



UNIVERSIDAD DE CONCEPCIÓN
DIRECCIÓN DE POSTGRADO

Facultad de Ingeniería - Programa de Magíster en Ciencias de la Ingeniería con mención en
Ingeniería Mecánica

CONTROL PREDICTIVO BASADO EN MODELO APLICADO A DIRIGIBLE PARA INTERIORES

Tesis para optar al grado de
Magíster en Ciencias de la Ingeniería con mención en Ingeniería Mecánica

Cristian Basoalto Contreras
CONCEPCIÓN - CHILE
2023

Profesores Guías: Cristián Vicuña M.
Dpto. Ingeniería Mecánica, Facultad de Ingeniería
Universidad de Concepción

Agradecimientos

En primer lugar agradezco a mi familia por acompañarme en este proceso y por contar siempre con su apoyo.

Además quiero agradecer a los académicos del DIM-UdeC que de alguna manera han contribuido a mi formación y crecimiento profesional y personal. En particular, agradezco a los profesores Cristián Vicuña, Cristian Cuevas y Frak Tinnap, por su apoyo y correcciones durante el desarrollo de esta investigación. También agradezco al profesor Daniel Sbarbaro por proporcionarme el tema de investigación y la información necesaria para iniciar este trabajo.

Agradezco a la dirección del departamento del DIMEC-UBB, y a mis colegas a quienes tengo en gran estima y afecto. En particular a don Reinaldo Sánchez Arriagada por su confianza y el apoyo brindado en este proceso, así como también en los cursos que propuse mientras él fue director del departamento.

Resumen

En las últimas décadas se han desarrollado diversos estudios en dirigibles, ya sea para vigilancia, transporte de carga, e incluso para mostrar obras de arte en museos. Por esto se genera el interés en explorar sistemas de control de movimiento optimizados, para generar respuestas más rápidas y confiables. En comparación con las investigaciones hasta ahora desarrolladas, en este trabajo se espera tener todos los sensores de medición a bordo del dirigible, con el fin de tener una mayor movilidad.

Esta investigación se centra en el diseño de un dirigible de navegación autónoma controlado por un Control Predictivo Basado en Modelo (MPC) para controlar la altura y la orientación. El proceso comienza con la identificación del sistema mediante un modelo de caja gris, donde se determina el modelo dinámico para el desplazamiento vertical, horizontal y de rotación (guiñada). A las mediciones de los estados se les agrega un observador mediante un Filtro de Kalman para reducir el ruido, determinando principalmente la velocidad del dirigible para el control de altura y orientación. Posteriormente, se aplica un control MPC considerando restricciones en las entradas, con el objetivo de controlar el efecto de una posible saturación en los motores. Finalmente, se evalúa el desempeño del MPC comparándolo con un sistema de control PID, que es el método de control comúnmente aplicado a estos dispositivos.

En el control de altura usando el MPC, se alcanzó un sobrepaso máximo del 11%, junto con un tiempo de levantamiento de 1.9 segundos y un tiempo de asentamiento de 10.8 segundos. En cuanto al control de orientación, los resultados mostraron un tiempo de levantamiento de 1.7 segundos, un tiempo de asentamiento de 2.8 segundos y un sobrepaso máximo del 0%. Al comparar el desempeño del MPC con el PID, la simulación con el PID mostró tiempos de asentamiento y levantamiento mejores a los del MPC. Es importante señalar que esta diferencia se debe a que la simulación del PID no tomó en cuenta las restricciones propias de los motores, resultando en una mayor fuerza aplicada en el control de altura y un mayor momento para el control de orientación.

Índice de contenidos

Página

CAPÍTULO 1

Introducción	6
1.1 Hipótesis	7
1.2 Objetivos	7
1.2.1 Objetivo general	7
1.2.2 Objetivos específicos	7
1.3 Metodología	8

CAPÍTULO 2

Marco teórico	9
2.1 Ecuaciones de movimiento	9
2.2 Identificación de sistemas	11
2.3 Control PID	11
2.4 Control Predictivo Basado en Modelo	13
2.4.1 Función de costo	15
2.4.2 Restricciones	16
2.5 Filtro de Kalman	18
2.6 Localización mediante marcadores	21

CAPÍTULO 3

Componentes del dirigible	24
3.1 Globo	24
3.2 Transmisión y recepción de datos	24
3.3 Tarjeta programable	26
3.4 Sensores y actuadores	26
3.4.1 Medición de altura	26
3.4.2 Medición de orientación	27
3.4.3 Cámara utilizada para localización	28
3.4.4 Motores y Puente H	28
3.5 Góndola	29
3.5.1 Masa total de componentes	30

