi_n, z_n) \end{bmatrix} = \begin{bmatrix} T g_\xi(\xi_n, z_n) \\ T g_z(\xi_n, z_n) \end{bmatrix} u_n \quad (2.23)$$

que puede ser resumido como

$$b_n = A_n u_n \quad (2.24)$$

El valor requerido de $z_{ref,n}$ se determinará de forma que se garantice que (2.23) es resoluble.

Paso 5. Una vez $z_{ref,n}$ ha sido evaluado, el valor de $z_{ref,n+1}$ será estimado extrapolando los valores previamente definidos de la secuencia $\{z_{ref,n}\}$. Diferentes enfoques pueden ser seguidos para esta extrapolación, pero métodos simples dan lugar a resultados excelentes si las referencias son lentas con respecto al tiempo de muestreo.

Resolviendo (2.24) mediante mínimos cuadrados se obtiene la acción de control como

$$u_n = A_n^\dagger b_n \quad (2.25)$$

3 Aplicación a un Robot Móvil

Los robots móviles son muy utilizados en la industria, la agricultura y en los hogares, lo que los convierte hoy en día en uno de los dispositivos más comúnmente utilizados como objetivos del problema de seguimiento de trayectoria.

En este capítulo se utilizará la metodología LAB CD para ilustrar el diseño de controladores para robots móviles. Se comenzará empleando el más simple de los modelos cinemáticos en CT y no se considerarán perturbaciones.

El robot móvil que se trata en este capítulo se representa a continuación

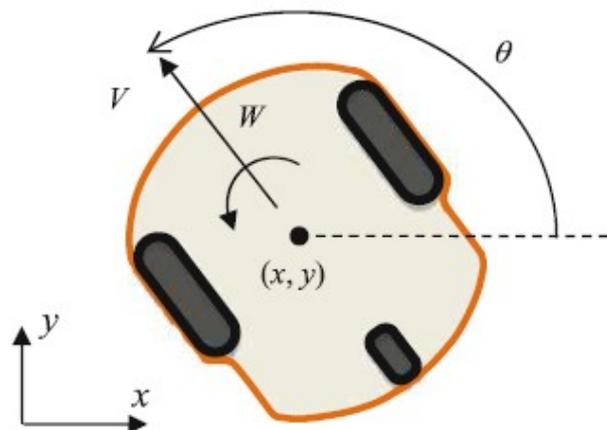


Figura 3.1 Representación Gráfica del Robot Móvil objetivo del estudio.

3.1 Control Cinemático en CT

3.1.1 Diseño del Controlador

A continuación se muestra un modelo cinemático no lineal para un robot móvil, el cual será utilizado para el control cinemático.

$$\begin{cases} \dot{x}(t) = V(t) \cos \theta(t) \\ \dot{y}(t) = V(t) \sin \theta(t) \\ \dot{\theta}(t) = W(t) \end{cases} \quad (3.1)$$

donde V es la velocidad lineal del robot y W es la velocidad angular, (x,y) es la posición en coordenadas Cartesianas y θ es la orientación del robot. Este es un modelo usado ampliamente en la bibliografía y que presenta la ventaja de que sus variables de control V y W son las señales de entrada de la mayoría de robots comerciales.

Por tanto, el objetivo del controlador será hallar los valores de V y W tales que el robot siga la trayectoria preestablecida (x_{ref}, y_{ref}) con el mínimo error posible. En este caso, la orientación será una variable sacrificada para guiar de forma suave la posición del robot, de forma que θ_{ref} se determinará posteriormente.

Aplicando la metodología LAB CD descrita en el capítulo 2, se asume un acercamiento suave de forma que las derivadas de las variables de estado pueden escribirse como:

$$\begin{cases} \dot{x}(t) = \dot{x}_{ref}(t) + k_x e_x(t) \\ \dot{y}(t) = \dot{y}_{ref}(t) + k_y e_y(t) \\ \dot{\theta}(t) = \dot{\theta}_{ref}(t) + k_\theta e_\theta(t) \end{cases} \quad (3.2)$$

donde los coeficientes k son los parámetros del controlador asumiendo una trayectoria con acercamiento proporcional al error. Para simplificar la notación se denotará la dinámica de la siguiente forma:

$$\begin{cases} \Delta x = \dot{x}(t) = \dot{x}_{ref}(t) + k_x e_x(t) \\ \Delta y = \dot{y}(t) = \dot{y}_{ref}(t) + k_y e_y(t) \\ \Delta \theta = \dot{\theta}(t) = \dot{\theta}_{ref}(t) + k_\theta e_\theta(t) \end{cases} \quad (3.3)$$

De esta forma, combinando (3.1) y (3.3), suprimiendo el argumento del tiempo, escribiendo de forma matricial y extrayendo los parámetros de control, se tiene que el modelo del problema de control puede escribirse como:

$$\begin{bmatrix} \Delta x \\ \Delta y \\ \Delta \theta \end{bmatrix} = \begin{bmatrix} \cos \theta & 0 \\ \sin \theta & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} V \\ W \end{bmatrix} \quad (3.4)$$

Esta ecuación del tipo $b = Au$ es la ecuación del problema que se buscaba. Para que la ecuación tenga una solución exacta, b debe estar en el espacio columna de A , de esto se llega a la siguientes condiciones:

$$\begin{cases} \frac{\Delta x}{\Delta y} = \frac{V \cos \theta}{V \sin \theta} = \frac{\cos \theta}{\sin \theta} \\ \Delta \theta = W \end{cases} \quad (3.5)$$

De forma que, para satisfacer la primera condición, se escoge la siguiente referencia para la orientación:

$$\theta_{ref} = \arctan \frac{\Delta y}{\Delta x} \quad (3.6)$$

De esta forma, se puede computar las acciones de control aplicando mínimos cuadrados a (3.4) una vez se ha sustituido (3.6) en el valor de θ para obtener la solución exacta; llegando a:

$$\begin{bmatrix} V \\ W \end{bmatrix} = \begin{bmatrix} \Delta x \cos \theta_{ref} + \Delta y \sin \theta_{ref} \\ \Delta \theta \end{bmatrix} \quad (3.7)$$

3.1.2 Errores de Seguimiento

La convergencia a cero de los errores de seguimiento dependerá de los parámetros del controlador, de forma que se va a demostrar los valores que deben tomar estos para que esto ocurra.

Considérese la orientación en la última ecuación de (3.1) y la acción de control de (3.7), se tiene:

$$\dot{\theta}(t) = \dot{\theta}_{ref}(t) + k_\theta (\theta_{ref}(t) - \theta(t)) \Rightarrow \dot{e}_\theta = -k_\theta e_\theta \quad (3.8)$$

De esta forma, la dinámica del error de la orientación viene definida por:

$$\frac{de_\theta}{dt} = -k_\theta e_\theta \Rightarrow \int_0^t \frac{de_\theta}{e_\theta} = dt \Rightarrow e_\theta(t) = e^{-k_\theta t} e_\theta(0) \quad (3.9)$$

donde se puede observar que $k_\theta > 0$ garantiza que el error se anule cuando $t \rightarrow \infty$.

Ahora, se considerarán los errores e_x y e_y .

Considerando la serie de Taylor truncada al primer orden de $\cos \theta$ y de $\sin \theta$ en torno a θ_{ref} se tiene:

$$\begin{cases} \cos \theta = \cos \theta_{ref} - \sin \theta_{ref} (\theta - \theta_{ref}) \\ \sin \theta = \sin \theta_{ref} + \cos \theta_{ref} (\theta - \theta_{ref}) \end{cases} \quad (3.10)$$

Combinando (3.1) y (3.7)

$$\begin{cases} \dot{x}(t) = V \cos \theta = (\Delta x \cos \theta_{ref} + \Delta y \sin \theta_{ref}) \cos \theta \\ \dot{y}(t) = V \sin \theta = (\Delta x \cos \theta_{ref} + \Delta y \sin \theta_{ref}) \sin \theta \end{cases} \quad (3.11)$$

Según (3.6)

$$(\dot{y}_{ref} + k_y e_y) \cos \theta_{ref} = (\dot{x}_{ref} + k_x e_x) \sin \theta_{ref} \quad (3.12)$$

De forma que desarrollando (3.11) y sustituyendo (3.12) se llega a:

$$\begin{cases} \dot{e}_x = -k_x e_x - f_\gamma e_\theta \\ \dot{e}_y = -k_y e_y - f_\Psi e_\theta \end{cases} \quad (3.13)$$

donde f_γ y f_Ψ son función de $k_x, k_y, e_x, e_y, \theta_{ref}, \dot{x}_{ref}$ y \dot{y}_{ref} .

Como se ha demostrado anteriormente $t \rightarrow \infty \Rightarrow e_\theta \rightarrow 0$, por tanto, se puede observar que para $k_x, k_y > 0$ los errores e_x y e_y tienden a cero cuando $t \rightarrow \infty$.

Por tanto, como se ha demostrado los errores tienden a cero con el paso del tiempo bajo la condición de que se tomen unos parámetros de controlador positivos.

3.1.3 Resultados de Simulaciones

Se han realizado una serie de simulaciones en Simulink siguiendo distintas trayectorias para comprobar el funcionamiento del controlador diseñado en distintos entornos.

Los parámetros de los controladores escogidos para las simulaciones son los que se muestran a continuación:

$$k_x = k_y = k_\theta = 0.95$$

El diagrama del controlador, así como los del resto de controladores se encuentra en el apéndice A.

Recta en OX

Como primera simulación se ha realizado una trayectoria recta en OX con una velocidad lineal de 5 m/s , el objetivo era comprobar el seguimiento de una recta con la que comienza alineado.

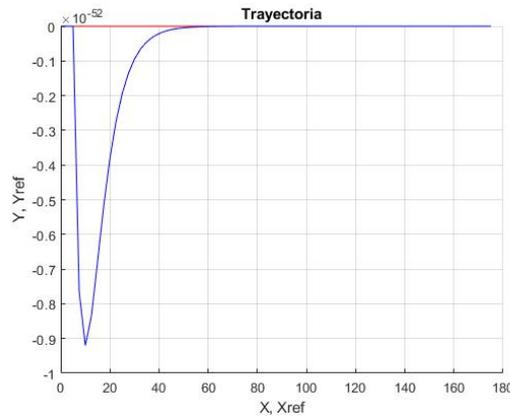


Figura 3.2 Trayectoria realizada por el robot móvil.

Se puede observar en los resultados de la simulación que el robot sigue la trayectoria de forma correcta aunque se puede observar una muy pequeña perturbación en la dirección OY, esta perturbación es negligible y puede asociarse a errores numéricos, adicionalmente esta se suprime rápidamente.

Si se observa el error se puede ver que tiende a cero tal y como se esperaba según lo demostrado en el apartado anterior.

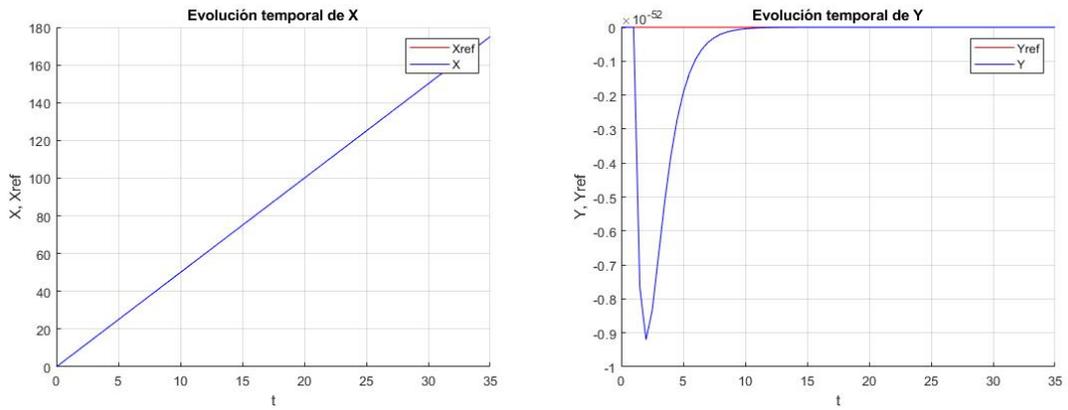


Figura 3.3 Evolución temporal de las variables de posición.

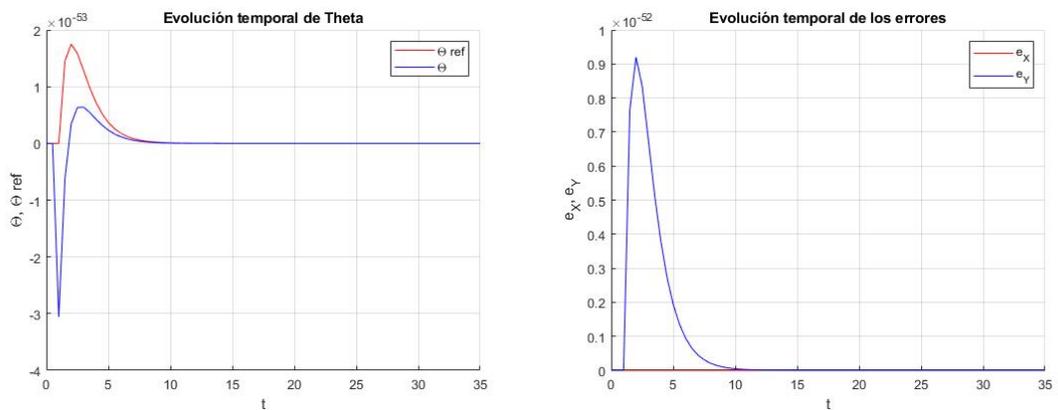


Figura 3.4 Evolución temporal de la orientación y los errores.

Recta en OY

Como segunda simulación se ha tomado como guía una recta en la dirección OY, nuevamente con una velocidad de 5 m/s, esto se hace para comprobar como lidia el controlador con una trayectoria recta con la que no está alineada y se ha supuesto el peor caso posible.

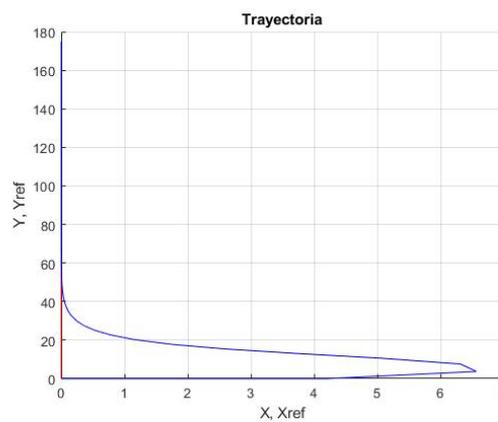


Figura 3.5 Trayectoria realizada por el robot móvil.

De los resultados se puede obtener como el controlador avanza girando hasta alinearse con la recta de referencia, además puede observarse como esto se consigue con bastante agilidad, apenas unos 10 s.

Por otro lado, se puede observar como inicialmente se tienen unos errores perceptibles, estos son debidos a la trayectoria de corrección, una vez se termina esta corrección, los errores decaen a cero como se esperaba.

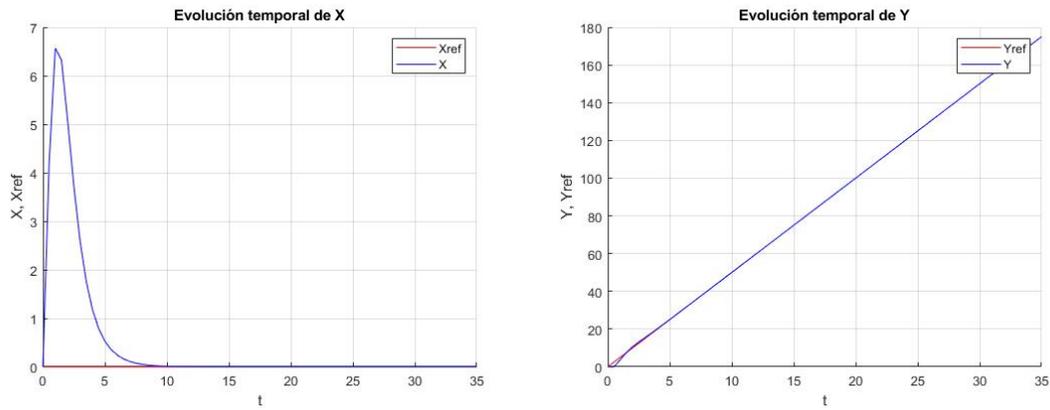


Figura 3.6 Evolución temporal de las variables de posición.

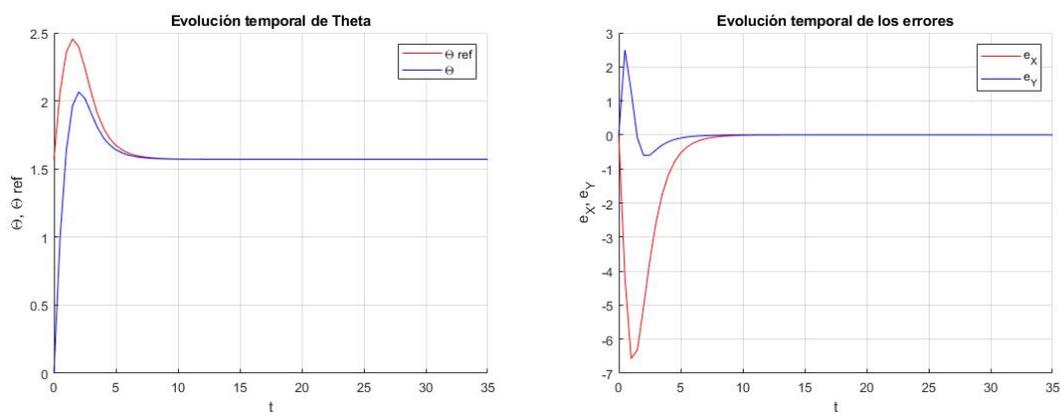


Figura 3.7 Evolución temporal de la orientación y los errores.

Circunferencia

Como tercera simulación se ha tomado como guía una circunferencia con un radio de 100 m y un periodo de 100 s con el objetivo de ver el comportamiento del controlador frente a una trayectoria que no comienza en el mismo punto y que tiene una velocidad de giro constante.

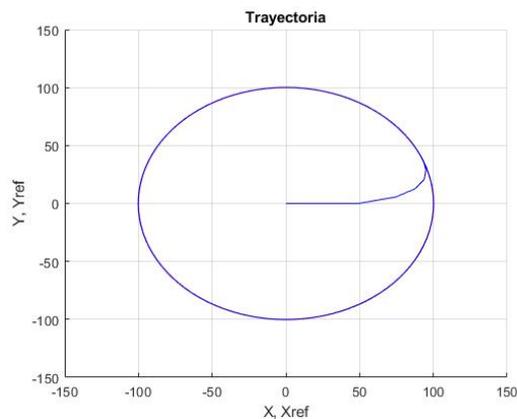


Figura 3.8 Trayectoria realizada por el robot móvil.

Puede observarse que el robot alcanza rápidamente la trayectoria (~ 10 s) y una vez lo hace la sigue sin problema alguno, anulándose el error durante todo el resto de la trayectoria.

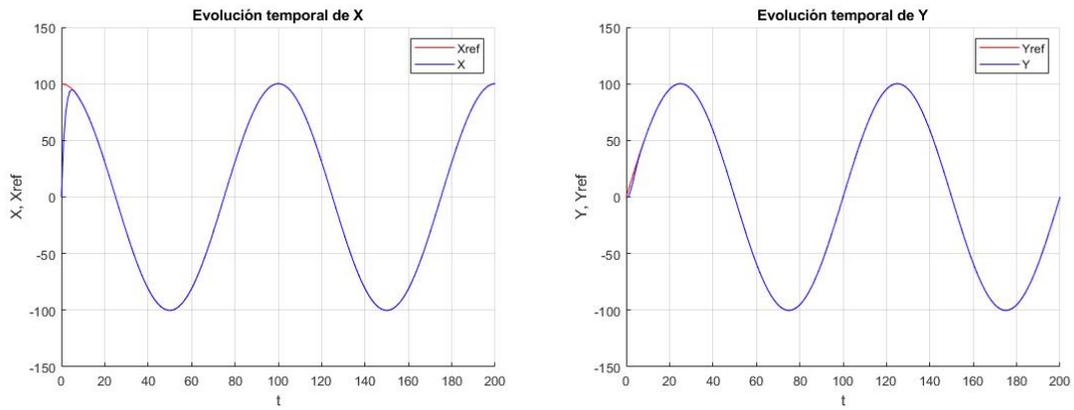


Figura 3.9 Evolución temporal de las variables de posición.

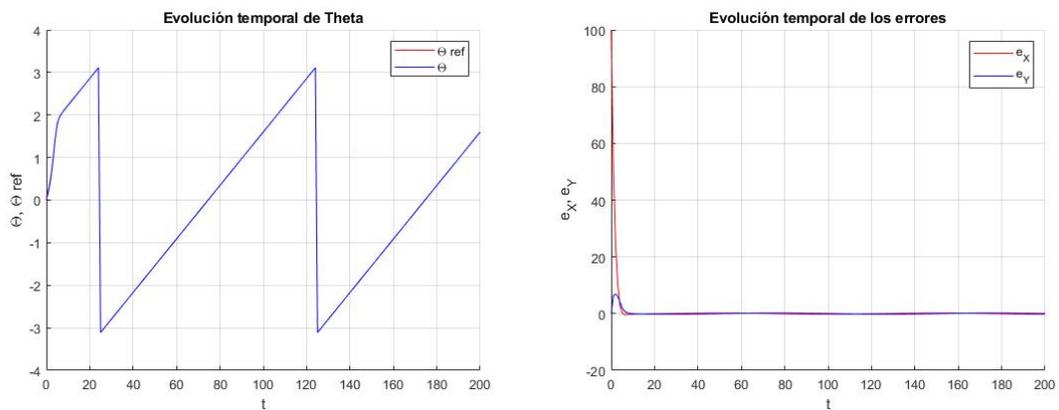


Figura 3.10 Evolución temporal de la orientación y los errores.

Lemniscata

Por último se ha simulado siguiendo una lemniscata con radio de 100 m y un periodo de 100 s, esta última simulación busca comprobar el funcionamiento del controlador frente a una referencia compleja, con la que tiene que hacer un acercamiento y tiene una orientación con variación no constante.

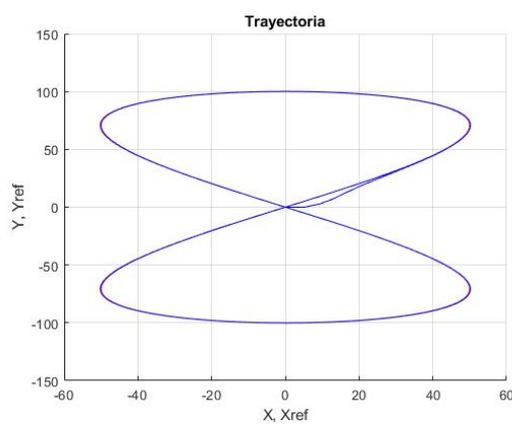


Figura 3.11 Trayectoria realizada por el robot móvil.

Puede visualizarse en los resultados que el robot es capaz de realizar el acercamiento correctamente y posteriormente seguir la trayectoria correctamente.

Como esta se considera la trayectoria más desfavorable, el resto de controladores se someterán exclusivamente a esta simulación y a la circunferencia, ya que esta última tiene una mayor trayectoria de acercamiento.

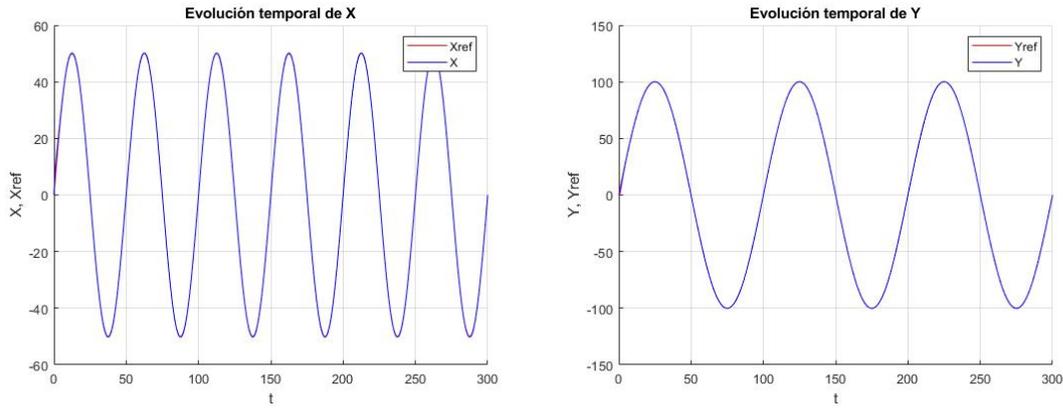


Figura 3.12 Evolución temporal de las variables de posición.

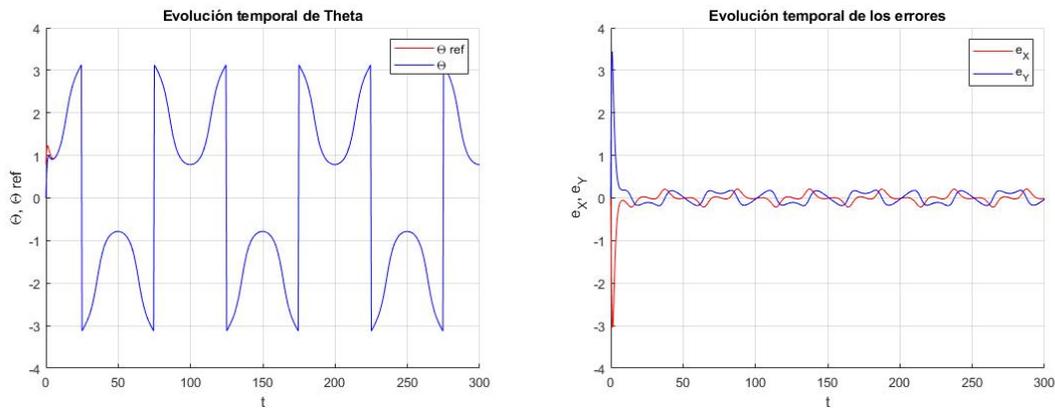


Figura 3.13 Evolución temporal de la orientación y los errores.

3.2 Control Cinemático en DT (Aprox. de Euler)

3.2.1 Diseño del Controlador

Para el diseño de este controlador se ha partido del mismo modelo que para el control en CT:

$$\begin{cases} \dot{x} = V \cos \theta \\ \dot{y} = V \sin \theta \\ \dot{\theta} = W \end{cases} \quad (3.14)$$

Para la aplicación de controlador en CT se procede a discretizar el modelo anterior, para este primer controlador se discretizará utilizando la aproximación de Euler, de forma que el modelo quedará de la siguiente forma en CT:

$$\begin{cases} x_{n+1} = x_n + TV_n \cos \theta_n \\ y_{n+1} = y_n + TV_n \sin \theta_n \\ \theta_{n+1} = \theta_n + TW_n \end{cases} \quad (3.15)$$

La cual puede ser reordenada en la siguiente forma matricial:

$$\begin{bmatrix} x_{n+1} \\ y_{n+1} \\ \theta_{n+1} \end{bmatrix} = \begin{bmatrix} x_n \\ y_n \\ \theta_n \end{bmatrix} + T \begin{bmatrix} \cos \theta_n & 0 \\ \sin \theta_n & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} V_n \\ W_n \end{bmatrix} \quad (3.16)$$

Si se supone un acercamiento suave y proporcional al error como se ha desarrollado anteriormente (aunque sería posible tomar otro enfoque para el acercamiento), se llega al siguiente sistema:

$$\begin{bmatrix} \cos \theta_n & 0 \\ \sin \theta_n & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} V_n \\ W_n \end{bmatrix} = \frac{1}{T} \begin{bmatrix} x_{ref,n+1} - k_x(x_{ref,n} - x_n) - x_n \\ y_{ref,n+1} - k_y(y_{ref,n} - y_n) - y_n \\ \theta_{ref,n+1} - k_\theta(\theta_{ref,n} - \theta_n) - \theta_n \end{bmatrix} \quad (3.17)$$

De (3.17) se puede obtener el valor de $\theta_{ref,n}$ que permite cumplir las condiciones para que el sistema tenga solución exacta:

$$\tan \theta_{ref,n} = \frac{y_{ref,n+1} - k_y(y_{ref,n} - y_n) - y_n}{x_{ref,n+1} - k_x(x_{ref,n} - x_n) - x_n} \quad (3.18)$$

La orientación obtenida θ_{ref} representa la orientación necesaria para que el robot móvil alcance la trayectoria objetivo.

Finalmente, la ley de control obtenida tras resolver el el sistema (3.17) es:

$$\begin{bmatrix} V_n \\ W_n \end{bmatrix} = \frac{1}{T} \begin{bmatrix} [x_{ref,n+1} - k_x(x_{ref,n} - x_n) - x_n] \cos \theta_{ref,n} + [y_{ref,n+1} - k_y(y_{ref,n} - y_n) - y_n] \sin \theta_{ref,n} \\ \theta_{ref,n+1} - k_\theta(\theta_{ref,n} - \theta_n) - \theta_n \end{bmatrix} \quad (3.19)$$

Es importante tener en cuenta que (3.18) da el valor de la referencia de la orientación en el instante n , sin embargo para la computación de (3.19) es necesaria también la referencia para el instante $n + 1$, para ello será necesario estimar este valor según los conocimientos de la serie $\{\theta_{ref,i}\}$.

En el controlador que tratará en el siguiente apartado se aproximará utilizando el acercamiento suave aproximando las posiciones de $n + 1$ a las de n .

$$\tan \theta_{ref,n+1} \simeq \frac{y_{ref,n+2} - k_y(y_{ref,n+1} - y_n) - y_n}{x_{ref,n+2} - k_x(x_{ref,n+1} - x_n) - x_n} \quad (3.20)$$

Donde, las referencias en el instante $n + 2$ son conocidas y proporcionadas al controlador.

3.2.2 Resultados de la Simulación

A continuación se realizan las simulaciones utilizando Simulink para comprobar el funcionamiento del controlador desarrollado.

Para estas simulaciones se escogen los siguientes valores para los parámetros:

$$k_x = k_y = 0.94 \quad k_\theta = 0.92$$

Nuevamente, el diagrama del controlador utilizado para la simulación se encuentra en el apéndice ??.

Circunferencia

La primera referencia a seguir se trata de una circunferencia de radio 100 m y periodo 100 s, esta simulación tiene como objetivo principal observar el comportamiento del robot cuando tiene que realizar una maniobra de acercamiento apreciable.

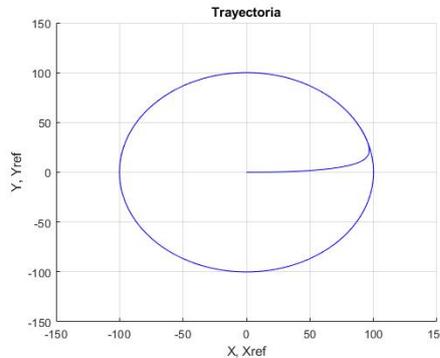


Figura 3.14 Trayectoria realizada por el robot móvil.

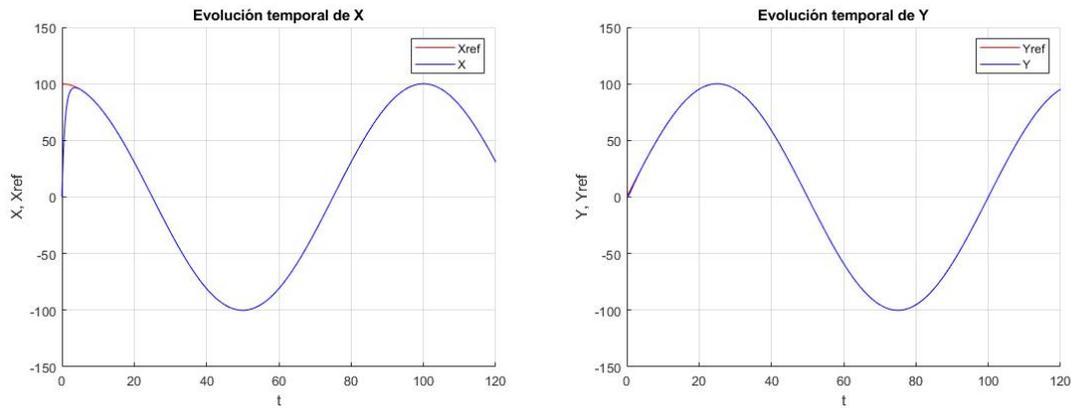


Figura 3.15 Evolución temporal de las variables de posición.

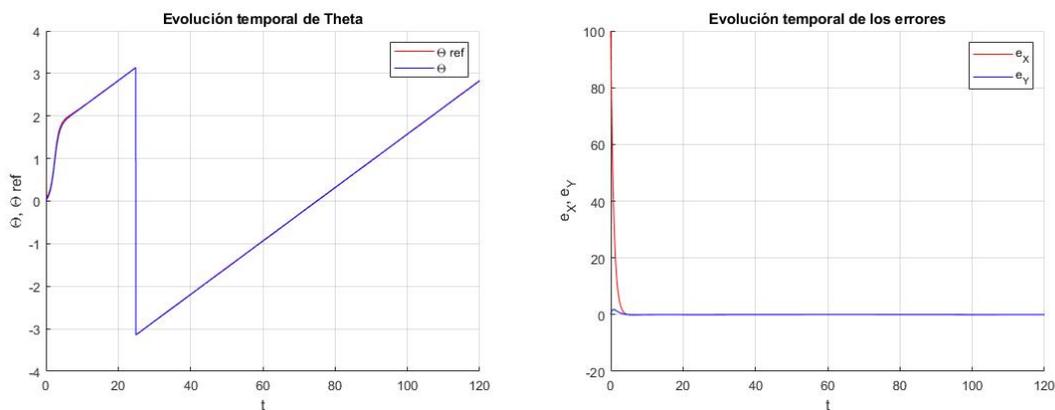


Figura 3.16 Evolución temporal de la orientación y los errores.

En los resultados se puede apreciar la trayectoria de acercamiento que se realiza en ~ 10 s, tras esto, se sigue la trayectoria sin problema y con un error nulo.

Lemniscata

La segunda trayectoria objetivo de simulación es la lemniscata con radio de 100 m y periodo de 100 s, con el objetivo de comprobar el funcionamiento frente a una referencia más compleja.

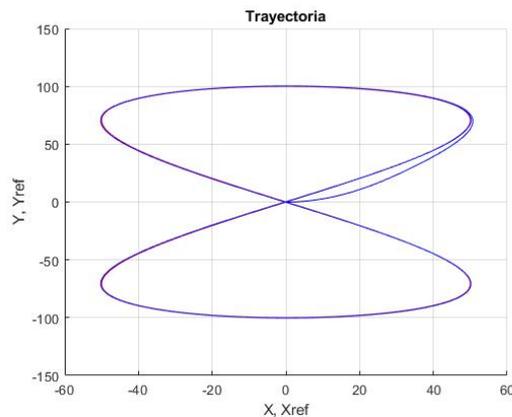


Figura 3.17 Trayectoria realizada por el robot móvil.

Tras una trayectoria de acercamiento de unos 20 s, se alcanza la referencia deseada y se sigue con unos errores pequeños ($e \sim 0.1$ m).

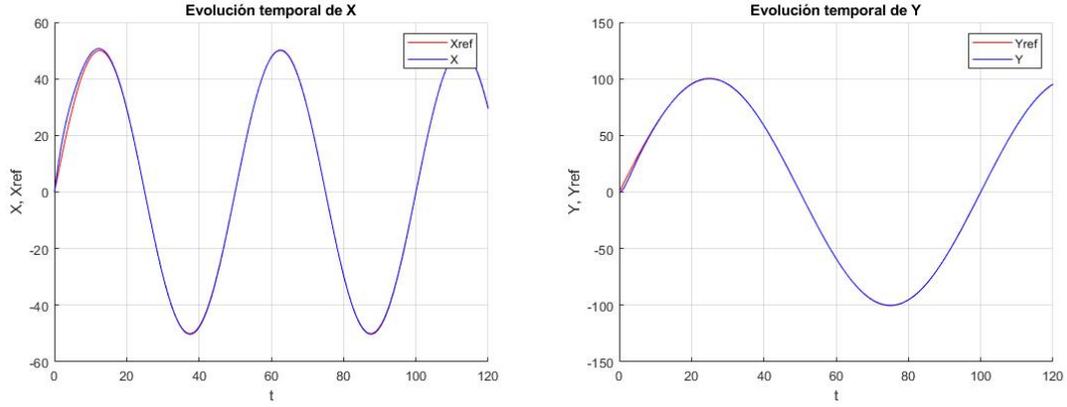


Figura 3.18 Evolución temporal de las variables de posición.

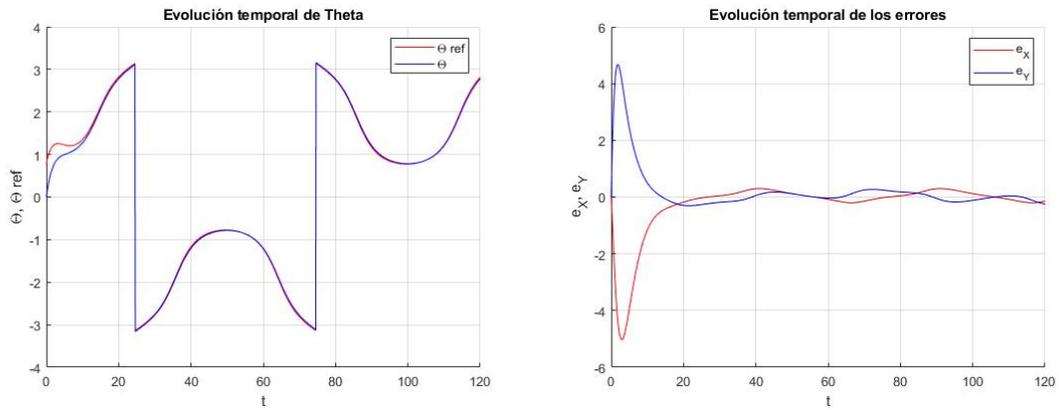


Figura 3.19 Evolución temporal de la orientación y los errores.

3.3 Control Cinemático en DT (Aprox. por Trapecios)

3.3.1 Diseño del Controlador

A modo de comparativa se ha procedido a discretizar el modelo (3.14) utilizando la aproximación por trapecios en lugar de la aproximación de Euler, y con ello realizar el controlador para esta discretización y ver como esta afecta al mismo.

Con esta discretización se llega al modelo discretizado en forma matricial (3.21).

$$\begin{bmatrix} x_{n+1} \\ y_{n+1} \\ \theta_{n+1} \end{bmatrix} = \begin{bmatrix} x_n \\ y_n \\ \theta_n \end{bmatrix} + \frac{T}{2} \times \left(\begin{bmatrix} \cos \theta_{n+1} & 0 \\ \sin \theta_{n+1} & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} V_{n+1} \\ W_{n+1} \end{bmatrix} + \begin{bmatrix} \cos \theta_n & 0 \\ \sin \theta_n & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} V_n \\ W_n \end{bmatrix} \right) \quad (3.21)$$

Despejando el modelo matricial con las acciones de control de (3.21) se llega a

$$\begin{bmatrix} \cos \theta_{n+1} & 0 \\ \sin \theta_{n+1} & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} V_{n+1} \\ W_{n+1} \end{bmatrix} = \begin{bmatrix} \frac{2}{T}(x_{n+1} - x_n) - V_n \cos \theta_n \\ \frac{2}{T}(y_{n+1} - y_n) - V_n \sin \theta_n \\ \frac{2}{T}(\theta_{n+1} - \theta_n) - W_n \end{bmatrix} \quad (3.22)$$

que, si se asume un acercamiento suave proporcional al error, puede reescribirse como

$$\begin{bmatrix} \cos \theta_{n+1} & 0 \\ \sin \theta_{n+1} & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} V_{n+1} \\ W_{n+1} \end{bmatrix} = \begin{bmatrix} \frac{2\Delta x}{T} - V_n \cos \theta_n \\ \frac{2\Delta y}{T} - V_n \sin \theta_n \\ \frac{2\Delta \theta}{T} - W_n \end{bmatrix} \quad (3.23)$$

donde

$$\begin{cases} \Delta x = x_{ref,n+1} - k_x(x_{ref,n} - x_n) - x_n \\ \Delta y = y_{ref,n+1} - k_y(y_{ref,n} - y_n) - y_n \\ \Delta \theta = \theta_{ref,n+1} - k_\theta(\theta_{ref,n} - \theta_n) - \theta_n \end{cases} \quad (3.24)$$

Puede observarse que en este caso las acciones de control que se computan en cada instante son las del instante siguiente no las del mismo instante como ocurría en los controladores anteriores, de forma que se deben guardar en una memoria para utilizarlas cuando son necesarias.

Buscando las condiciones necesarias para que (3.23) tenga una solución exacta, se llega al valor necesario para la referencia, $\theta_{ref,n+1}$.

$$\tan \theta_{ref,n+1} = \frac{\Delta y - V_n \sin \theta_n}{\Delta x - V_n \cos \theta_n} \quad (3.25)$$

Por último, se resuelve (3.23) mediante mínimos cuadrados y se llega a las acciones de control necesarias.

$$\begin{cases} V_{n+1} = \left(\frac{2\Delta x}{T} - V_n \cos \theta_n \right) \cos \theta_{ref,n+1} + \left(\frac{2\Delta y}{T} - V_n \sin \theta_n \right) \sin \theta_{ref,n+1} \\ W_{n+1} = \frac{2\Delta \theta}{T} - W_n \end{cases} \quad (3.26)$$

3.3.2 Resultados de la Simulación

Se realizarán las simulaciones en Simulink para las comprobaciones de funcionamiento con los mismos valores para los parámetros que en la discretización utilizando Euler, es decir:

$$k_x = k_y = 0.94 \quad k_\theta = 0.92$$

Además se ha utilizado un tiempo de muestreo de 0.05 s.

Circunferencia

Como en los casos anteriores se comienza con la referencia de la circunferencia de radio 100 m y periodo 100 s para observar la trayectoria de acercamiento y su efecto.

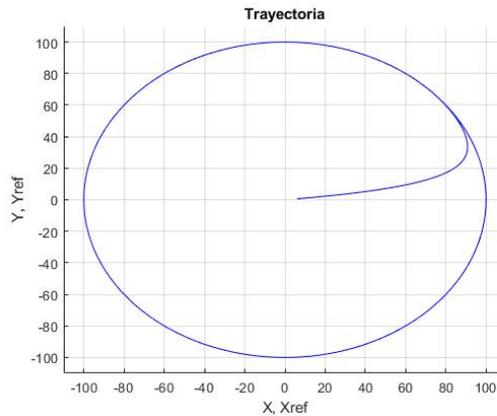


Figura 3.20 Trayectoria realizada por el robot móvil.