CAPÍTULO 4

Evaluaciones experimentales	32
4.0.1 Fuerza de empuje de motores	32
4.0.2 Identificación de parámetros	34
4.1 Aplicación de MPC	38
4.1.1 Parámetros del filtro de Kalman	38
4.1.2 Aplicación de MPC para control de altura	39
4.1.3 Aplicación de MPC para control de Orientación	43

CAPÍTULO 5	
Conclusiones y perspectivas a futuro	48
CAPÍTULO 6	
Anexos	52
6.1 Diagramas de simulaciones en Simulink	52
6.1.1 Matriz de calibración de cámara	53

Índice de figuras

2.1	Esquema posicionamiento y orientación del dirigible.	9
2.2	Esquema de orientación del dirigible respecto al marco de referencia η	10
2.3	Diagrama de bloques de control PID.	12
2.4	Esquema de predicciones del MPC	15
2.5	Diagrama de bloques de Control Predictivo Basado en Modelo	18
2.6	Diagrama de efecto de covarianzas Q y V en filtro de Kalman.	19
2.7	Diagrama de bloques de MPC con filtro de Kalman.	20
2.8	Esquema propuesto para localización mediante marcadores.	22
2.9	Representación de parámetros intrínsecos.	22
2.10	Transformación de ejes coordenados mediante transformadas homogéneas.	23
3.1	Estructura de comunicación serial UART.	25
3.2	Arduino Fio.	26
3.3	Medición de altura.	27
3.4	Sensor MPU 6050 con sus 3 ejes de medición.	27
3.5	Cámara Raspberry Pi V2.1.	28
3.6	Puente H TB6612FNG.	29
3.7	Góndola diseñada para el dirigible a) Prototipo inicial para Arduino Fio, b) prototipo mejorado para Raspberry Pi 4.	29
3.8	Dirigible con todos sus componentes.	30
4.1	Balanza para medición de fuerza generada por motores DC para cada voltaje aplicado.	32
4.2	Esquema de prueba para medición de fuerza de motores en eje z	33
4.3	Fuerza de empuje vs voltaje aplicado a motores.	33
4.4	Experimento para determinar constantes de amortiguamiento y masa añadida. a) ensayo para constantes m_z y D_z b) ensayo para constantes m_{xy} y D_{xy}	35
4.5	Esquema de experimento para determinar inercia respecto I_z y amortiguamiento J del dirigible.	36
4.6	a) Validación de modelo para desplazamiento en eje z . b) Validación de modelo para desplazamiento horizontal en ejes xy . c) Validación de modelo para rotación respecto al eje z	37
4.7	Control de altura z mediante control MPC sin restricciones.	40
4.8	Comparación entre control predictivo basado en modelo y control PID, en control de altura z	41
4.9	Aplicación de MPC en control de altura, sometiendo el dirigible a perturbaciones.	42
4.10	Aplicación de MPC para control de altura, con restricciones en entrada.	43
4.11	Aplicación de MPC para controlar orientación sin restricciones.	44
4.12	Comparación entre MPC y control PID aplicado a control de orientación θ	45
4.13	Aplicación de MPC para controlar orientación sometiendo el dirigible a perturbaciones.	46

4.14	Aplicación de MPC para controlar orientación aplicando restricciones al momento que generan los motores	47
6.1	Diagrama de simulación en Simulink para control PID de altura.	52
6.2	Diagrama de simulación en Simulink para control PID de orientación.	52

Índice de tablas

2.1	Ganancias del control PID, para altura y orientación del dirigible.	12
3.1	Parámetros de transmisión de sistema de transmisión de datos.	25
3.2	Masa de los componentes del dirigible.	30
4.1	Parámetros identificados experimentalmente.	36
4.2	Especificaciones de Filtro de Kalman para estimación de z y \dot{z}	38
4.3	Especificaciones de Filtro de Kalman para estimación de θ y $\dot{\theta}$	39
4.4	Especificaciones de MPC para control de posición en eje z	40
4.5	Especificaciones de MPC para control de orientación θ	44

Nomenclatura

MPC	Model Predictive control, o Control Predictivo Basado en Modelo
SISO	Single Input Single Output
MIMO	Multiple Input Multiple Output
PWM	Modulación por Ancho de Pulso (Pulse Wave Modulation)
<i>Planta</i>	Sistema físico al cual se le aplicará la señal de control
c_x	Centro del sensor de la cámara en eje x
c_y	Centro del sensor de la cámara en eje y
D	Coefficiente de amortiguamiento
f_x	Distancia focal en x
f_y	Distancia focal en y
f_z	Fuerza en eje z
I_z	Inercia respecto a eje z
k_p	Constante proporcional
k_{int}	Constante integral
k_d	Constante derivativa
m	Masa
M_z	Momento en eje z
u	Señal de control
Δu	Incremento en señal de control
y	Salida de sistema de control
\mathbf{x}	vector de estados
$\hat{\mathbf{x}}$	Predicción del vector de estados
\mathbf{y}	Salida de estados del sistema de control
\mathbf{A}	Matriz de estados
\mathbf{B}	Matriz de entrada
\mathbf{C}	Matriz de salida
\mathbf{C}_t	Matriz triangular inferior
$\mathbf{C}_{trabajo}$	Ciclo de trabajo de señal PWM
\mathbf{I}	Matriz de identidad
\mathbf{J}_{MPC}	Función de costo
\mathbf{K}	Ganancia de Kalman
\mathbf{N}_c	Horizonte de control
\mathbf{N}_p	Horizonte de predicción
\mathbf{P}	Matriz de covarianza
\mathbf{Q}	Matriz de covarianza de ruido en mediciones
\mathbf{V}	Covarianza de ruido en el proceso (perturbaciones)
\mathbf{V}_{Motor}	Voltaje en motores
\mathbf{v}	Ruido en mediciones
\mathbf{W}_x	Matriz de ponderación de salidas
$\mathbf{W}_{\Delta u}$	Matriz de ponderación de entradas
\mathbf{w}	Ruido en el proceso del sistema
\mathbf{Y}	Vector de predicción de salidas

Símbolos griegos

- ω Velocidad angular
 θ Ángulo de giro respecto a eje z

Subíndices

- i Iteración en el MPC
 k Posición de la muestra respecto a la iteración i en el MPC
 min Valor mínimo
 max Valor máximo

CAPÍTULO 1

Introducción

Desde su creación en el año 1785 y hasta el día de hoy, los dirigibles han sido utilizados en diversos ámbitos, dentro de los cuales se encuentran el levantamiento de carga, el monitoreo de locales comerciales, la vigilancia aérea, el transporte y también se han desarrollado aplicaciones militares [1]. Últimamente se han utilizado en algunos casos para reemplazar vehículos aéreos tales como los cuadricópteros, debido a que estos tienen una baja autonomía por unidad de carga de la batería, comúnmente menor a 20 minutos. Esta baja autonomía restringe sus aplicaciones y la duración de las pruebas de vuelo. En cambio, un dirigible podría mantenerse en el aire por más de dos horas [2], lo que representa una ventaja para las operaciones que requieren una mayor duración de vuelo, como pueden ser las aplicaciones de vigilancia o monitoreo de larga duración [3].

La última década se han desarrollado dirigibles para una amplia variedad de aplicaciones, tales como exposiciones artísticas en museos [4], emulación de vehículos acuáticos [5], y aplicaciones de entretenimiento en localidades con gran concurrencia de personas, demostrando la capacidad de realizar maniobras complejas tanto en ambientes interiores como exteriores, con un bajo riesgo para los asistentes [6]. También existen estudios que buscan la aplicación de dirigibles en Plataformas Estratosféricas o HAPS (High Altitude Platform System). Estas son plataformas que funcionan en la estratósfera, y son diseñadas para proporcionar redes de comunicación inalámbrica (telecomunicación 3G) [7], en esta área se puede mencionar el trabajo realizado en Chile por el profesor Fernando Ulloa Vásquez, donde se fabricó en el año 2013 un prototipo de dirigible utilizado para transmitir Redes inalámbricas [8].

La mayoría de las investigaciones realizadas a la fecha sobre el control de posicionamiento de dirigibles autónomos están basadas en el control de posicionamiento utilizando controladores PID [1][9][10], incluso utilizando Aprendizaje Reforzado para ajustar los parámetros de dicho sistema de control [11]. Por lo que se puede decir que no se ha explorado en profundidad la aplicación de teoría de control moderno como el Control Predictivo Basado en Modelo. La aplicación más cercana a lo que se espera desarrollar en este proyecto, es la investigación realizada en el Research into Artefacts Center for Engineering de la Universidad de Tokyo. Donde se desarrolló un dirigible autónomo implementando Control Predictivo Basado en Modelo considerando las restricciones en las entradas y salidas del sistema (altura, desplazamiento y voltaje en motores) [12]. Sin embargo, la investigación realizada en la Universidad de Tokyo utiliza un sistema láser para determinar la posición del dirigible, y esto limita la movilidad del dirigible.