Observando los resultados puede verse que sigue la trayectoria con gran fidelidad y tiene un muy pequeño tiempo de trayectoria de acercamiento, la gráfica de la orientación se ve muy sucia, esto es debido a perturbaciones en el modelo que hacen que varíe rápidamente, sin embargo, en un robot real, la señal se filtrará y se seguirá una referencia continua.

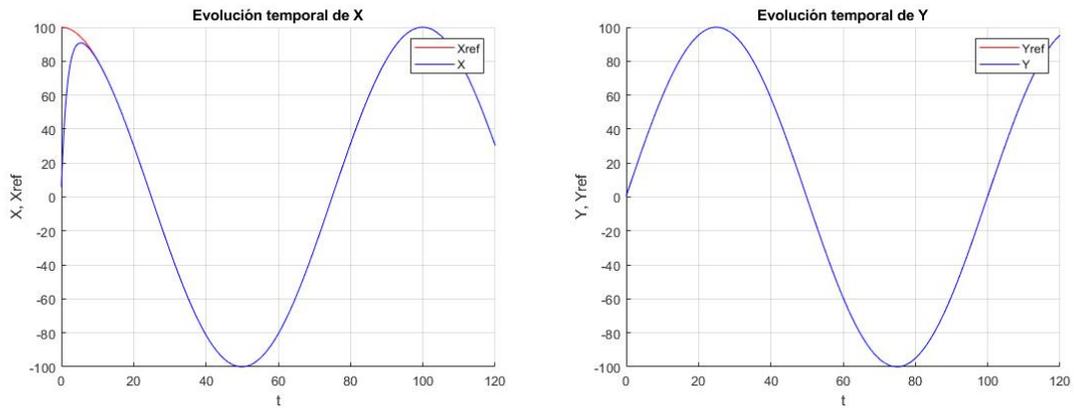


Figura 3.21 Evolución temporal de las variables de posición.

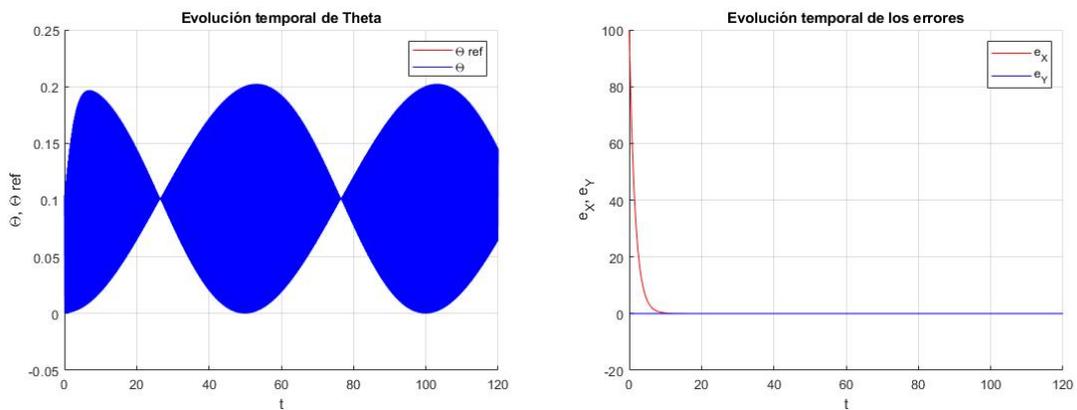


Figura 3.22 Evolución temporal de la orientación y los errores.

Lemniscata

Se estudia a continuación la simulación con la lemniscata como referencia.

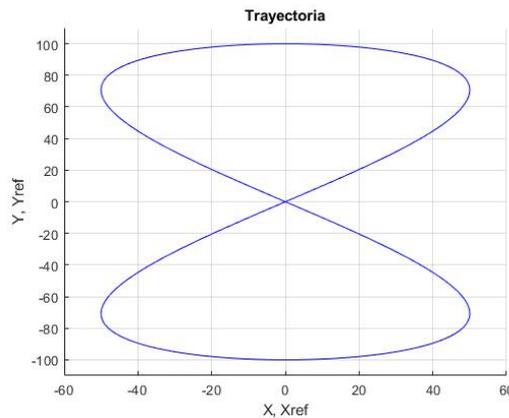


Figura 3.23 Trayectoria realizada por el robot móvil.

En el caso de la lemniscata se puede ver como el robot se gira rápidamente en el sitio para seguir la trayectoria, de forma que no existe una trayectoria de acercamiento, en todo momento se sigue la trayectoria con un error muy reducido.

Al igual que en el caso anterior la orientación tiene unas muy rápidas oscilaciones que serán filtradas en el robot real para su uso.

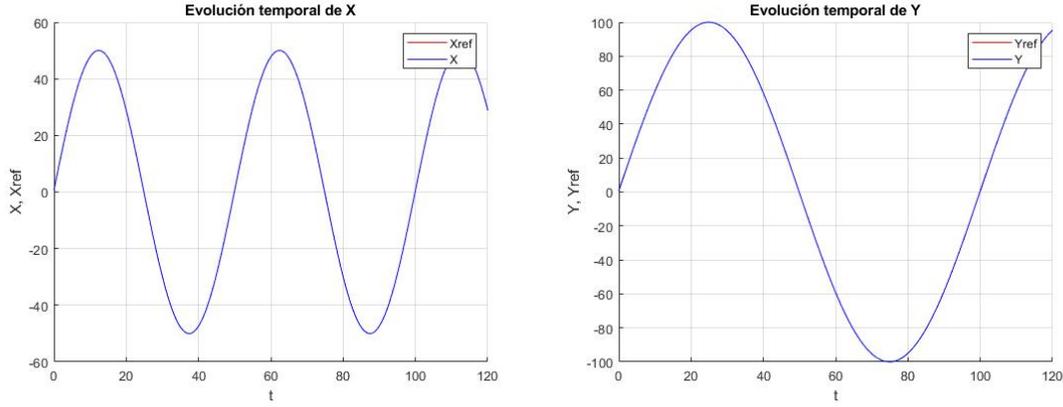


Figura 3.24 Evolución temporal de las variables de posición.

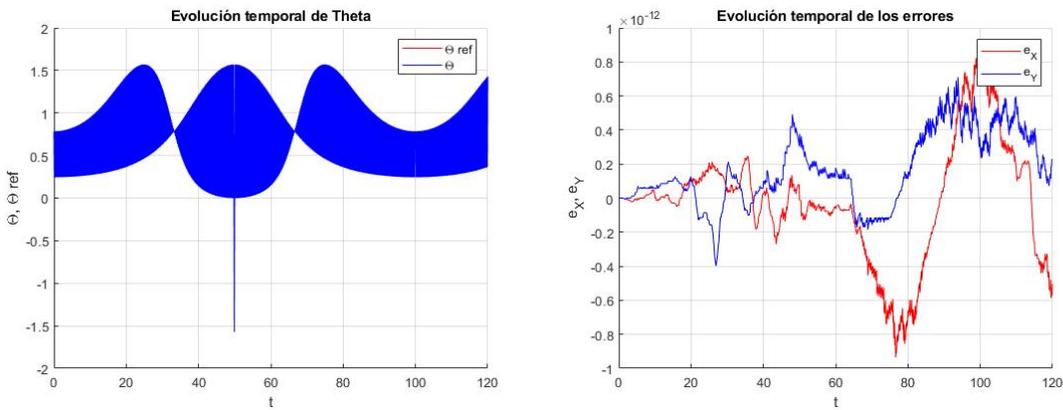


Figura 3.25 Evolución temporal de la orientación y los errores.

3.4 Modelo Dinámico en DT

El último modelo de robot móvil para el cual se va a diseñar un controlador es el modelo dinámico. El modelo dinámico es más complejo que los modelos cinemáticos anteriores, sin embargo, es necesario utilizarlo cuando se van a realizar tareas que requieran cargas grandes y/o altas velocidades.

El modelo dinámico dependerá además de las características dinámicas del robot como la masa y el centro de inercia, por ello, se diseña para un robot concreto. El modelo a analizar será el modelo dinámico no holonómico de un robot similar a un unicycle (De La Cruz & Carelli, 2008), el cual se muestra en (3.27) y en la figura 3.26.

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\psi} \\ \dot{u} \\ \dot{\omega} \end{bmatrix} = \begin{bmatrix} u \cos \psi - a\omega \sin \psi \\ u \sin \psi + a\omega \cos \psi \\ \omega \\ \frac{\theta_3}{\theta_1} \omega^2 - \frac{\theta_4}{\theta_1} u \\ -\frac{\theta_5}{\theta_2} u \omega - \frac{\theta_6}{\theta_2} \omega \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ \frac{1}{\theta_1} & 0 \\ 0 & \frac{1}{\theta_2} \end{bmatrix} \begin{bmatrix} uc \\ \omega c \end{bmatrix} + \begin{bmatrix} \delta_x \\ \delta_y \\ 0 \\ \delta_u \\ \delta_\omega \end{bmatrix} \quad (3.27)$$

donde (x,y) son las coordenadas cartesianas del robot, ψ es la orientación del mismo, u es la velocidad lineal, ω es la velocidad angular y los parámetros θ son los parámetros del modelo dinámico:

$$\begin{aligned} \theta_1 &= \left(\frac{R_a}{k_a} (mR_t r + 2I_e) + 2rk_{DT} \right) = 0.24089 & \theta_4 &= \frac{R_a}{k_a} \left(\frac{k_a k_b}{R_a} + B_e \right) / (rk_{PT} + 1) = 0.99629 \\ \theta_2 &= \left(\frac{R_a}{k_a} (I_e d^2 + 2R_t r (I_z + mbP2)) + 2rdk_{DR} \right) / (2rdk_{PR}) = 0.2424 & \theta_5 &= \frac{R_a}{k_a} mbR_t / (dk_{PR}) = -0.0037256 \\ \theta_3 &= \frac{R_a}{k_a} mbR_t / (2k_{PT}) = -0.00093603 & \theta_6 &= \frac{R_a}{k_a} \left(\frac{k_a k_b}{R_a} + B_e \right) d / (2rk_{PR} + 1) = 1.0915 \end{aligned} \quad (3.28)$$

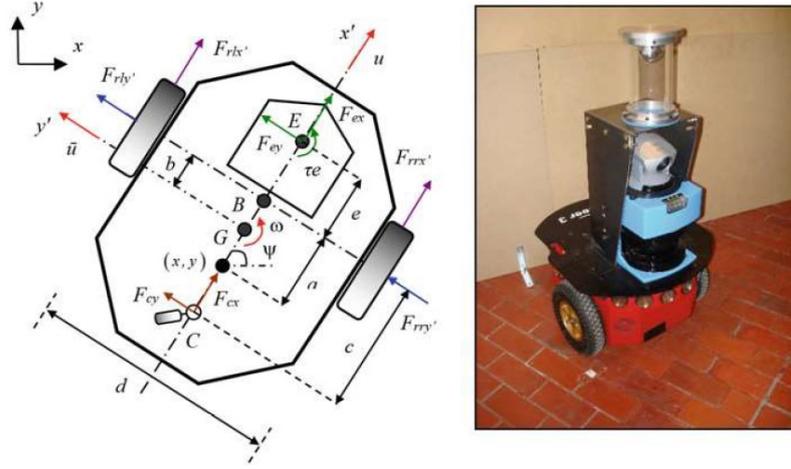


Figura 3.26 Robot móvil PIONNER 3DX, modelo y equipamiento de laboratorio.

donde m es la masa del robot; r es el radio de las ruedas laterales; k_b es la constante de voltaje multiplicada por el ratio de los engranajes; R_a es la constante de resistencia eléctrica; k_a es la constante de torsión multiplicada por el ratio de los engranajes; k_{PR} , k_{PT} , y k_{DT} son constantes positivas; I_e y B_e son el momento de inercia y el coeficiente de fricción viscosa total, respectivamente; y R_l es el radio nominal del neumático.

El vector $\delta = [\delta_x \ \delta_y \ 0 \ \delta_u \ \delta_\omega]^T$ es el vector de incertidumbre relacionadas con el robot móvil, donde δ_x y δ_y dependen de las velocidades debidas al deslizamiento de las ruedas y la orientación del robot y δ_u y δ_ω dependen de parámetros mecánicos del robot. Todos estos parámetros serán considerados como perturbaciones del modelo.

3.4.1 Diseño del Controlador

Para el diseño del controlador utilizando la metodología LAB CD en DT, se comienza a partir de una discretización aproximada del modelo dinámico (3.27) (Rosales, Scaglia, Mut & di Sciascio, 2009)

$$\begin{bmatrix} x_{n+1} \\ y_{n+1} \\ \Psi_{n+1} \\ u_{n+1} \\ \omega_{n+1} \end{bmatrix} = \begin{bmatrix} x_n \\ y_n \\ \Psi_n \\ u_n \\ \omega_n \end{bmatrix} + T \left\{ \begin{bmatrix} u_n \cos \Psi_n \\ u_n \sin \Psi_n \\ \omega_n \\ \frac{\theta_3}{\theta_1} \omega_n^2 - \frac{\theta_4}{\theta_1} u_n \\ -\frac{\theta_5}{\theta_2} u_n \omega_n - \frac{\theta_6}{\theta_2} \omega_n \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ \frac{1}{\theta_1} & 0 \\ 0 & \frac{1}{\theta_2} \end{bmatrix} \begin{bmatrix} u_{c,n} \\ \omega_{c,n} \end{bmatrix} \right\} \quad (3.29)$$

En este caso hay tres variables sacrificadas, Ψ_n , u_n y ω_n . Entonces, asumiendo acercamiento suave proporcional al error se llega al sistema (3.30)

$$\begin{bmatrix} \Delta x_n \\ \Delta y_n \\ \Delta \Psi_n \\ \Delta u_n \\ \Delta \omega_n \end{bmatrix} = \begin{bmatrix} x_{ref,n+1} - k_x(x_{ref,n} - x_n) - x_n \\ y_{ref,n+1} - k_y(y_{ref,n} - y_n) - y_n \\ \Psi_{ref,n+1} - k_\Psi(\Psi_{ref,n} - \Psi_n) - \Psi_n \\ u_{ref,n+1} - k_u(u_{ref,n} - u_n) - u_n \\ \omega_{ref,n+1} - k_\omega(\omega_{ref,n} - \omega_n) - \omega_n \end{bmatrix} = T \left\{ \begin{bmatrix} u_{ref,n} \cos \Psi_{ref,n} \\ u_{ref,n} \sin \Psi_{ref,n} \\ \omega_{ref,n} \\ \frac{\theta_3}{\theta_1} \omega_n^2 - \frac{\theta_4}{\theta_1} u_n \\ -\frac{\theta_5}{\theta_2} u_n \omega_n - \frac{\theta_6}{\theta_2} \omega_n \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ \frac{1}{\theta_1} & 0 \\ 0 & \frac{1}{\theta_2} \end{bmatrix} \begin{bmatrix} u_{c,n} \\ \omega_{c,n} \end{bmatrix} \right\} \quad (3.30)$$

A partir de (3.30) se obtienen las condiciones para que el sistema tenga una solución exacta. De las tres primeras filas del sistema se tiene:

$$\begin{bmatrix} \cos \Psi_{ref,n} & 0 \\ \sin \Psi_{ref,n} & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} u_{ref,n} \\ \omega_{ref,n} \end{bmatrix} = \frac{1}{T} \begin{bmatrix} \Delta x_n \\ \Delta y_n \\ \Delta \Psi_n \end{bmatrix} \quad (3.31)$$

De (3.31) se obtienen los valores de referencia de las variables sacrificadas, y analizando las condiciones para que el sistema tenga una solución exacta se llega a

$$\tan \psi_{ref,n} = \frac{\Delta y_n}{\Delta x_n} \tag{3.32}$$

Resolviendo mediante mínimos cuadrados (3.31) se llega a

$$\begin{bmatrix} u_{ref,n} \\ \omega_{ref,n} \end{bmatrix} = \frac{1}{T} \begin{bmatrix} \Delta x_n \cos \psi_{ref,n} + \Delta y_n \sin \psi_{ref,n} \\ \Delta \psi_n \end{bmatrix} \tag{3.33}$$

Para calcular (3.33) es necesario el valor de $\psi_{ref,n+1}$. Para los cálculos, su valor se aproximará al que tiene en el instante n , es decir:

$$\psi_{ref,n,1} \simeq \psi_{ref,n} \tag{3.34}$$

Por último, las acciones de control del sistema se computan resolviendo (3.27) por mínimos cuadrados, llegando a (3.35)

$$\begin{bmatrix} u_{c,n} \\ \omega_{c,n} \end{bmatrix} = \begin{bmatrix} \theta_1 \left(\frac{\Delta u_n}{T} - \left(\frac{\theta_3}{\theta_1} \omega_n^2 - \frac{\theta_4}{\theta_1} u_n \right) \right) \\ \theta_2 \left(\frac{\Delta \omega_n}{T} - \left(-\frac{\theta_5}{\theta_2} u_n \omega_n - \frac{\theta_6}{\theta_2} \omega_n \right) \right) \end{bmatrix} \tag{3.35}$$

donde $0 < k_x, k_y, k_\psi, k_u, k_\omega < 1$ y $e_{Z,n} = Z_{ref,n} - Z,n$, siendo Z una variable de estado cualquiera.

3.4.2 Resultados de la Simulación

Se realizan una serie de simulaciones en Simulink con un tiempo de muestreo de 0.05 s para comprobar el funcionamiento del controlador y el efecto de tener en cuenta la dinámica en el modelo. Los parámetros escogidos para el controlador son:

$$k_x = k_y = 0.95 \quad k_u = k_\omega = 0.94 \quad k_\psi = 0.9$$

Circunferencia

Se vuelve a simular con una circunferencia de radio 100 m y periodo 100 s como referencia para poder apreciar como trata el controlador con las trayectorias de acercamiento y su efecto en la trayectoria posterior.

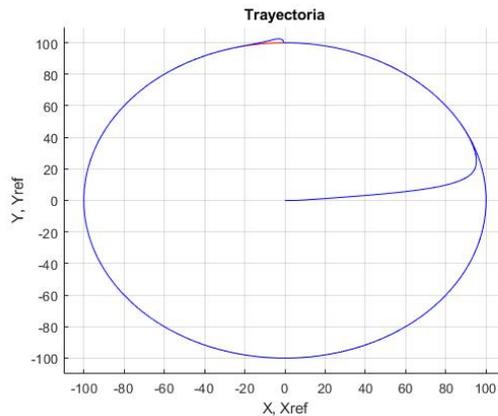


Figura 3.27 Trayectoria realizada por el robot móvil.

En los resultados puede visualizarse como realiza de forma correcta la trayectoria casi por completo, existiendo una perturbación puntual en y , esta perturbación ocurre cuando ψ pasa del 2º cuadrante al 3º cuadrante, el controlador hace que ψ muestre una discontinuidad en el ángulo ya que la referencia del mismo está acotada en $[-\pi, \pi]$ y la dinámica de ψ hace que aparezca una sobreoscilación en el ángulo que afecta al valor de y . A pesar de esto la perturbación es relativamente reducida y no afecta en exceso al comportamiento general, además, se podría buscar una solución que evitase este comportamiento, como asegurar la continuidad de ψ en la referencia.

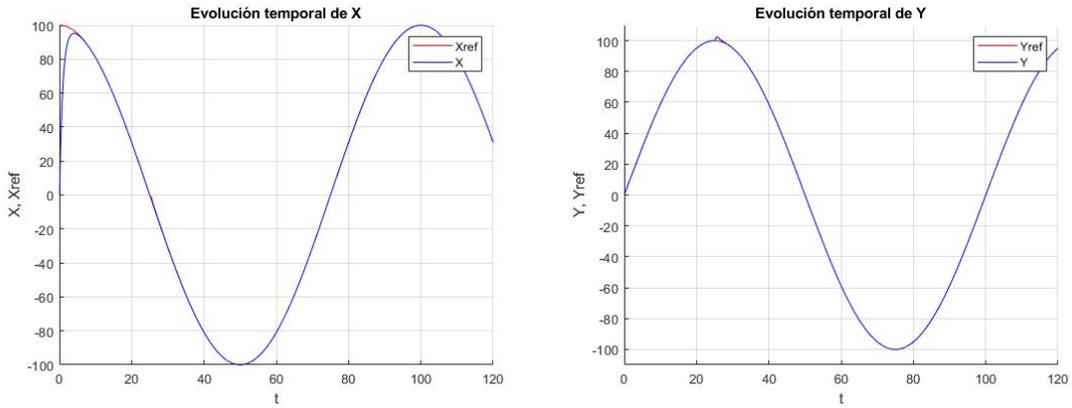


Figura 3.28 Evolución temporal de las variables de posición.

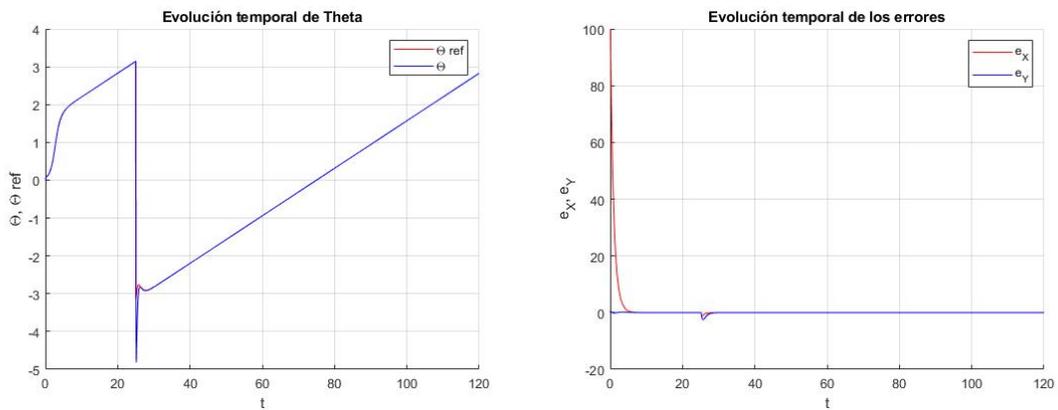


Figura 3.29 Evolución temporal de la orientación y los errores.

Se puede comprobar que la trayectoria de acercamiento se realiza sin problema y no presenta problemas de sobreoscilación ni provoca errores en el permanente.

Lemniscata

Se proporciona nuevamente como referencia una lemniscata con radio máximo 100 m y un periodo de 100 s, esto se hace para comprobar el funcionamiento frente a una referencia compleja.

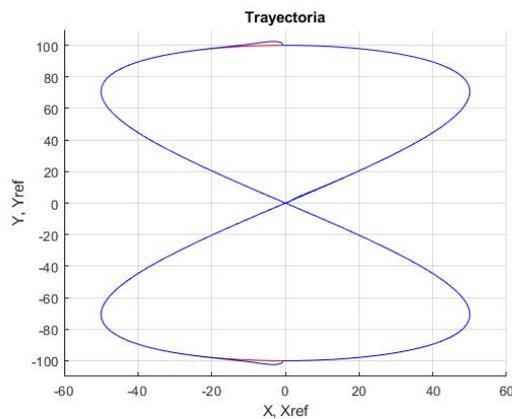


Figura 3.30 Trayectoria realizada por el robot móvil.

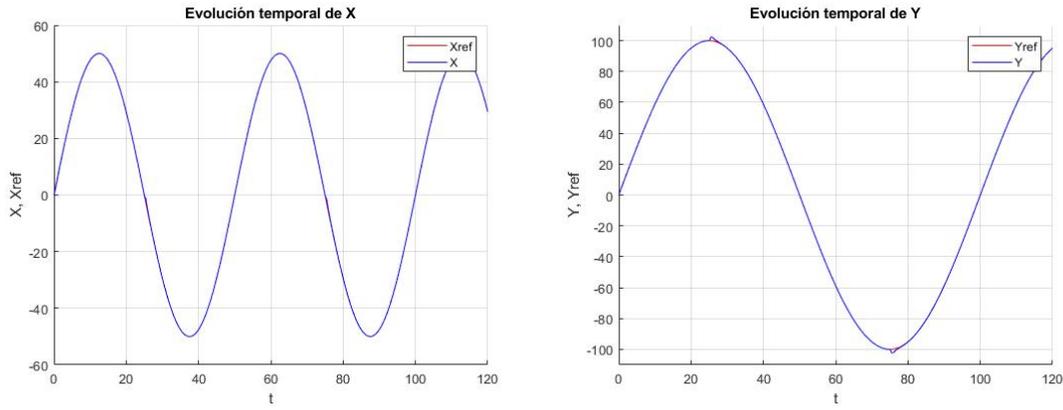


Figura 3.31 Evolución temporal de las variables de posición.

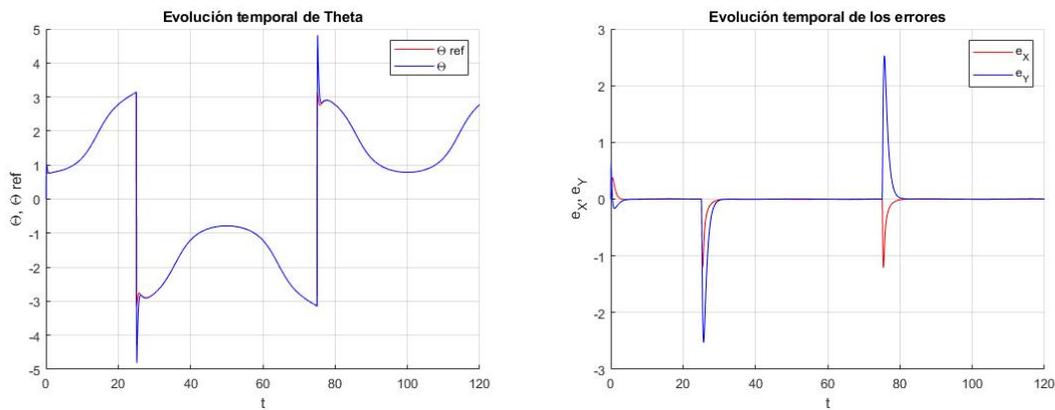


Figura 3.32 Evolución temporal de la orientación y los errores.

Observando los resultados se puede comprobar que existe el mismo comportamiento que para la referencia de circunferencia, se sigue de forma correcta la referencia dada a excepción de los errores que aparecen en y debidos, nuevamente, a la discontinuidad que aparece en la referencia de ψ .

3.5 Comparación de Controladores

Por último, se realiza una comparativa entre los distintos controladores creados para el control de un robot móvil.

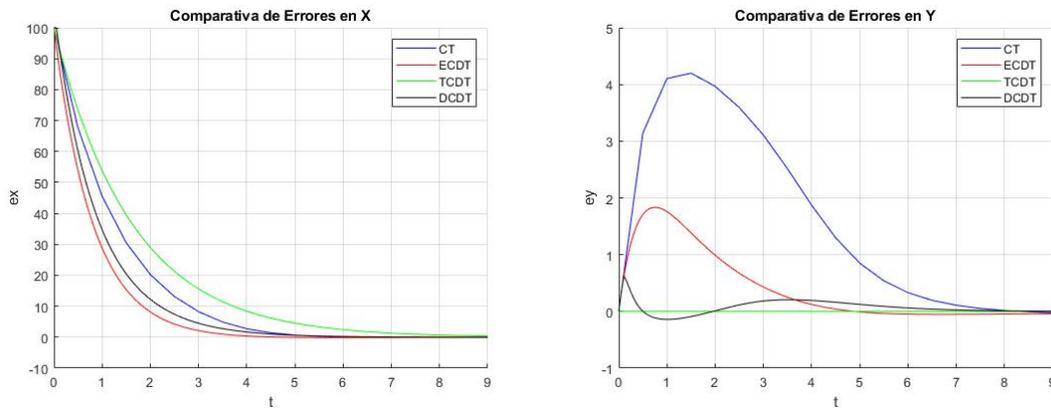


Figura 3.33 Comparativa de errores para la circunferencia durante la trayectoria de acercamiento.

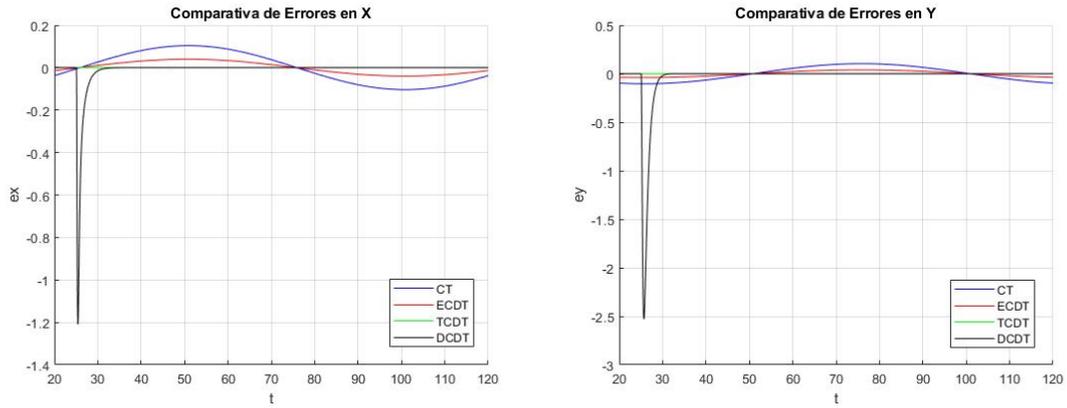


Figura 3.34 Comparativa de errores para la circunferencia en el permanente.

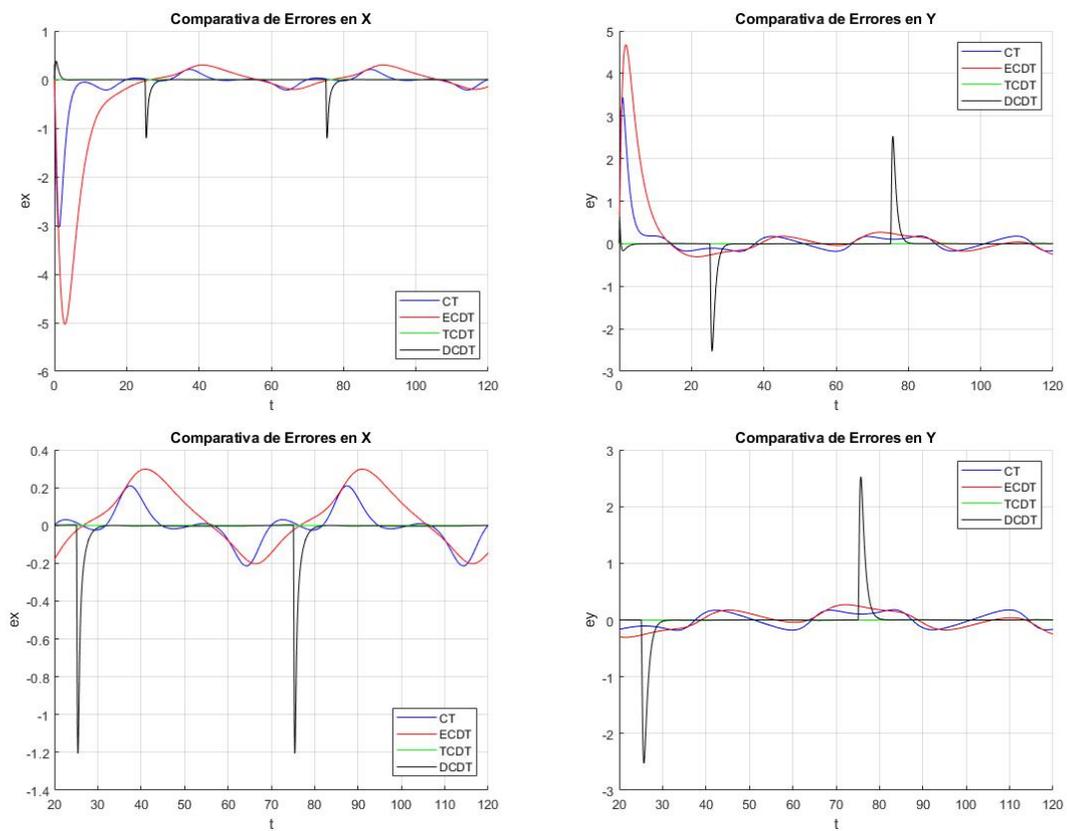


Figura 3.35 Comparativa de errores para la lemniscata.

Observando las gráficas de error se puede obtener información del comportamiento de cada controlador y permite comparar el funcionamiento de cada uno de forma visual.

Teniendo en cuenta la trayectoria de acercamiento con la circunferencia como referencia puede observarse que, aunque las diferencias temporales son mínimas, el controlador que realiza el acercamiento más rápidamente es el controlador que utiliza la aproximación de Euler para discretizar (o ECDT para abreviar), mientras que el más lento es que utiliza la aproximación por trapecios (o TCDT para abreviar).

En cuanto a la referencia de la lemniscata, ocurre lo contrario, siendo el más rápido el controlador TCDT y el controlador más lento ECDT.

Centrándose en el permanente puede observarse que los menores errores se tienen con el controlador TCDT y con el controlador que tiene en cuenta la dinámica (o DCDT), aunque este último solo si no se tiene en cuenta los grandes errores puntuales que genera el controlador debido a los errores en la referencia angular. Los mayores errores en el permanente los dan el controlador en tiempo continuo (o CT para abreviar) en el

caso de la circunferencia y el controlador ECDT en conjunto con CT para el caso de la lemniscata.

Teniendo en cuenta lo anterior se puede ver que los controladores que se crean mediante el LAB CD funcionan de forma muy correcta y los resultados mostrados en este último apartado permiten escoger el mejor controlador para el robot móvil a diseñar, aunque para ello se ha de tener en cuenta la carga computacional de cada modelo, que si se ordena de mayor a menor carga sería: DCDT, TCDT, ECDT, CT. Aunque también hay que tener presente el hecho de que el controlador CT no es viable para una implementación en un dispositivo discreto.

4 Aplicación a un Multirotor

En muchos casos la utilización de vehículos terrestres no tripulados (UGVs) como el robot móvil estudiado anteriormente es difícil debido al entorno de aplicación, para muchos de estos casos la utilización de vehículos aéreos no tripulados (UAVs) puede ser muy ventajosa debido a su capacidad de desplazamiento tridimensional.

Enfocándose en los UAVs se pueden encontrar diversos tipos distintos de los mismos: de ala fija, de ala rotatoria, dirigibles, de ala batiente, híbridos ala fija-rotatoria, etc. Si uno se centra en la realización de una tarea en la que se requiere de buena maniobrabilidad una muy buena opción es el uso de ala rotatoria, ya que además de ser capaces de maniobras muy precisas de tener una alta maniobrabilidad permiten la opción de *hovering*, aunque presentan la desventaja de que su mayor consumo de energía los lleva a unas menores autonomías.

Dentro de los UAV de ala fija se encuentran los multirotores que cuentan con una serie de rotores (superior a 3) unidas a un cuerpo central y que se encargan de mantener a la plataforma en el aire y sus diferencias de tracción de realizar las maniobras necesarias. Este capítulo se centrará en la aplicación del método LAB CD a un multirotor genérico (no se asume el número de rotores) con los rotores no inclinados, por lo que toda la maniobrabilidad la tiene mediante giros del vector de tracción total de los rotores.



Figura 4.1 Ejemplo de multirotor (DJI Matrice 600 Pro) y de los grados de libertad de un multirotor (Montgomery, 2014).

4.1 Controlador Simplificado

4.1.1 Diseño del Controlador

Inicialmente se realiza el controlador para el multirotor utilizando el modelo dinámico no lineal mostrado en (4.1) (Rosales, Gandolfo, Scaglia, Jordan, & Careli, 2015).

$$\begin{bmatrix} m\ddot{x} \\ m\ddot{y} \\ m\ddot{z} \\ \ddot{\phi} \\ \ddot{\theta} \\ \ddot{\psi} \end{bmatrix} = \begin{bmatrix} -u \sin \theta \\ u \cos \theta \sin \phi \\ \tau_{\phi} \\ \tau_{\theta} \\ \tau_{\psi} \end{bmatrix} \quad (4.1)$$

donde (x,y,z) representan las coordenadas del centro de masas con respecto al sistema de referencia inercial, (ϕ, θ, ψ) son los ángulos de Euler y $(u, \tau_{\phi}, \tau_{\theta}, \tau_{\psi})$ son las acciones de control, correspondientes a la tracción total, y los momentos en alabeo, cabeceo y guiñada respectivamente. ϕ es el ángulo de alabeo (roll), θ es el ángulo de cabeceo (pitch) y ψ es el ángulo de guiñada (yaw).

El objetivo del control es encontrar la combinación de acciones de control $(u, \tau_{\phi}, \tau_{\theta}, \tau_{\psi})$ que hace que el multirotor alcance y siga la trayectoria de referencia $(x_{ref}, y_{ref}, z_{ref}, \phi_{ref})$.

A continuación, se obtiene un modelo que representa la planta para discretizar posteriormente.

$$\begin{bmatrix} \dot{x} \\ \ddot{x} \\ \dot{y} \\ \ddot{y} \\ \dot{z} \\ \ddot{z} \\ \dot{\phi} \\ \ddot{\phi} \\ \dot{\theta} \\ \ddot{\theta} \\ \dot{\psi} \\ \ddot{\psi} \end{bmatrix} = \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \\ \dot{x}_5 \\ \dot{x}_6 \\ \dot{x}_7 \\ \dot{x}_8 \\ \dot{x}_9 \\ \dot{x}_{10} \\ \dot{x}_{11} \\ \dot{x}_{12} \end{bmatrix} = \begin{bmatrix} x_2 \\ -\frac{u}{m} \sin x_9 \\ x_4 \\ \frac{u}{m} \cos x_9 \sin x_7 \\ x_6 \\ \frac{u}{m} \cos x_9 \cos x_7 - g \\ x_8 \\ \tau_{\phi} \\ x_{10} \\ \tau_{\theta} \\ x_{12} \\ \tau_{\psi} \end{bmatrix} \quad (4.2)$$

Tras esto, se discretiza (4.2) utilizando la aproximación de Euler, llegando a (4.3).

$$\begin{bmatrix} -\frac{T}{m} \sin x_{9,n} & 0 & 0 & 0 \\ \frac{T}{m} \cos x_{9,n} \sin x_{7,n} & 0 & 0 & 0 \\ \frac{T}{m} \cos x_{9,n} \cos x_{7,n} & 0 & 0 & 0 \\ 0 & T & 0 & 0 \\ 0 & 0 & T & 0 \\ 0 & 0 & 0 & T \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} u_n \\ \tau_{\phi,n} \\ \tau_{\theta,n} \\ \tau_{\psi,n} \end{bmatrix} = \begin{bmatrix} x_{2,n+1} - x_{2,n} \\ x_{4,n+1} - x_{4,n} \\ x_{6,n+1} - x_{6,n} + gT \\ x_{8,n+1} - x_{8,n} \\ x_{10,n+1} - x_{10,n} \\ x_{12,n+1} - x_{12,n} \\ x_{1,n+1} - x_{1,n} - Tx_{2,n} \\ x_{3,n+1} - x_{3,n} - Tx_{4,n} \\ x_{5,n+1} - x_{5,n} - Tx_{6,n} \\ x_{7,n+1} - x_{7,n} - Tx_{8,n} \\ x_{9,n+1} - x_{9,n} - Tx_{10,n} \\ x_{10,n+1} - x_{10,n} - Tx_{12,n} \end{bmatrix} \quad (4.3)$$

Puede observarse que (4.3) se trata de un sistema algebraico lineal de la forma $Au = b$, siendo u el vector de acciones de control.

El siguiente paso consiste en despejar las variables en el instante de tiempo posterior, llegándose a (4.4).

$$\begin{bmatrix} x_{1,n+1} \\ x_{2,n+1} \\ x_{3,n+1} \\ x_{4,n+1} \\ x_{5,n+1} \\ x_{6,n+1} \\ x_{7,n+1} \\ x_{8,n+1} \\ x_{9,n+1} \\ x_{10,n+1} \\ x_{11,n+1} \\ x_{12,n+1} \end{bmatrix} = \begin{bmatrix} x_{1,n} + Tx_{2,n} \\ x_{2,n} - T\frac{u}{m} \sin x_{9,n} \\ x_{3,n} + Tx_{4,n} \\ x_{4,n} + T\frac{u}{m} \cos x_{9,n} \sin x_{7,n} \\ x_{5,n} + Tx_{6,n} \\ x_{6,n} + T\frac{u}{m} \cos x_{9,n} \cos x_{7,n} - gT \\ x_{7,n} + Tx_{8,n} \\ x_{8,n} + T\tau_{\phi,n} \\ x_{9,n} + Tx_{10,n} \\ x_{10,n} + T\tau_{\theta,n} \\ x_{11,n} + Tx_{12,n} \\ x_{12,n} + T\tau_{\psi,n} \end{bmatrix} \quad (4.4)$$

Como en los casos anteriores se asume un acercamiento a la referencia suave y proporcional al error. Esta aproximación lleva a los valores que deben cumplir ciertas referencias para que exista una solución exacta:

$$x_{2,ref,n} = \frac{x_{1,ref,n+1} - k_1(x_{1,ref,n} - x_{1,n}) - x_{1,n}}{T} = \frac{\Delta x_1}{T} \quad (4.5)$$

$$x_{4,ref,n} = \frac{x_{3,ref,n+1} - k_3(x_{3,ref,n} - x_{3,n}) - x_{3,n}}{T} = \frac{\Delta x_3}{T} \quad (4.6)$$

$$x_{6,ref,n} = \frac{x_{5,ref,n+1} - k_5(x_{5,ref,n} - x_{5,n}) - x_{5,n}}{T} = \frac{\Delta x_5}{T} \quad (4.7)$$

$$x_{12,ref,n} = \frac{x_{11,ref,n+1} - k_{11}(x_{11,ref,n} - x_{11,n}) - x_{11,n}}{T} = \frac{\Delta x_{11}}{T} \quad (4.8)$$

De forma análoga, se tienen los valores de referencia para las variables x_8 y x_{10} , dando lugar a:

$$x_{8,ref,n} = \frac{x_{7,ref,n+1} - k_7(x_{7,ref,n} - x_{7,n}) - x_{7,n}}{T} = \frac{\Delta x_7}{T} \quad (4.9)$$

$$x_{10,ref,n} = \frac{x_{9,ref,n+1} - k_9(x_{9,ref,n} - x_{9,n}) - x_{9,n}}{T} = \frac{\Delta x_9}{T} \quad (4.10)$$

Observando el sistema (4.3), se pueden ver que existe una serie de condiciones para obtener un valor exacto de u_n , ya que se tiene:

$$\begin{bmatrix} -\sin x_{9,n} \\ \cos x_{9,n} \sin x_{7,n} \\ \cos x_{9,n} \cos x_{7,n} \end{bmatrix} u_n = \frac{m}{T} \begin{bmatrix} \Delta x_{2,n} \\ \Delta x_{4,n} \\ \Delta x_{6,n} + gT \end{bmatrix} \quad (4.11)$$

donde

$$\Delta x_{i,n} = x_{i,ref,n+1} - k_i(x_{i,ref,n} - x_{i,n}) - x_{i,n} \quad (4.12)$$

Por tanto, para que (4.11) tenga solución exacta se tienen que cumplir las condiciones (4.13) y (4.14).

$$\tan x_{7,ref,n} = \frac{\Delta x_{4,n}}{\Delta x_{6,n} + gT} \quad (4.13)$$

$$\tan x_{9,ref,n} = -\frac{\Delta x_{2,n}}{\Delta x_{4,n}} \sin x_{7,ref,n} \quad (4.14)$$

Sin embargo, se comprueba que en el caso de que la referencia en la coordenada y sea alcanzada previamente, se tiene $\Delta x_{4,n} = 0$ y por tanto, $x_{7,n} = 0$ lo que hace que se anule su seno y con ello la referencia de x_9 , haciendo

que el multirrotor se detenga, para evitar esto, se toma (4.15) en lugar de (4.14) como ecuación para calcular $x_{9,ref,n}$ cuando se detecta que $\Delta x_{4,n} = 0$.

$$\tan x_{9,ref,n} = -\frac{\Delta x_{2,n}}{\Delta x_{6,n} + gT} \cos x_{7,n} \quad (4.15)$$

Tras esto, se aplica la resolución mediante mínimos cuadrados a (4.3) para hallar las acciones de control necesarias.

$$\begin{bmatrix} u_n \\ \tau_{\phi,n} \\ \tau_{\theta,n} \\ \tau_{\psi,n} \end{bmatrix} = \frac{1}{T} \begin{bmatrix} m \times [-\Delta x_{2,n} \sin x_{9,ref,n} + \Delta x_{4,n} \cos x_{9,ref,n} \sin x_{7,ref,n} + (\Delta x_{6,n} + gT) \cos x_{9,ref,n} \cos x_{7,ref,n}] \\ \Delta x_{8,n} \\ \Delta x_{10,n} \\ \Delta x_{12,n} \end{bmatrix} \quad (4.16)$$

4.1.2 Resultados de Simulación

Se realizan múltiples simulaciones en Simulink para distintas trayectorias de referencia con el objetivo de estudiar el comportamiento del controlador.