El Control Predictivo Basado en Modelo o MPC (por sus siglas en inglés) es una estrategia de control ampliamente utilizada tanto en el área industrial, como en aplicaciones académicas [13]. Debido a que es uno de los pocos sistemas de control que tienen la capacidad de lidiar con sistemas con múltiples entradas y múltiples salidas, los cuales podrían tener restricciones de operación (en entradas y salidas)[13]. Esto representa una ventaja respecto a los controladores PID.

Este proyecto de investigación se realizará en un dirigible a escala diseñado para operar en interiores, ya que en interiores se evitan las posibles perturbaciones generadas por el clima, simplificando el trabajo desde esta perspectiva. Pero al trabajar en interiores se generan restricciones sobre las condiciones de operación tanto en la altitud y desplazamiento máximo, ya que se limitan los rangos de movimiento. También existen restricciones de hardware, como puede ser la restricción sobre el voltaje máximo que se puede aplicar a los motores, y además se vuelve más complejo el sistema de localización, al no poder utilizar un GPS. A partir de esta investigación se espera diseñar un sistema de control que considere las restricciones mencionadas mediante la aplicación de Control Predictivo Basado en Modelo, diferenciándose respecto a las investigaciones realizadas hasta la fecha, las cuales utilizan principalmente controladores PID y no tienen sus sensores de medición a bordo del dispositivo. Esto añade complejidad a la construcción y selección de componentes debido a la limitación de la fuerza de flotabilidad generada por el helio, ya que el peso de los dirigible debe ser menor a dicha fuerza.

Mediante el desarrollo de este trabajo se espera comparar el desempeño del control predictivo basado en Modelo con el control clásico o PID.

1.1 Hipótesis

La hipótesis que sustenta el desarrollo de este trabajo es:

- El control PID comúnmente aplicado en el control de dirigibles para interiores, podría ser reemplazado por control predictivo basado en modelo, ya que este tipo de control permite generar predicciones sobre el comportamiento del sistema, generando una señal de control optimizada para dicha predicción.

1.2 Objetivos

1.2.1 Objetivo general

El Objetivo General de este trabajo es:

- Evaluar experimentalmente la capacidad de un dirigible de seguir referencias mediante la aplicación de control predictivo basado en modelo.

1.2.2 Objetivos específicos

Los Objetivos Específicos de este trabajo son:

- OE1: Diseñar un dirigible a escala, incluyendo la selección de: motores, driver de motores, microcontrolador, globo, transceptor inalámbrico, y sensores para determinar posición y orientación.

- OE2: Desarrollar un código para el procesamiento en tiempo real de la información obtenida por los sensores.
- OE3: Realizar identificación de sistemas para determinar la función de transferencia del dirigible.
- OE4: Diseñar el Control Predictivo adecuado para la dinámica del sistema.
- OE5: Evaluar el desempeño del Control Predictivo diseñado, teniendo como referencia un control PID.

1.3 Metodología

La metodología desarrollada para cumplir cada uno de los objetivos de este trabajo es la siguiente:

OE1: Para realizar las pruebas en primer lugar se debe construir el dirigible con el que se trabajará. Se deben seleccionar los componentes adecuados, como son el driver de los motores, microcontrolador, globo, transceptor inalámbrico, sensores de medición inercial para determinar orientación, y sensores para determinar desplazamiento y altura del dirigible.

OE2: La información entregada por los sensores a bordo del dirigible es transmitida por medio del transceptor a través de comunicación serial a un computador en terreno, y esta información es procesada a través de Matlab. A partir de la información obtenida por los sensores se debe crear un programa que procese los datos provenientes de los sensores, con el fin de cuantificar la altura, orientación y el desplazamiento del dispositivo.

OE3: Se utiliza una señal de prueba enviada a los motores del dirigible, para obtener su comportamiento ante dicha señal, y por medio de técnicas de identificación de sistemas se determina las ecuaciones que modelan la dinámica del dirigible. Determinando funciones de transferencia y el modelo en Espacio de Estados a través de la respuesta de altura, orientación angular, desplazamiento y sus derivadas. Se utilizarán tres señales de prueba diferentes (una para cada tipo de movimiento), considerando el dispositivo como una caja gris.