A continuación, se muestran los parámetros escogidos para el controlador diseñado.

$$\begin{array}{cccc} k_1 = 0.78 & k_2 = 0.8 & k_3 = 0.77 & k_4 = 0.77 \\ k_5 = 0.9 & k_6 = 0.95 & k_7 = 0.7 & k_8 = 0.67 \\ k_9 = 0.7 & k_{10} = 0.67 & k_{11} = 0.8 & k_{12} = 0.85 \end{array}$$

En el modelo se puede observar que la dinámica de x_{11} está desacoplada del resto del modelo, por lo que en el modelo se ha considerado un valor constante, $x_{11} = \frac{\pi}{2}$.

Recta en OX

La primera simulación realizada es una trayectoria recta en OX, con el objetivo de comprobar el seguimiento de una recta.

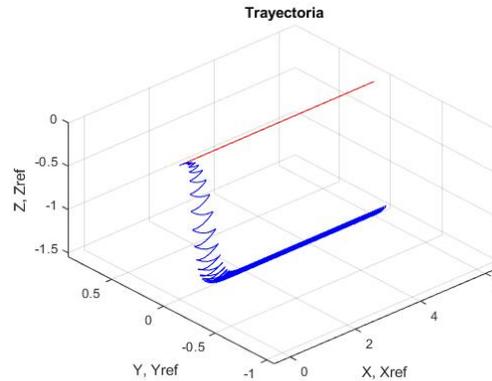


Figura 4.2 Trayectoria recorrida por el multirrotor.

Puede observarse en los resultados obtenidos que se sigue de una forma relativamente correcta la trayectoria dada, aunque presenta pequeñas oscilaciones del orden de 0.4 en la dirección OX.

También puede visualizarse la existencia de un error en régimen permanente para la altitud, sin embargo, al tratarse de ser en el permanente se puede solucionar añadiéndole este error a la referencia de Z, el error es de en torno a -11.7 . Existen adicionalmente perturbaciones en el eje OZ, sin embargo estas son muy reducidas (del orden de 0.1).

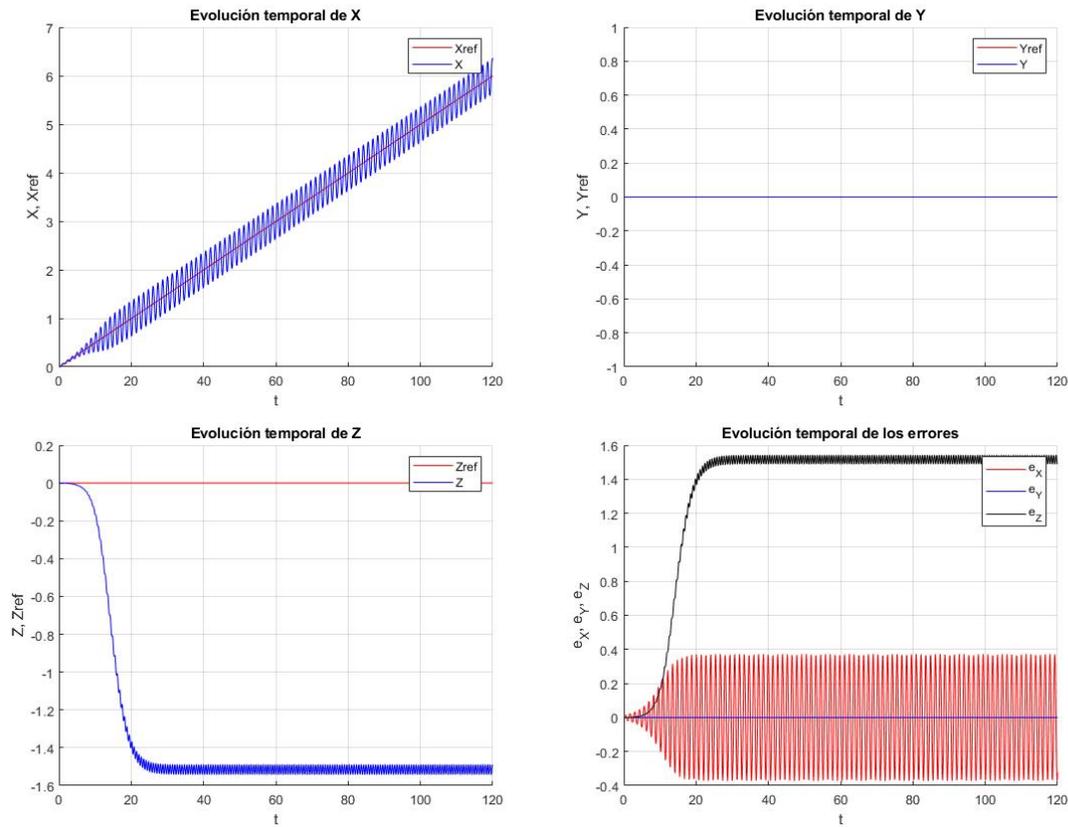


Figura 4.3 Evolución de las variables de posición y del error.

Recta en OY

A continuación, se simula una trayectoria recta en OY.

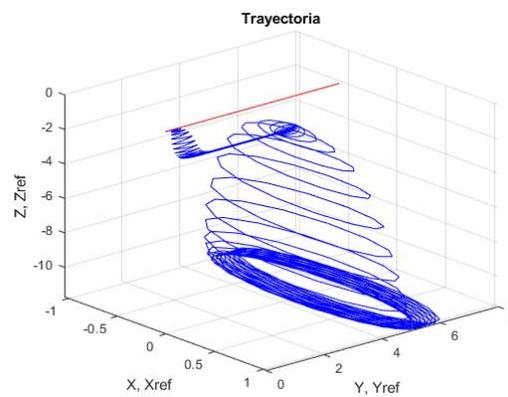


Figura 4.4 Trayectoria recorrida por el multirrotor.

En los resultados pueden observarse la existencia de oscilaciones en OX y OY, las cuales se intensifican tras 80 s, llegando al orden de la unidad.

En cuanto a la altitud se aprecia que el error inicial es de -2 m y después alcanza el error permanente que ocurría en la simulación anterior.

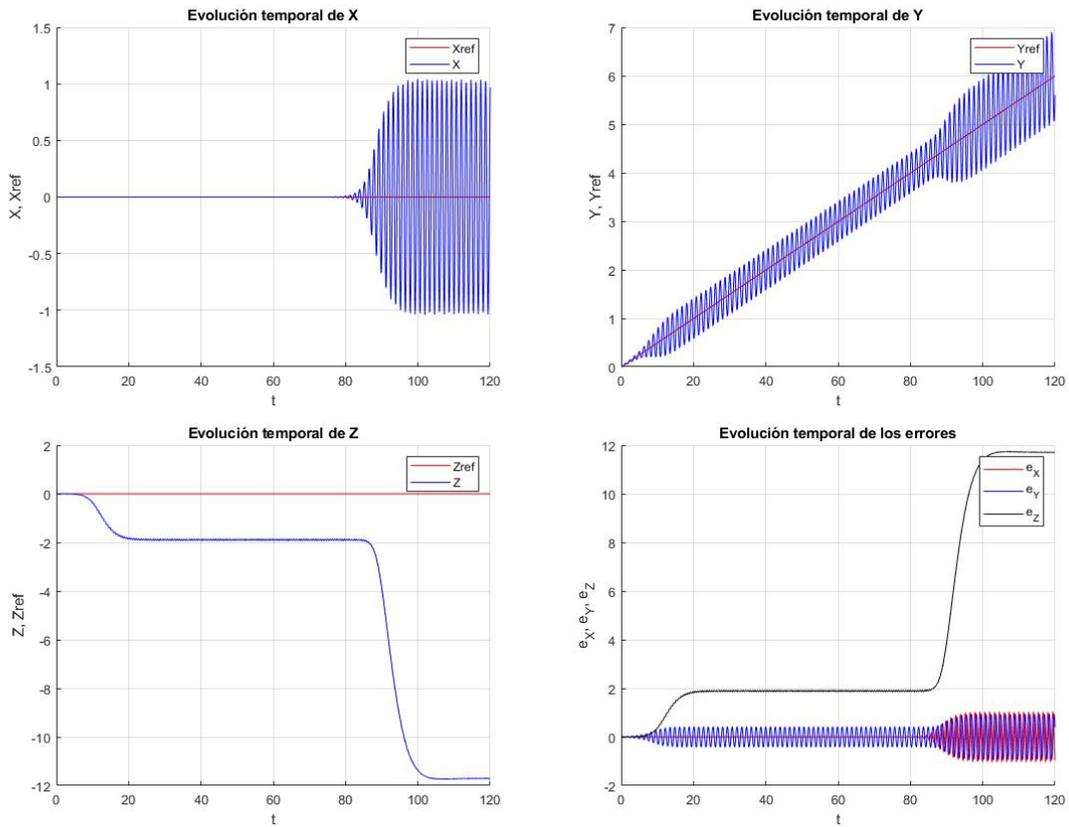


Figura 4.5 Evolución de las variables de posición y del error.

Ascenso Rectilíneo

La siguiente simulación consiste en un ascenso a velocidad constante, sin desplazamientos laterales.

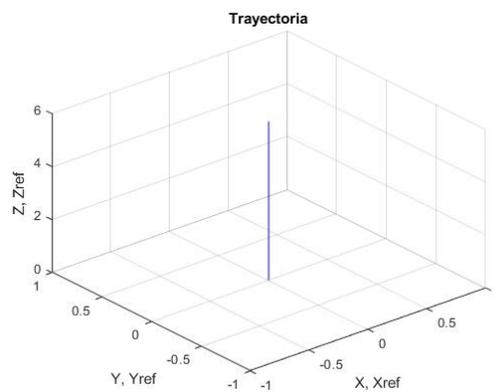


Figura 4.6 Trayectoria recorrida por el multirroto.

Se observa que cuando se toma como objetivo un ascenso sin desplazamientos laterales, la trayectoria se sigue de forma perfecta, no apareciendo ni las oscilaciones en OX, OY ni el error en régimen permanente en altitud.

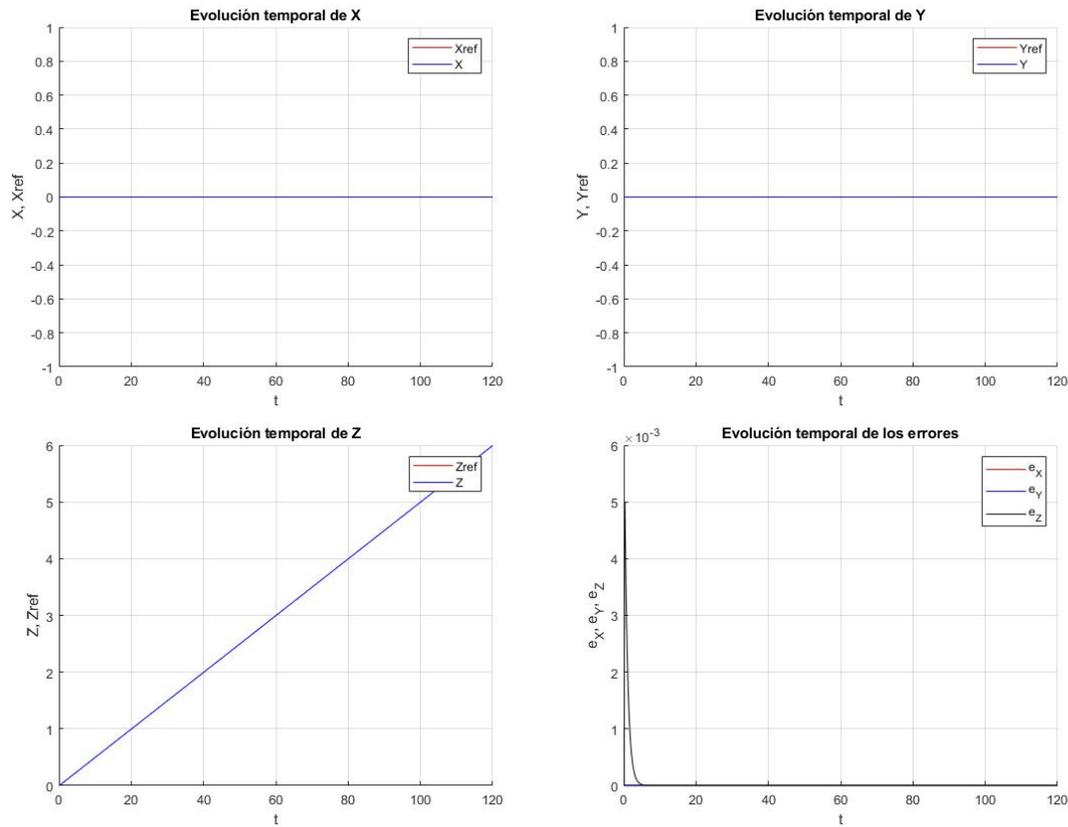


Figura 4.7 Evolución de las variables de posición y del error.

Lemniscata

A continuación, se realiza una trayectoria con una lemniscata de referencia en el plano horizontal y una altitud estática, esto se hace para comprobar el funcionamiento frente a una referencia compleja y las variaciones de altitud que esta provoca.

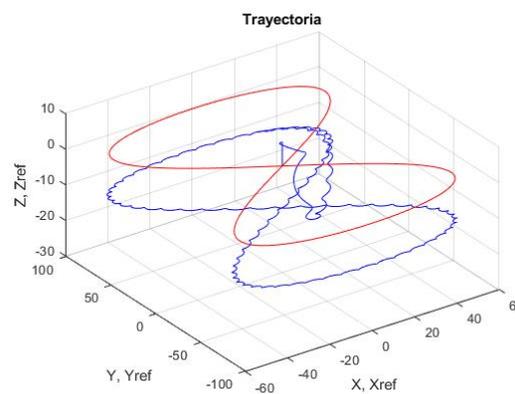


Figura 4.8 Trayectoria recorrida por el multirrotor.

Analizando los resultados se puede comprobar la existencia de una pequeña maniobra de transferencia, previa al seguimiento correcto de la trayectoria, tras esto se puede observar que el seguimiento de la trayectoria se realiza con oscilaciones en OX y OY del orden de 2 m .

En cuanto a la altitud, se observa un ascenso de unos 5 m y después una caída de unos 25 m para después quedarse con el error permanente que se daba en los primeros casos.

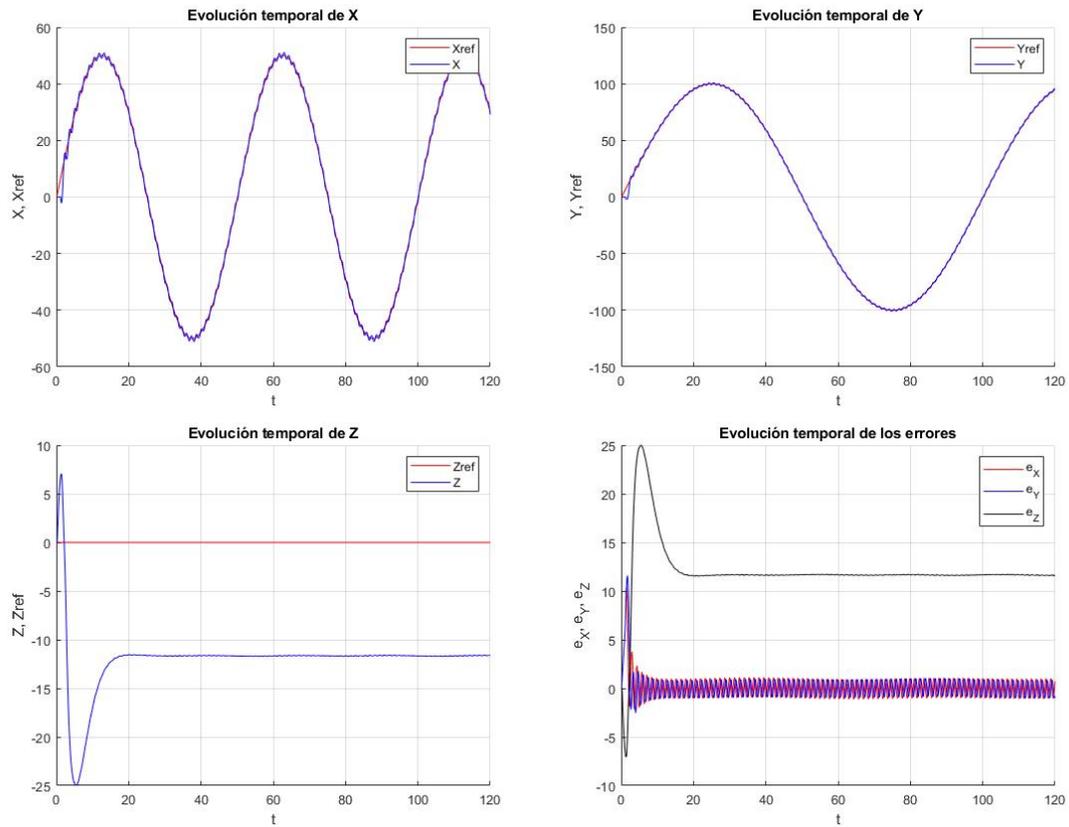


Figura 4.9 Evolución de las variables de posición y del error.

Ascenso Helicoidal

Por último, se da como referencia una trayectoria con ascenso helicoidal para estudiar el comportamiento del controlador frente a una referencia con desplazamiento coordinado en X-Y al mismo tiempo que asciende.

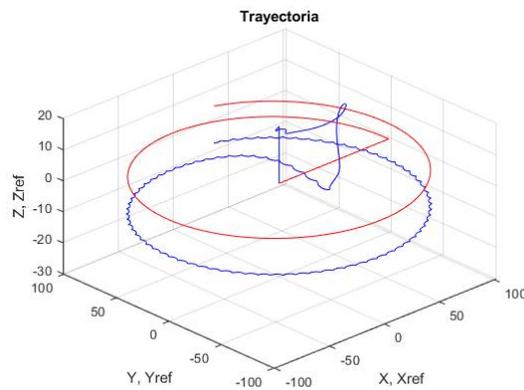


Figura 4.10 Trayectoria recorrida por el multirrotor.

Se observa la existencia de la maniobra de acercamiento den las coordenadas X e Y, por otro lado, se observa una oscilación en Z, tras la cual sigue un ascenso paralelo al programado pero con el mismo error permanente que en otros casos.

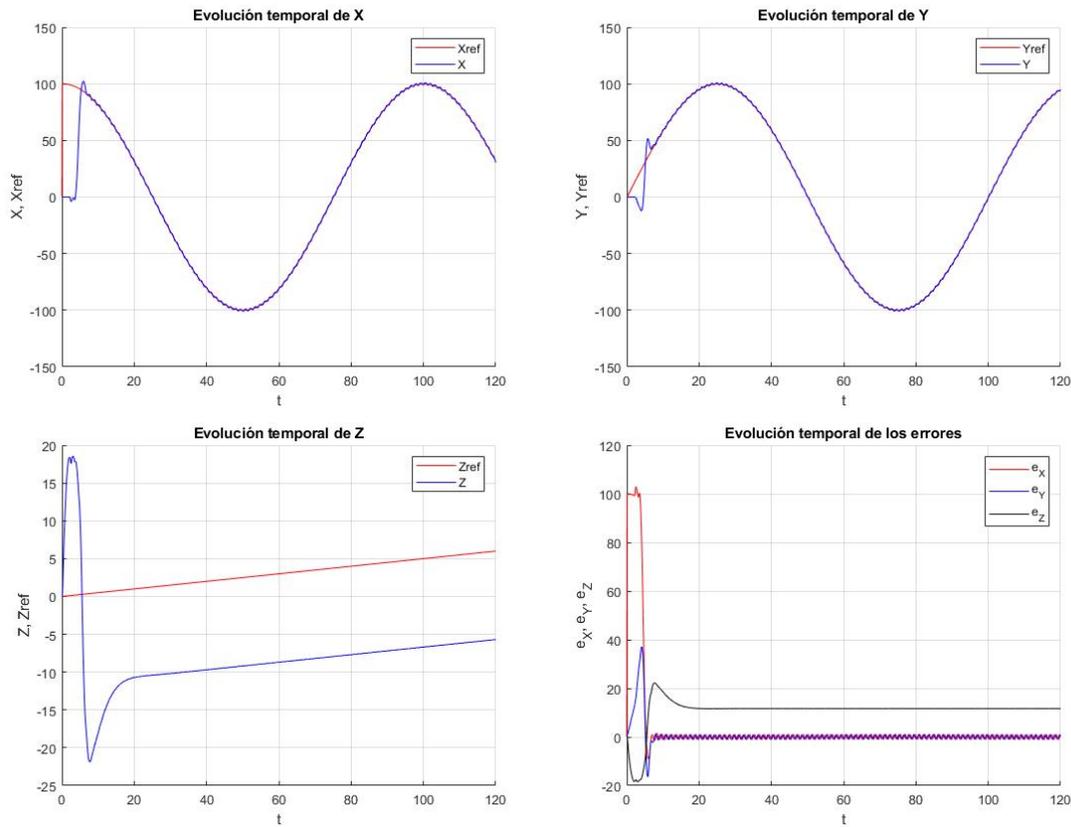


Figura 4.11 Evolución de las variables de posición y del error.

4.1.3 Mejoras del Controlador

Como se ha comentado anteriormente, el error permanente en altitud puede solucionarse calibrando con la adición de un término constante $+11.7 m$ en la referencia de la misma, lo cual reduce en gran medida el error en esta coordenada, tal y como puede visualizarse en las Figura 4.12 y 4.13.

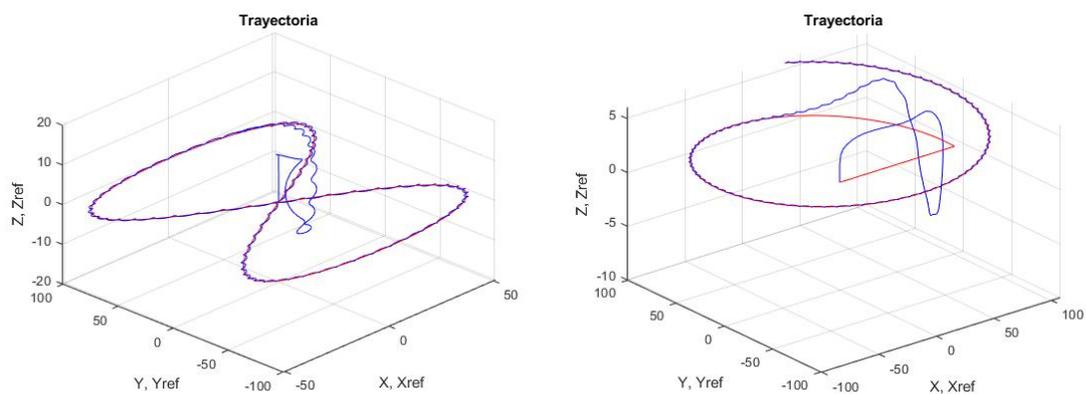


Figura 4.12 Trayectoria recorrida por el multirrotor tras calibrar para la lemniscata y el ascenso helicoidal.

Puede observarse que sigue existiendo una oscilación inicial en altitud, sin embargo desaparece el error permanente y, tras esto, se procede a seguir la trayectoria sin error en altitud.

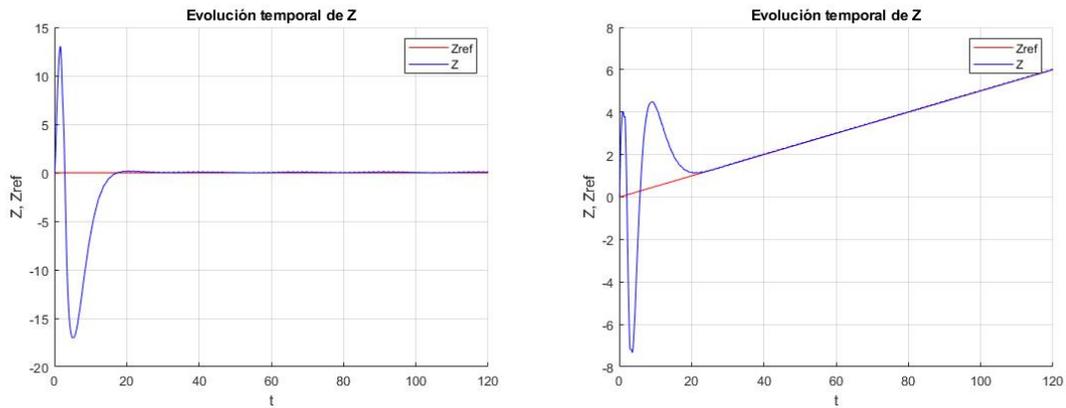


Figura 4.13 Evolución de la altitud tras calibrar para la lemniscata y el ascenso helicoidal.

Una propuesta para reducir las perturbaciones existentes en la trayectoria es el aumento de la frecuencia del reloj del controlador, en los resultados mostrados anteriormente se ha tomado un periodo de muestreo $T = 0.1 s$, en la Figura 4.14 se puede ver como se reduce el error si se reduce el tiempo de muestreo a $T = 0.01 s$ (tras hacer esto hay que modificar el término constante del calibrado a $+3 m$, al reducirse el error en el permanente).

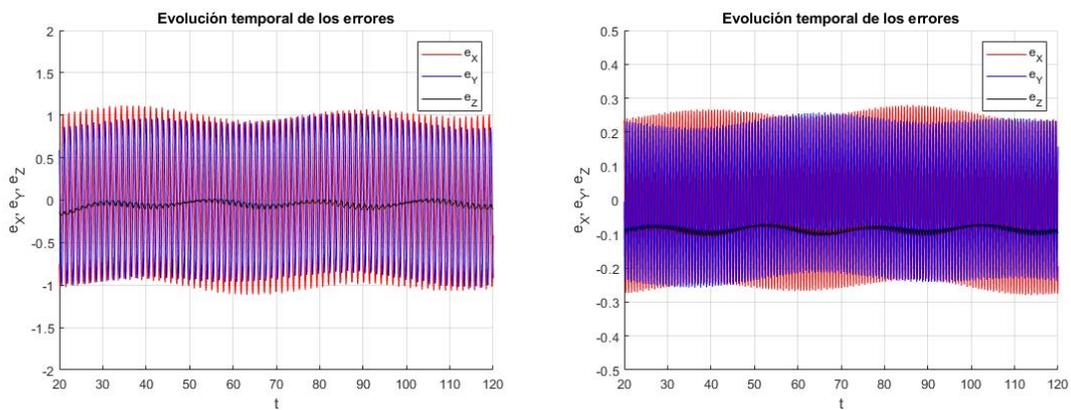


Figura 4.14 Errores para los casos con distintas frecuencias de reloj.

Se puede observar que al reducir al duplicar la frecuencia del reloj se pueden reducir los errores unas 10 veces, sin embargo, hay que tener en cuenta el efecto de este aumento de frecuencia en el equipo en el que se implemente, ya que aumenta la carga de trabajo del controlador y el consumo de batería, además de requerir un reloj más rápido.

Aunque no se ha mostrado gráficamente, otro de los efectos del aumento de la frecuencia del reloj es el aumento de la sobreoscilación inicial en las coordenadas X e Y.

4.2 Controlador "Completo"

4.2.1 Diseño del Controlador

El siguiente controlador se realiza a partir del modelo completo de un multirrotor en el cual se han tomado una serie de hipótesis:

- 1.-Las velocidades traslacionales no son muy altas.
- 2.-La resistencia aerodinámica es pequeña (por lo que se puede despreciar).
- 3.-Rotores de pequeño tamaño, por lo que la inercia de los mismos es mucho más pequeña que los momentos de inercia del multirrotor, pudiendo despreciar efectos giroscópicos.
- 4.-Las velocidades de rotación pueden aproximarse por las derivadas de los ángulos de Euler.

Una vez aplicadas estas hipótesis se llega al modelo (4.17).

$$\begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \\ \ddot{\phi} \\ \ddot{\theta} \\ \ddot{\psi} \end{bmatrix} = \begin{bmatrix} -(\cos \psi \sin \theta \cos \phi + \sin \psi \sin \phi) \frac{T_F}{m} \\ -(\sin \psi \sin \theta \cos \phi - \cos \psi \sin \phi) \frac{T_F}{m} \\ g - \cos \theta \cos \phi \frac{T_F}{m} \\ \frac{\tau_\phi}{I_x} \\ \frac{\tau_\theta}{I_y} \\ \frac{\tau_\psi}{I_z} \end{bmatrix} \quad (4.17)$$

donde (x,y,z) representan las coordenadas del centro de masas respecto al sistema de referencia inercial (donde el sentido positivo de OZ es la dirección de la gravedad), (ϕ,θ,ψ) son los ángulos de Euler y $(T_F, \tau_\phi, \tau_\theta, \tau_\psi)$ son las acciones de control que corresponden a la tracción total y los momentos de alabeo, cabeceo y guiñada respectivamente.

Para simplificar la notación se tomará:

$$\begin{bmatrix} \tau_\phi / I_x \\ \tau_\theta / I_y \\ \tau_\psi / I_z \end{bmatrix} = \begin{bmatrix} \bar{\tau}_\phi \\ \bar{\tau}_\theta \\ \bar{\tau}_\psi \end{bmatrix} \quad (4.18)$$

El objetivo del controlador será encontrar la combinación de acciones de control $(T_F, \bar{\tau}_\phi, \bar{\tau}_\theta, \bar{\tau}_\psi)$ que hace que el control alcance y siga la trayectoria de referencia $(x_{ref}, y_{ref}, z_{ref}, \phi_{ref})$.

El siguiente paso consiste en convertir el modelo (4.17) en un modelo que se pueda discretizar.

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \\ \dot{x}_5 \\ \dot{x}_6 \\ \dot{x}_7 \\ \dot{x}_8 \\ \dot{x}_9 \\ \dot{x}_{10} \\ \dot{x}_{11} \\ \dot{x}_{12} \end{bmatrix} = \begin{bmatrix} x_2 \\ -(\cos x_{11} \sin x_9 \cos x_7 + \sin x_{11} \sin x_7) \frac{T_F}{m} \\ x_4 \\ -(\sin x_{11} \sin x_9 \cos x_7 - \cos x_{11} \sin x_7) \frac{T_F}{m} \\ x_6 \\ g - \cos x_9 \cos x_7 \frac{T_F}{m} \\ x_8 \\ \bar{\tau}_\phi \\ x_{10} \\ \bar{\tau}_\theta \\ x_{11} \\ \bar{\tau}_\psi \end{bmatrix} \quad (4.19)$$

Tras esto, se discretiza el modelo del sistema (4.19).

$$\begin{bmatrix} -(\cos x_{11,n} \sin x_{9,n} \cos x_{7,n} + \sin x_{11,n} \sin x_{7,n}) \frac{T}{m} & 0 & 0 & 0 \\ -(\sin x_{11,n} \sin x_{9,n} \cos x_{7,n} - \cos x_{11,n} \sin x_{7,n}) \frac{T}{m} & 0 & 0 & 0 \\ -\cos x_{9,n} \cos x_{7,n} \frac{T}{m} & 0 & 0 & 0 \\ 0 & T & 0 & 0 \\ 0 & 0 & T & 0 \\ 0 & 0 & 0 & T \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} T_{F,n} \\ \bar{\tau}_{\phi,n} \\ \bar{\tau}_{\theta,n} \\ \bar{\tau}_{\psi,n} \end{bmatrix} = \begin{bmatrix} x_{2,n+1} - x_{2,n} \\ x_{4,n+1} - x_{4,n} \\ x_{6,n+1} - x_{6,n} - gT \\ x_{8,n+1} - x_{8,n} \\ x_{10,n+1} - x_{10,n} \\ x_{12,n+1} - x_{12,n} \\ x_{1,n+1} - x_{1,n} \\ x_{3,n+1} - x_{3,n} \\ x_{5,n+1} - x_{5,n} \\ x_{7,n+1} - x_{7,n} \\ x_{9,n+1} - x_{9,n} \\ x_{11,n+1} - x_{11,n} \end{bmatrix} \quad (4.20)$$

Se asume un acercamiento a la trayectoria suave y proporcional al error, de esta forma, para que el sistema te una solución exacta se tiene:

$$x_{2,ref,n} = \frac{\Delta x_{1,n}}{T} \quad (4.21)$$

$$x_{4,ref,n} = \frac{\Delta x_{3,n}}{T} \quad (4.22)$$

$$x_{6,ref,n} = \frac{\Delta x_{5,n}}{T} \quad (4.23)$$

$$x_{12,ref,n} = \frac{\Delta x_{11,n}}{T} \quad (4.24)$$

donde

$$\Delta x_{i,n} = x_{i,ref,n+1} - k_i(x_{i,ref,n} - x_{i,n}) - x_{i,n} \quad (4.25)$$

De forma análoga, se tienen los valores de referencia para x_8 y x_{10} .

$$x_{8,ref,n} = \frac{\Delta x_{7,n}}{T} \quad (4.26)$$

$$x_{10,ref,n} = \frac{\Delta x_{9,n}}{T} \quad (4.27)$$

Analizando las tres primeras filas del sistema (4.20), se puede comprobar que existen tres ecuaciones que contienen la tracción total:

$$\begin{bmatrix} -(\cos x_{11,n} \sin x_{9,n} \cos x_{7,n} + \sin x_{11,n} \sin x_{7,n}) \frac{T}{m} \\ -(\sin x_{11,n} \sin x_{9,n} \cos x_{7,n} - \cos x_{11,n} \sin x_{7,n}) \frac{T}{m} \\ -\cos x_{9,n} \cos x_{7,n} \frac{T}{m} \end{bmatrix} T_F = \frac{m}{T} \begin{bmatrix} \Delta x_{2,n} \\ \Delta x_{4,n} \\ \Delta x_{6,n} - gT \end{bmatrix} \quad (4.28)$$

Para que el sistema tenga solución exacta T_F tiene que tener un valor único en (4.28), por tanto, se deben cumplir las condiciones (4.29) y (4.30).

$$\tan x_{7,n} = \frac{\cos x_{9,ref,n}}{\sin x_{11,n}} \left(\frac{\Delta x_{2,n}}{\Delta x_{6,n} - gT} - \cos x_{11,n} \tan x_{9,ref,n} \right) \quad (4.29)$$

$$\frac{\Delta x_{4,n} \tan x_{11,n} + \Delta x_{2,n}}{(\Delta x_{6,n} - gT)(\tan x_{11,n} \sin x_{11,n} + \cos x_{11,n})} \quad (4.30)$$

Por último, se aplica la resolución mediante mínimos cuadrados a (4.20) para hallar las acciones de control necesarias.

$$\begin{bmatrix} T_{F,n} \\ \tau_{\phi,n} \\ \tau_{\theta,n} \\ \tau_{\psi,n} \end{bmatrix} = \frac{1}{T} \begin{bmatrix} -m \times \left[\begin{array}{l} (\cos x_{11,n} \sin x_{9,n} \cos x_{7,n} + \sin x_{11,n} \sin x_{7,n}) \Delta x_{2,n} + \dots \\ \dots + (\sin x_{11,n} \sin x_{9,n} \cos x_{7,n} - \cos x_{11,n} \sin x_{7,n}) \Delta x_{4,n} + \dots \\ \dots + \cos x_{9,n} \cos x_{7,n} (\Delta x_{6,n} - gT) \end{array} \right] \\ \Delta x_{8,n} \\ \Delta x_{10,n} \\ \Delta x_{12,n} \end{bmatrix} \quad (4.31)$$

4.2.2 Resultados de Simulación

Una vez diseñado, se estudia el comportamiento del controlador con distintas trayectorias de referencia mediante simulaciones realizadas en Simulink.

Los parámetros escogidos para el controlador a simular son los siguientes:

$$\begin{array}{cccc} k_1 = 0.8 & k_2 = 0.8 & k_3 = 0.9 & k_4 = 0.95 \\ k_5 = 0.9 & k_6 = 0.95 & k_7 = 0.8 & k_8 = 0.85 \\ k_9 = 0.7 & k_{10} = 0.67 & k_{11} = 0.7 & k_{12} = 0.67 \end{array}$$

En este modelo, se puede apreciar que la dinámica de x_{11} si que se encuentra acoplada al resto del modelo, por lo que este afectará.

A excepción de los casos en los que se especifique, se mantendrá el valor de $x_{11} = \pi/2$ constante.

En este caso se omitirá la simulación de las rectas, ya que si las trayectorias más complejas se realizan de forma correcta, se deberán realizar correctamente las más simples.

Lemniscata

Se simula una trayectoria con forma de lemniscata en el plano horizontal, al igual que en el caso anterior, con el objetivo de realizar comprobaciones de funcionamiento frente a referencia medianamente compleja y su efecto en la altitud.

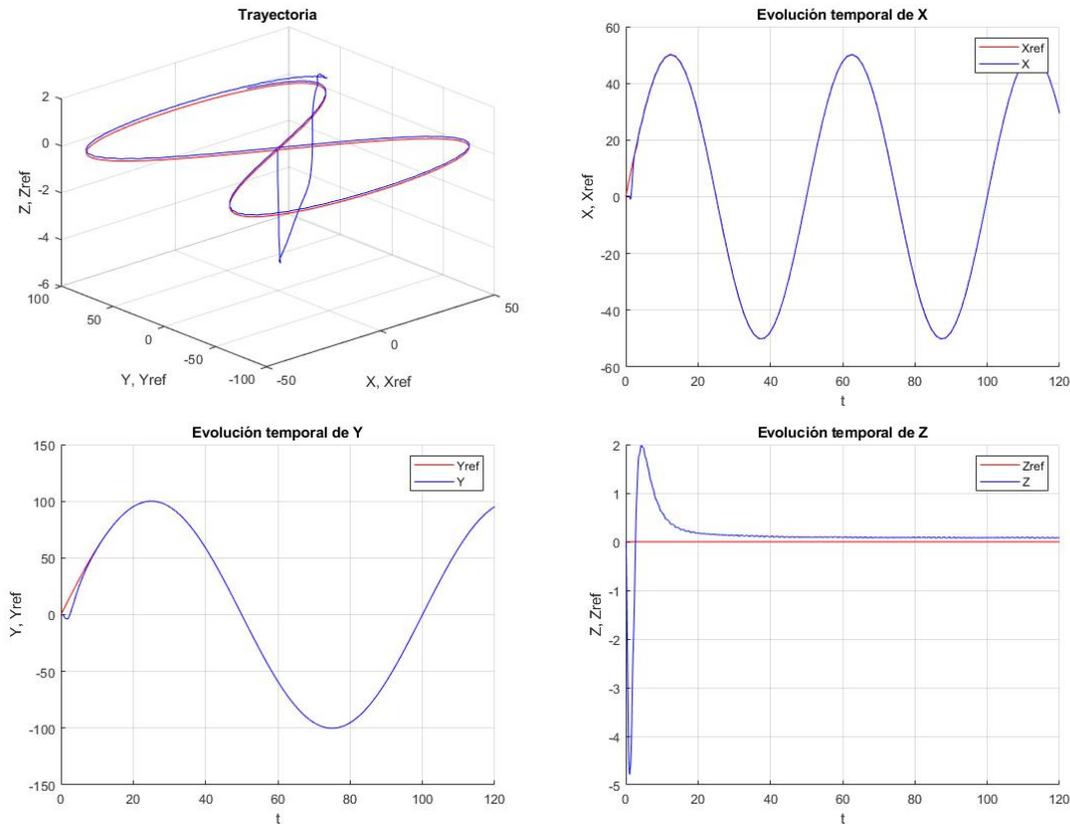


Figura 4.15 Trayectoria recorrida por el multirrotor y evolución de las variables de posición durante la misma.

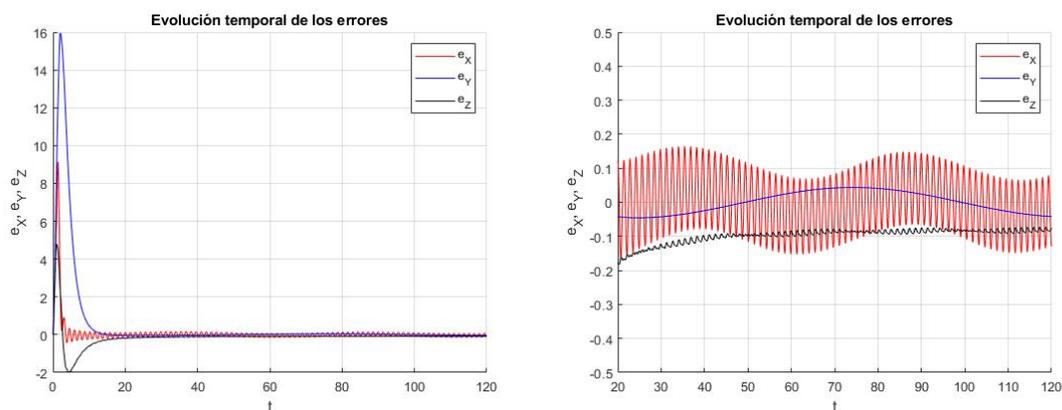


Figura 4.16 Variación de los errores durante la trayectoria.

Se puede apreciar que inicialmente hay un pequeño error en x e y, sin embargo, esto se corrige rápidamente y, tras ello, el error en estas coordenadas es muy pequeño.