OE4: A partir de la función de transferencia del sistema se determina el horizonte de predicción y el horizonte de control mediante simulaciones. Se incorporan las restricciones del sistema (voltaje, y límites de desplazamiento), y se determinan las matrices de ponderación a utilizar en la función de costo del control predictivo. Todas estas variables son analizadas mediante simulación en Matlab y posteriormente aplicadas al dirigible para verificar que el controlador diseñado mediante simulación funciona en la realidad.

OE5: Se compara un controlador PID con el Control Predictivo, y se analizan ventajas y desventajas de cada sistema de control. Las pruebas se realizarán en una habitación de 2.3 metros de alto, 5 metros de ancho y 6 metros de largo. Se comparará la calidad en la respuesta del dirigible al seguir trayectorias y referencias establecidas, mediante la utilización de control PID y MPC. Por ejemplo, en el caso de la altura se establecerá una altura objetivo que será 1 metro y se evaluarán ambos controles.

CAPÍTULO 2

Marco teórico

2.1 Ecuaciones de movimiento

Para diseñar el sistema de control se debe determinar las ecuaciones de movimiento del dirigible (con sus respectivos parámetros). En esta investigación se utilizará un modelo con 3 grados de libertad, en el cual la fuerza de los motores que generan avance no generan momento en el eje y o x de acuerdo al esquema de las figuras 2.1 y 2.2, misma simplificación que la utilizada en [14] donde se aplicó un sistema de control PID en los 3 sentidos de movimiento. Esta simplificación se utiliza debido a que de otra forma el sistema es no-lineal y acoplado, lo que dificulta la aplicación del sistema de control propuesto en este proyecto.

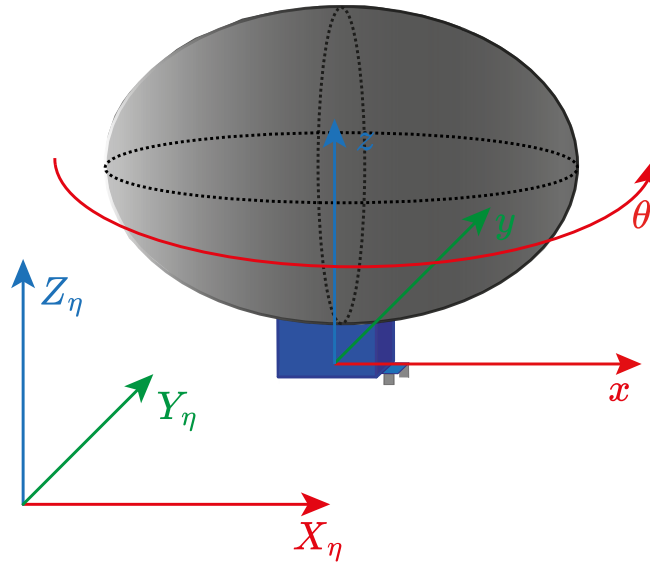


Figura 2.1: Esquema posicionamiento y orientación del dirigible.

El globo a utilizar es de perfil circular en el plano xy como se observa en la figura 2.1. Al agregar Helio al dirigible se agrega el lastre suficiente para generar una fuerza de flotabilidad neutra. De acuerdo al esquema de la figura 2.1 y considerando la flotabilidad neutra, las ecuaciones de movimiento primitivas del dirigible son las siguientes:

$$m_{xy} \cdot \dot{v}_{xy} + D_{xy} \cdot v_{xy} = f_{xy} \quad (2.1)$$

$$m_z \cdot \dot{v}_z^b + D_z \cdot v_z = f_z \quad (2.2)$$

$$I_z \dot{\omega} + D_\theta \omega = M_z \quad (2.3)$$

Donde m_{xy} y m_z corresponden a la suma de masa de los componentes y la masa añadida (por arrastre de aire) del dirigible en la dirección horizontal (ejes x e y) y dirección vertical (eje z) respectivamente. D_z , D_{xy} y D_θ corresponden al amortiguamiento debido al roce con el aire. f_{xy} y f_z son las fuerzas que generan desplazamiento en el dirigible. M_z es el momento generado en el eje z . v_{xy} y v_z son velocidades horizontal y vertical.

Los 3 movimientos a controlar para seguir patrones de movimiento son los siguientes:

- **Posición en el eje z :** El dirigible debe seguir referencias de altitud y mantener la posición constante mientras se desplaza.
- **Orientación θ :** Se debe regular la orientación de manera que se pueda regular la dirección en la cual el dirigible se desplaza.
- **Velocidad en dirección horizontal v_x :** La velocidad se debe mantener constante mientras el dirigible se está desplazando.