En cuanto a la altitud se aprecia una caída inicial, seguida de una sobreoscilación para llegar a la altitud deseada, aunque se puede apreciar un error permanente de ~ -0.1 m. La caída es debida en parte al hecho de

que al comienzo de la simulación se tienen los rotores apagados, por lo que cae hasta que estos estabilizan el sistema, esto no debe ocurrir en la realidad ya que si el multirrotor se encuentra en el aire debe tener los rotores en funcionamiento. En parte, esta caída también es producida por el controlador al cambiar de referencia.

Ascenso Helicoidal

A continuación, se estudia el comportamiento frente a una referencia de ascenso helicoidal, para comprobar el comportamiento frente al desplazamiento coordinado en los 3 ejes y el efecto de gran error en la posición inicial.

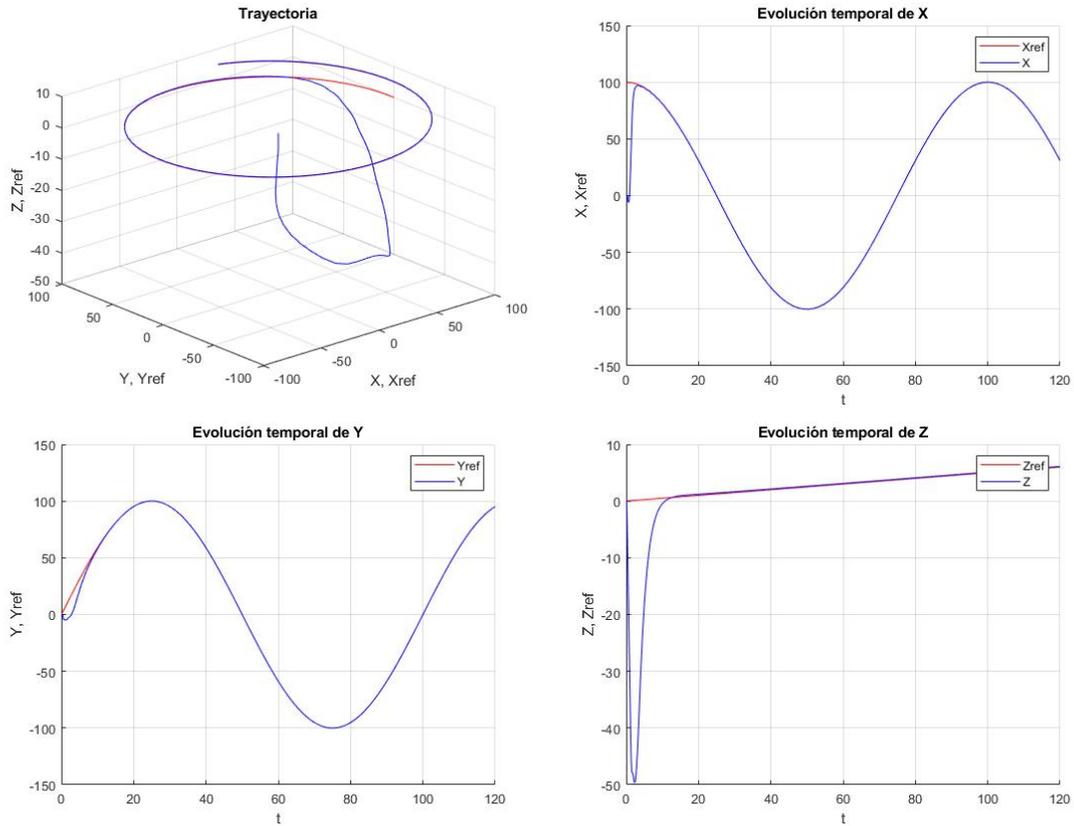


Figura 4.17 Trayectoria recorrida por el multirrotor y evolución de las variables de posición durante la misma.

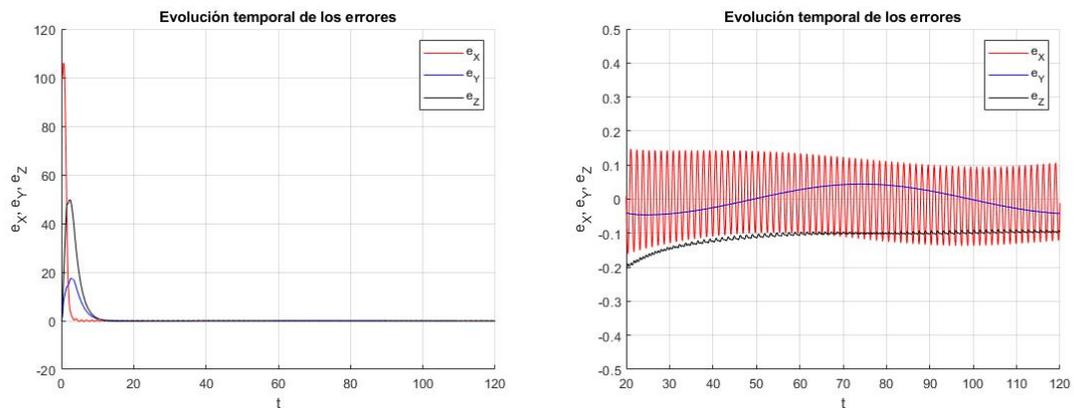


Figura 4.18 Variación de los errores durante la trayectoria.

En la trayectoria se aprecia una bastante rápida transferencia desde el punto inicial hasta la referencia, sin embargo, esta transferencia provoca una gran caída del multirrotor en altitud (unos 50 m).

Una vez alcanzada la trayectoria deseada se sigue con gran fidelidad existiendo unos errores muy pequeños en régimen permanente.

Para solucionar este gran descenso en la altitud se propone el indicar la transferencia dentro de la referencia, esto se hará mediante una recta que conecta el inicio con el inicio del ascenso.

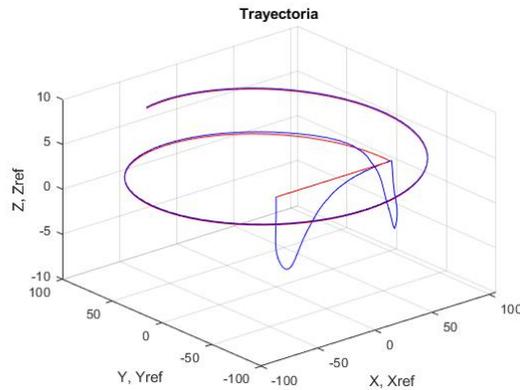


Figura 4.19 Trayectoria recorrida por el multirrotor.

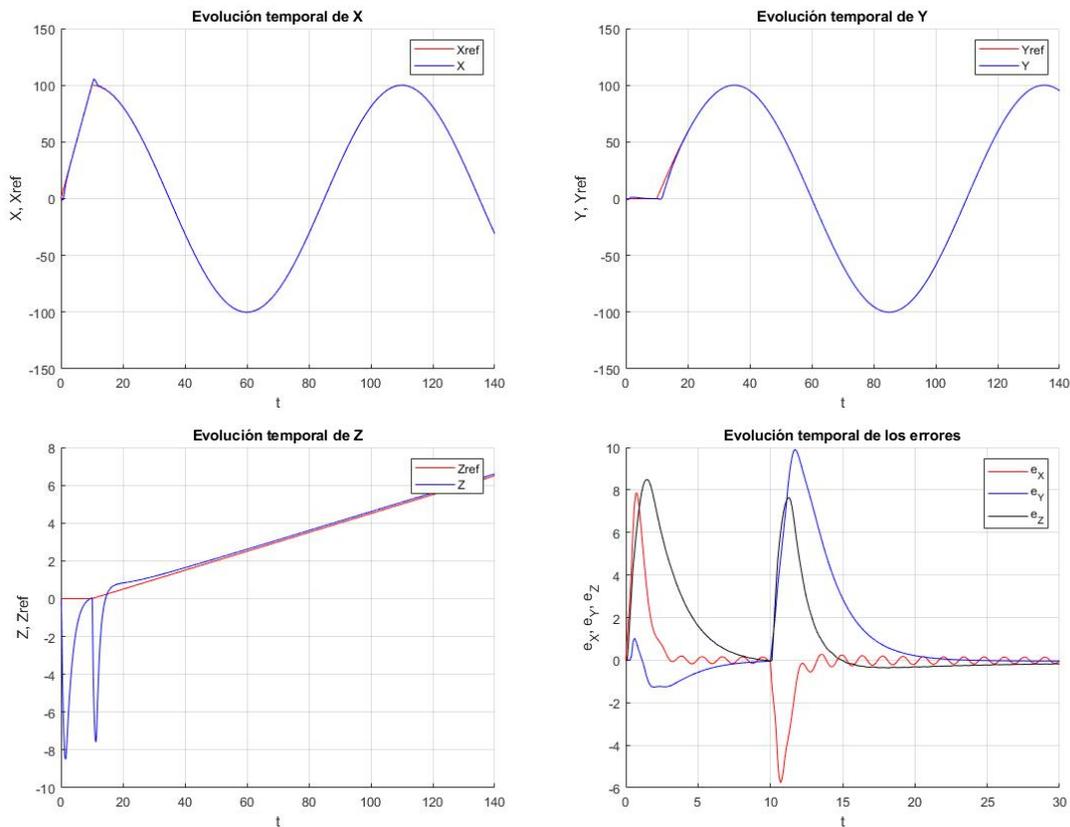


Figura 4.20 Evolución de las variables de posición y el error durante la trayectoria.

Se puede comprobar que la adición de la trayectoria de transferencia a la referencia se acota en gran medida el error en todas las coordenadas de posición, especialmente en la altitud, donde la caída queda acotada dentro de la decena de metros, reduciendo el riesgo de colisión.

Efecto del Giro

Finalmente, se va a concluir la sección de simulaciones comprobando el efecto de una referencia de x_{11} que provoque su aumento de forma lineal (es decir, un giro continuo a velocidad constante).

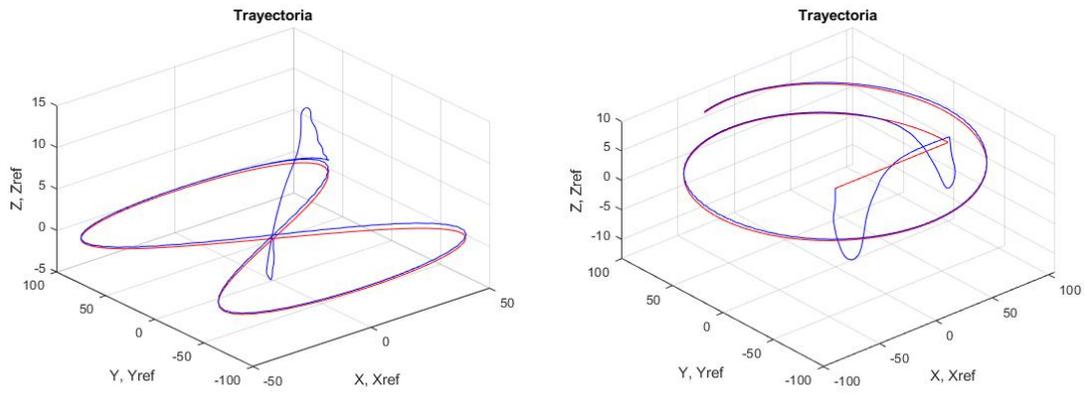


Figura 4.21 Trayectorias recorridas por el multirrotor.

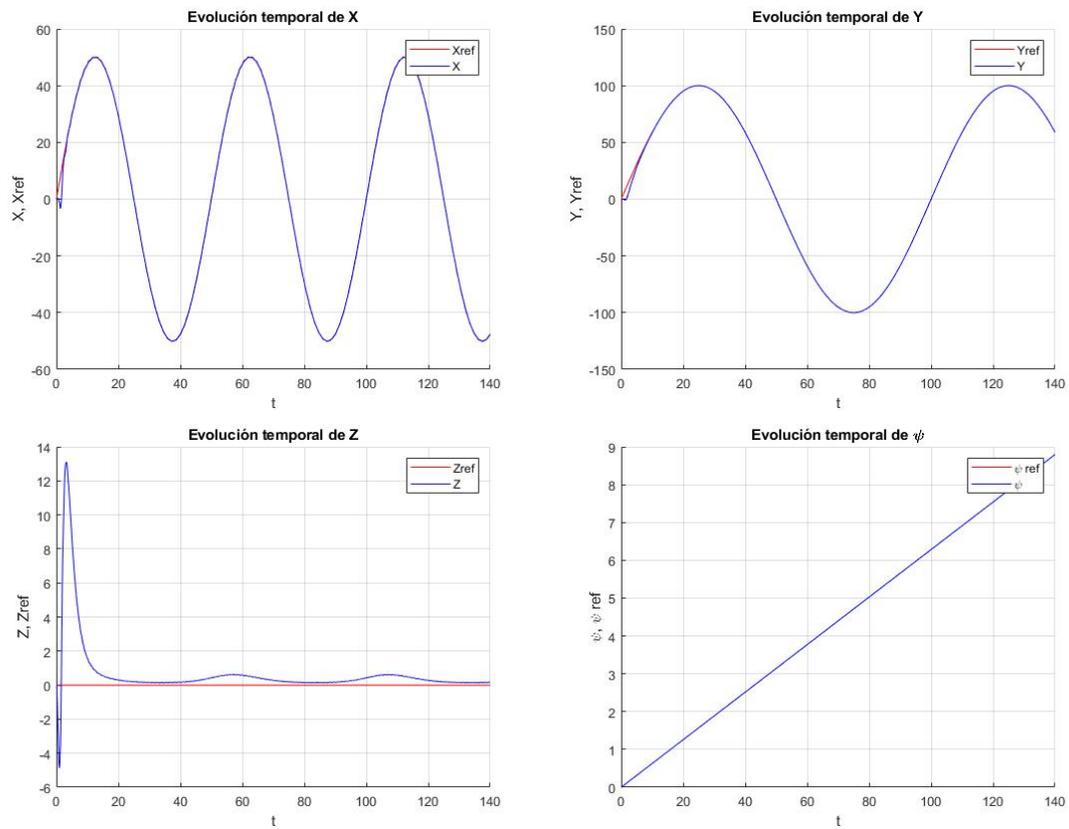


Figura 4.22 Evolución de las variables de posición y del giro en la trayectoria de lemniscata.

Observando los resultados obtenidos en las simulaciones se puede comprobar que el coordinar el giro con los otros desplazamientos provoca aumento en los errores de seguimiento del controlador, sin embargo, los aumentos en los errores son relativamente pequeños, por lo que se puede coordinar el error sin problema para orientar el multirrotor en caso de tener un equipo con movilidad reducida.

Puede apreciarse que el error máximo se tiene en los instantes de tiempo en los que el giro toma como valor un múltiplo de π y este se da para la altitud, aunque a pesar de esto, el error es pequeño, de unas decenas de centímetros.

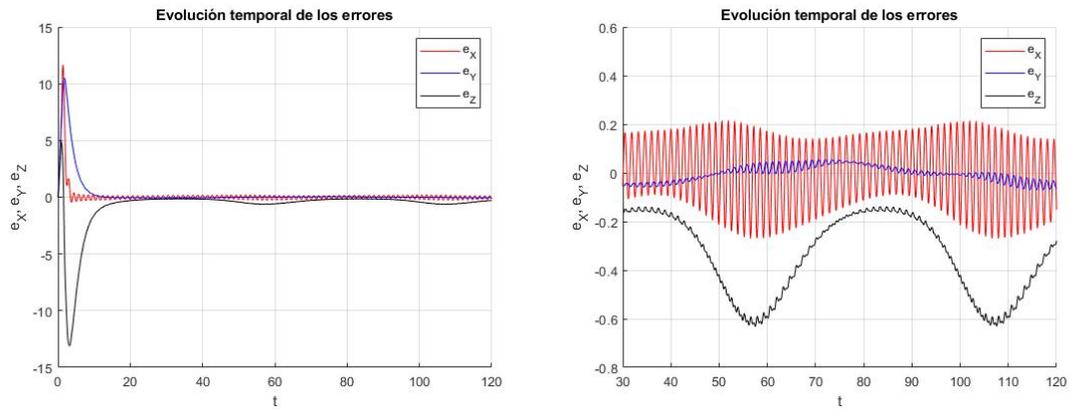


Figura 4.23 Errores en la trayectoria de la lemniscata.

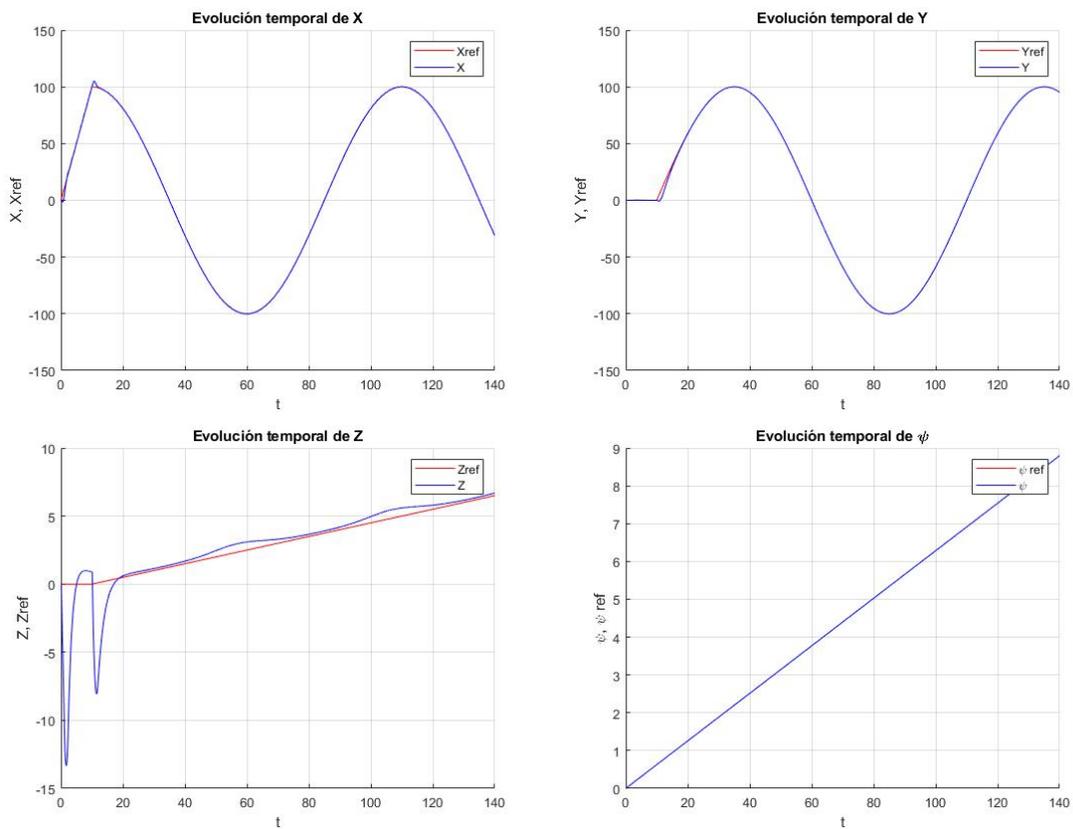


Figura 4.24 Evolución de las variables de posición y del giro en la trayectoria helicoidal.

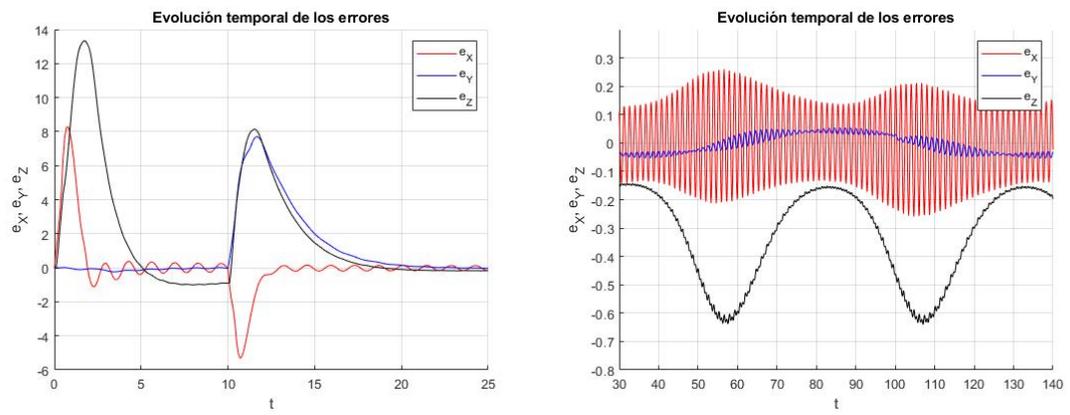


Figura 4.25 Errores en la trayectoria helicoidal.

5 Conclusiones Generales

A lo largo del documento presentado se ha presentado la metodología LAB CD y ejemplos de la implementación de la misma en un robot móvil y un multirrotor mediante simulaciones.

En vista de los resultados obtenidos se puede comentar que esta metodología cumple con su objetivo general, poder solucionar el problema del seguimiento de trayectoria mediante un método aplicable de forma lo más general posible y con un método no tan complejo como muchos de los utilizados de forma más común.

Adicionalmente, como se ha comprobado, el método permite que estos controladores diseñados proporcionen unos errores pequeños que permiten su utilización en un gran número de aplicaciones que no requieran precisiones descomunales. Ejemplos de posibles implementaciones son un robot móvil de transporte de piezas o un multirrotor que tome imágenes de un lugar elevado para organizar el mantenimiento.

Es cierto que este método presenta algunos inconvenientes, principalmente derivados de la necesidad de tener una trayectoria de referencia lo suficientemente suave como para no incurrir en grandes errores, sin embargo, estos pueden ser paliados si se tiene en cuenta a la hora de diseñar el programa encargado de generar la misma.

Como conclusión final se puede decir que el método LAB CD es un buen método para el diseño de controladores de sistemas que no requieran el seguimiento de una trayectoria sin necesidad de precisiones astronómicas y un buen punto de partida para el estudio del comportamiento de un sistema o modelo frente al control.

A Diagramas Simulink

En esta sección se recopilan los modelos de Simulink que se han utilizado para implementar los distintos controladores desarrollados a lo largo del documento en tamaño lo suficientemente grande como para poder visualizarse de forma correcta.

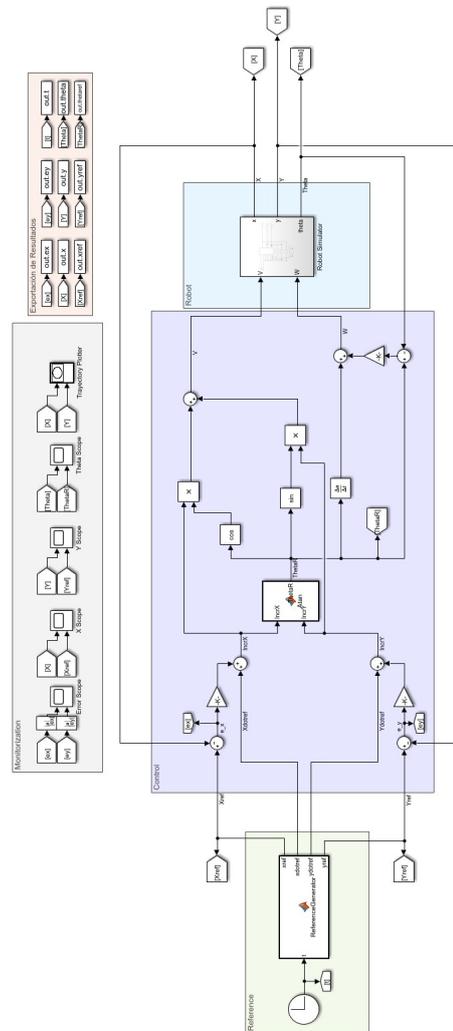


Figura A.1 Controlador en tiempo continuo para el robot móvil.

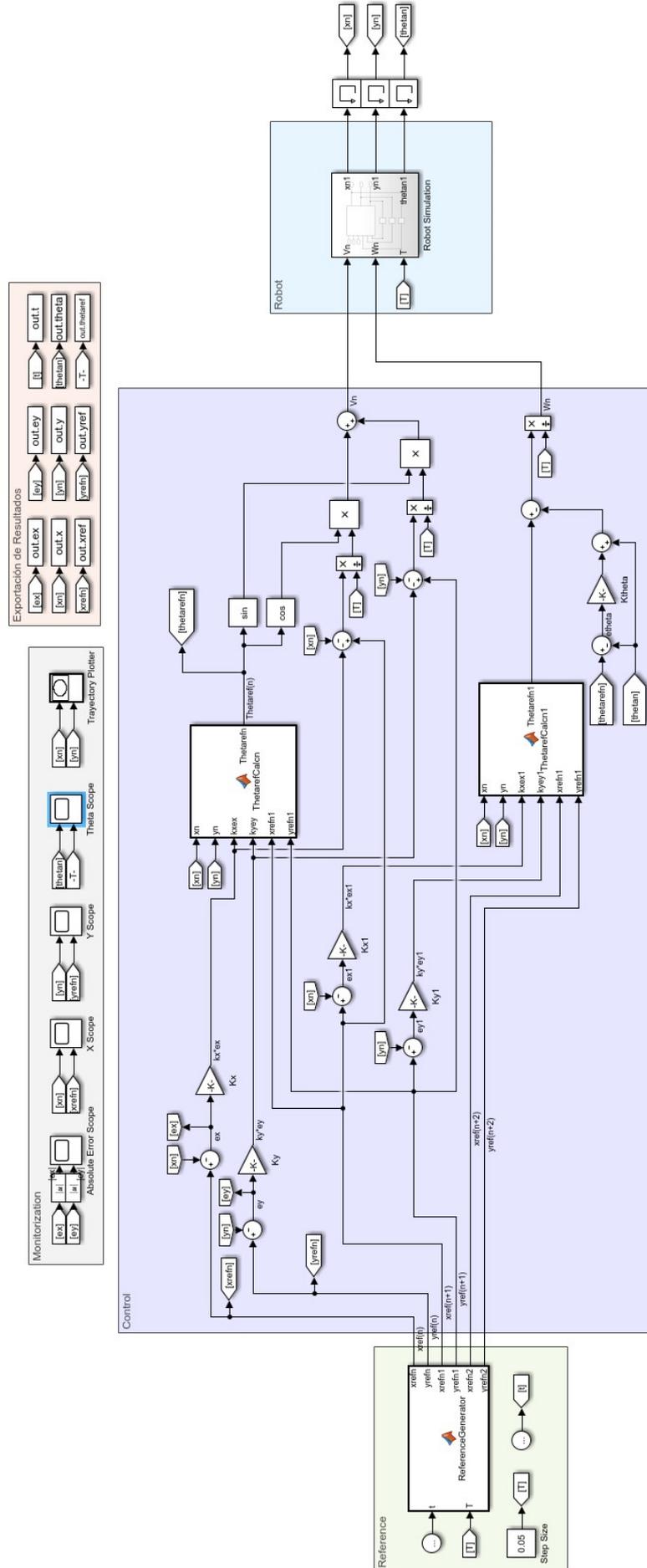


Figura A.2 Controlador en tiempo discreto con aproximación de Euler en el modelo cinemático para el robot móvil.

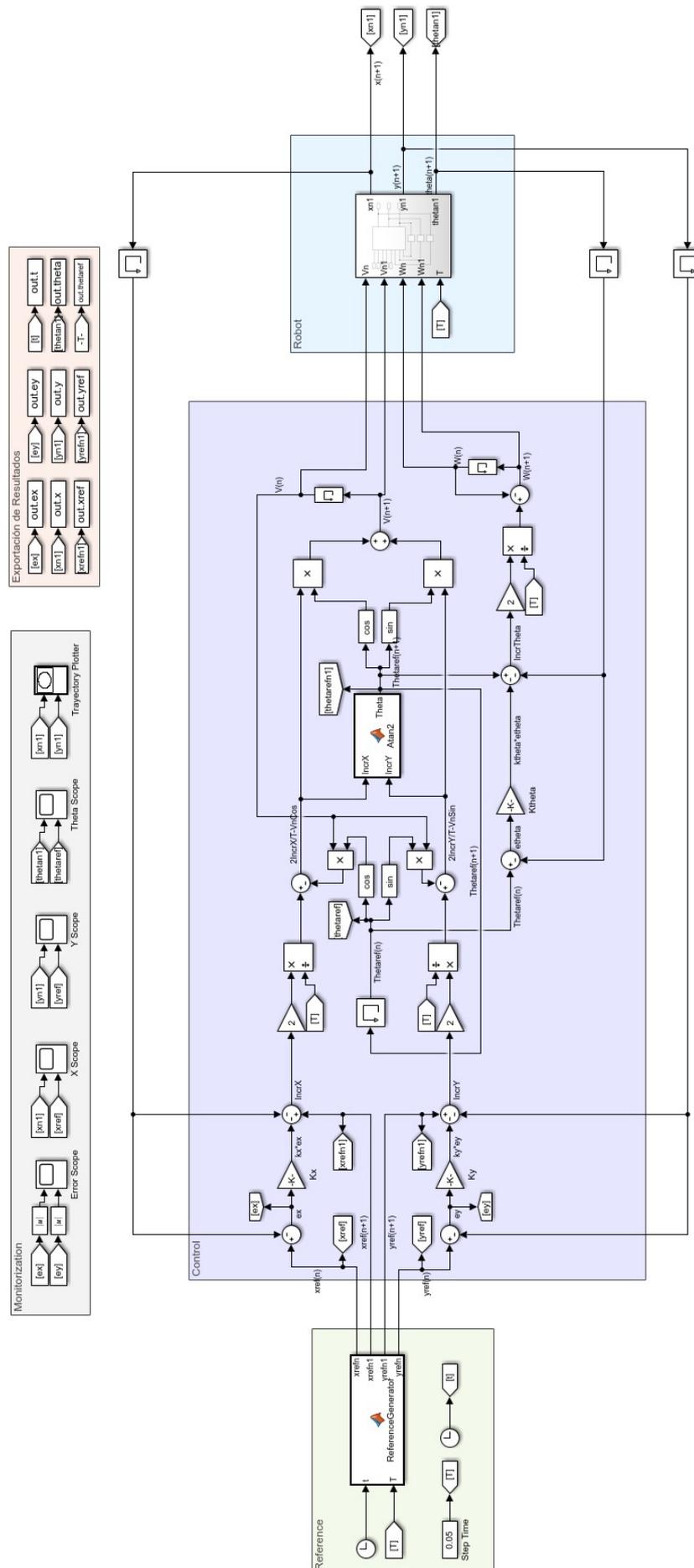


Figura A.3 Controlador en tiempo discreto con aproximación por trapezios en el modelo cinemático para el robot móvil.

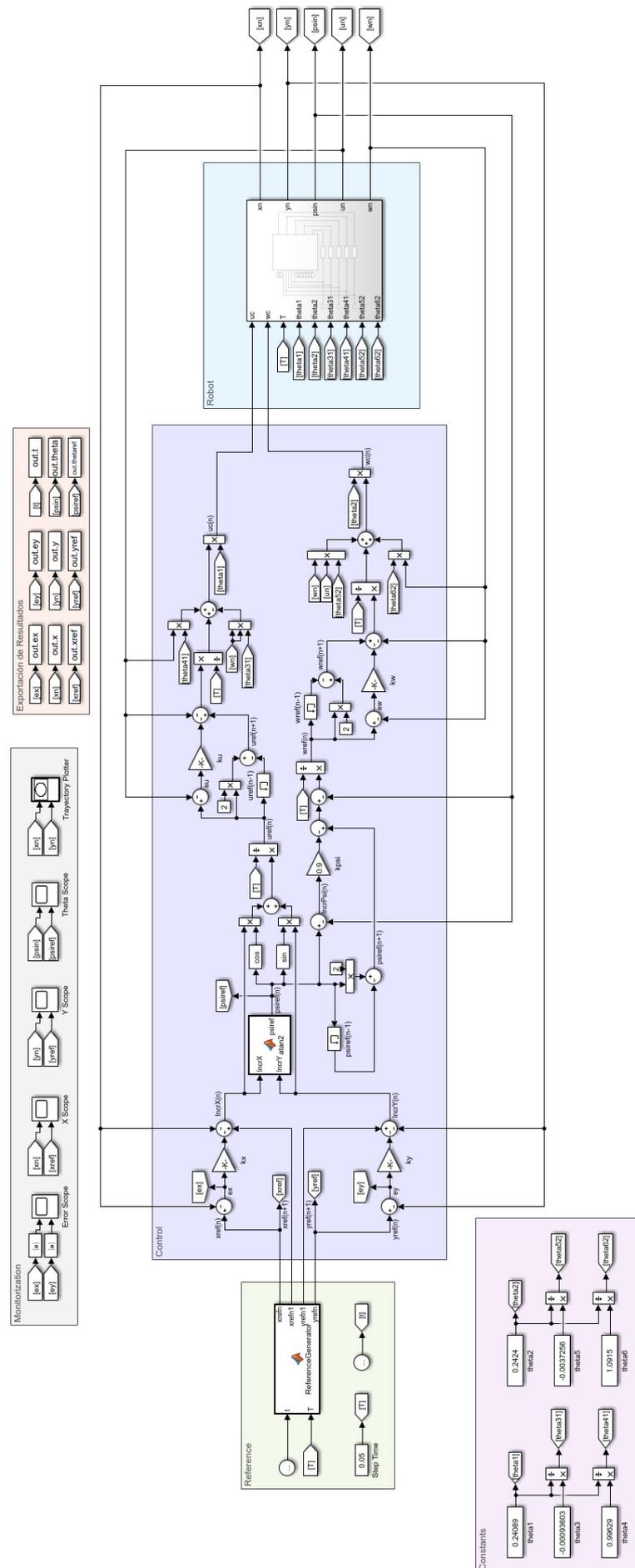


Figura A.4 Controlador en tiempo discreto basado en el modelo dinámico del robot móvil.

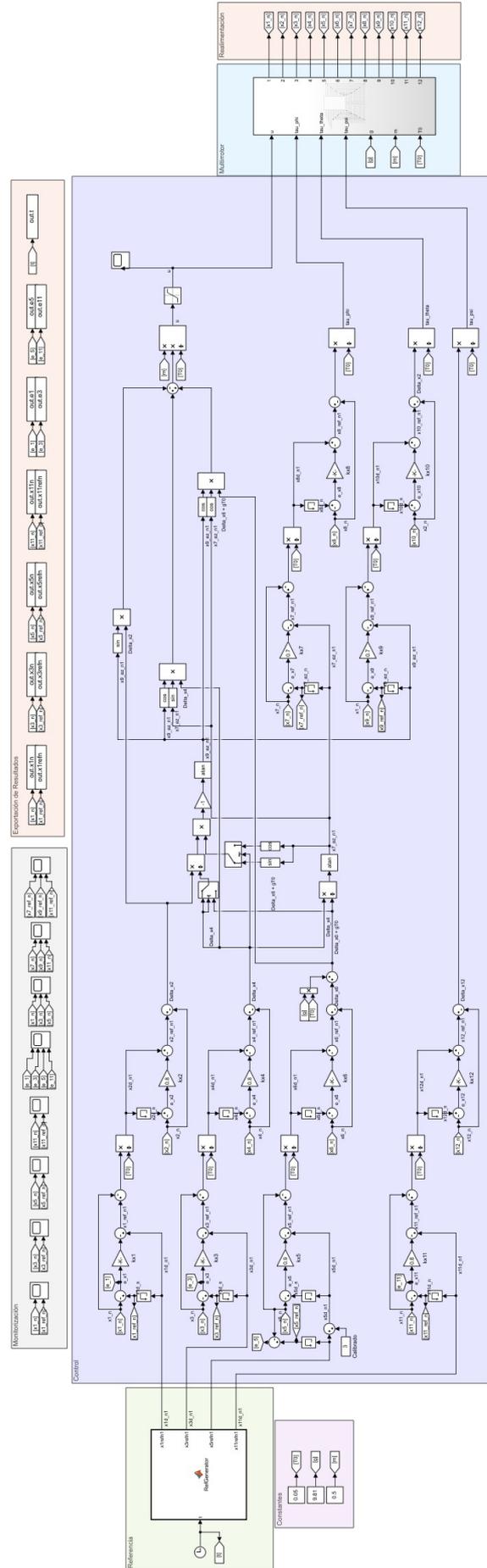


Figura A.5 Controlador en tiempo discreto basado en el modelo simplificado del multirotor.

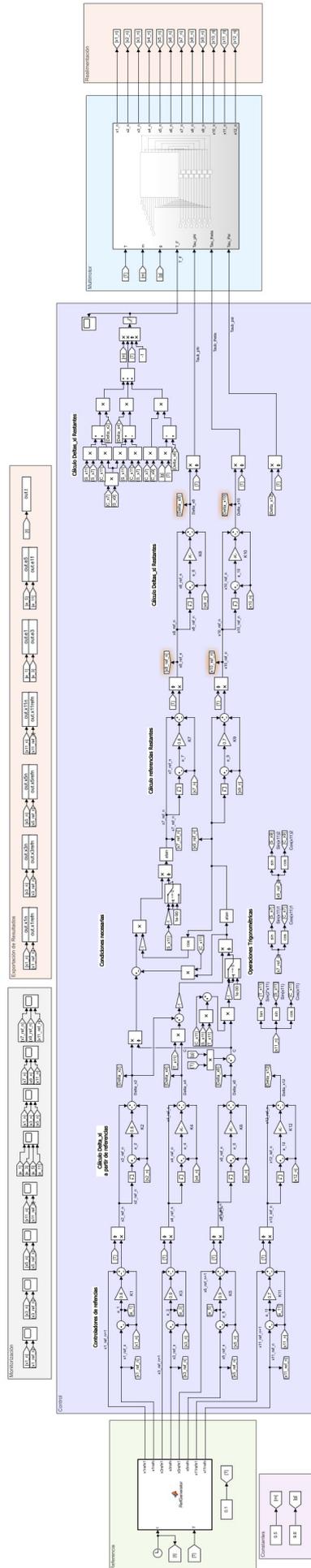


Figura A.6 Controlador en tiempo discreto basado en el modelo "completo" del multirrotor.

B Bibliografía

En esta sección se recopilan las referencias utilizadas para la redacción del documento.

Scaglia, G. J. E., Serrano, M. E., Albertos, P. 2020. Linear algebra based trajectory control. *Revista Iberoamericana de Automática e Informática Industrial* 17, 344-353. <https://doi.org/10.4995/riai.2020.13584>

Scaglia, G. J. E., Serrano, M. E., Albertos, P. 2020. *Linear Algebra Based Controllers. Design and Applications*. Editorial Springer.

Ollero, Anibal. 2020. *Apuntes de la asignatura Vehículos Aéreos no Tripulados del Grado en Ingeniería Aeroespacial*. Universidad de Sevilla.

Índice de Figuras

3.1	Representación Gráfica del Robot Móvil objetivo del estudio	11
3.2	Trayectoria realizada por el robot móvil	13
3.3	Evolución temporal de las variables de posición	14
3.4	Evolución temporal de la orientación y los errores	14
3.5	Trayectoria realizada por el robot móvil	14
3.6	Evolución temporal de las variables de posición	15
3.7	Evolución temporal de la orientación y los errores	15
3.8	Trayectoria realizada por el robot móvil	15
3.9	Evolución temporal de las variables de posición	16
3.10	Evolución temporal de la orientación y los errores	16
3.11	Trayectoria realizada por el robot móvil	16
3.12	Evolución temporal de las variables de posición	17
3.13	Evolución temporal de la orientación y los errores	17
3.14	Trayectoria realizada por el robot móvil	18
3.15	Evolución temporal de las variables de posición	19
3.16	Evolución temporal de la orientación y los errores	19
3.17	Trayectoria realizada por el robot móvil	19
3.18	Evolución temporal de las variables de posición	20
3.19	Evolución temporal de la orientación y los errores	20
3.20	Trayectoria realizada por el robot móvil	21
3.21	Evolución temporal de las variables de posición	22
3.22	Evolución temporal de la orientación y los errores	22
3.23	Trayectoria realizada por el robot móvil	22
3.24	Evolución temporal de las variables de posición	23
3.25	Evolución temporal de la orientación y los errores	23
3.26	Robot móvil PIONNER 3DX, modelo y equipamiento de laboratorio	24
3.27	Trayectoria realizada por el robot móvil	25
3.28	Evolución temporal de las variables de posición	26
3.29	Evolución temporal de la orientación y los errores	26
3.30	Trayectoria realizada por el robot móvil	26
3.31	Evolución temporal de las variables de posición	27
3.32	Evolución temporal de la orientación y los errores	27
3.33	Comparativa de errores para la circunferencia durante la trayectoria de acercamiento	27
3.34	Comparativa de errores para la circunferencia en el permanente	28
3.35	Comparativa de errores para la lemniscata	28
4.1	Ejemplo de multirrotor (DJI Matrice 600 Pro) y de los grados de libertad de un multirrotor (Montgomery, 2014)	31
4.2	Trayectoria recorrida por el multirrotor	34
4.3	Evolución de las variables de posición y del error	35
4.4	Trayectoria recorrida por el multirrotor	35

4.5	Evolución de las variables de posición y del error	36
4.6	Trayectoria recorrida por el multirotor	36
4.7	Evolución de las variables de posición y del error	37
4.8	Trayectoria recorrida por el multirotor	37
4.9	Evolución de las variables de posición y del error	38
4.10	Trayectoria recorrida por el multirotor	38
4.11	Evolución de las variables de posición y del error	39
4.12	Trayectoria recorrida por el multirotor tras calibrar para la lemniscata y el ascenso helicoidal	39
4.13	Evolución de la altitud tras calibrar para la lemniscata y el ascenso helicoidal	40
4.14	Errores para los casos con distintas frecuencias de reloj	40
4.15	Trayectoria recorrida por el multirotor y evolución de las variables de posición durante la misma	43
4.16	Variación de los errores durante la trayectoria	43
4.17	Trayectoria recorrida por el multirotor y evolución de las variables de posición durante la misma	44
4.18	Variación de los errores durante la trayectoria	44
4.19	Trayectoria recorrida por el multirotor	45
4.20	Evolución de las variables de posición y el error durante la trayectoria	45
4.21	Trayectorias recorridas por el multirotor	46
4.22	Evolución de las variables de posición y del giro en la trayectoria de lemniscata	46
4.23	Errores en la trayectoria de la lemniscata	47
4.24	Evolución de las variables de posición y del giro en la trayectoria helicoidal	47
4.25	Errores en la trayectoria helicoidal	48
A.1	Controlador en tiempo continuo para el robot móvil	51
A.2	Controlador en tiempo discreto con aproximación de Euler en el modelo cinemático para el robot móvil	52
A.3	Controlador en tiempo discreto con aproximación por trapecios en el modelo cinemático para el robot móvil	53
A.4	Controlador en tiempo discreto basado en el modelo dinámico del robot móvil	54
A.5	Controlador en tiempo discreto basado en el modelo simplificado del multirotor	55
A.6	Controlador en tiempo discreto basado en el modelo "completo" del multirotor	56