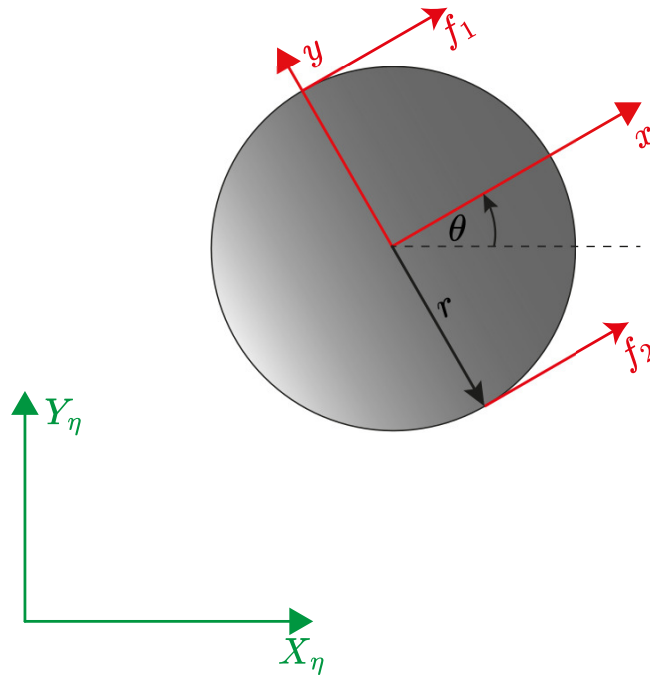


Figura 2.2: Esquema de orientación del dirigible respecto al marco de referencia η .

La aplicación del sistema de control MPC requiere que las ecuaciones de la dinámica del sistema estén descritas en espacio de estados. Por este motivo es necesario reordenar las ecuaciones de movimiento.

A partir de la ecuación 2.2 se obtiene la siguiente expresión en espacio de estados:

$$\begin{pmatrix} \dot{x}_z \\ \ddot{x}_z \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 0 & -D_z/m_z \end{pmatrix} \begin{pmatrix} x_z \\ \dot{x}_z \end{pmatrix} + \begin{pmatrix} 0 \\ 1/m_z \end{pmatrix} (f_z) \quad (2.4)$$

En el caso del desplazamiento y la orientación en el plano horizontal, las ecuaciones 2.1 y 2.3 se pueden combinar utilizando el esquema que se muestra en la figura 2.2 donde se presentan las fuerzas f_1 y f_2 , la cuales corresponden a la fuerza ejercida por los motores responsables del desplazamiento y rotación del dirigible.

$$\begin{pmatrix} \dot{x}_x \\ \ddot{x}_x \\ \dot{\theta} \\ \ddot{\theta} \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & -D_x/m_z & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & -D_\theta/I_z \end{pmatrix} \begin{pmatrix} x_x \\ \dot{x}_x \\ \theta \\ \dot{\theta} \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ 1/m_z & 1/m_z \\ 0 & 0 \\ -r/I_z & r/I_z \end{pmatrix} \begin{pmatrix} f_1 \\ f_2 \end{pmatrix} \quad (2.5)$$

Las ecuaciones 2.4 y 2.5 son la utilizadas en la teoría de Control Predictivo Basado en Modelo.

2.2 Identificación de sistemas

Previo a diseñar el sistema de control se debe determinar el modelo matemático que aproxime la respuesta de la planta (en este caso el dirigible) en relación a un determinado tipo de variable de entrada, considerando sus condiciones de operación. Existen dos métodos principales para determinar el modelo del sistema[15], los cuales son:

White box model: Se determinan de forma analítica las principales características físicas del modelo, a través de leyes de Newton, conservación de energía, etc. Este método tiene mayor complejidad cuando existen propiedades físicas difíciles de determinar en el sistema.

Black box model Se aproxima la relación entre la entrada y la salida del sistema, sin considerar las propiedades físicas del sistemas, es decir, el sistema es considerado una caja negra. Algunos de los métodos mas utilizados son los modelos no-lineales ARX y modelos de Hammerstein-Wiener.

Existe un método intermedio llamado *Grey box model* el cual es una combinación de los métodos mencionados anteriormente. Consiste en determinar la ecuación dinámica del la planta y aproximar sus parámetros a través de los datos obtenidos en un ensayo experimental, utilizando una señal de entrada conocida y midiendo su variable de salida.

Para determinar los parámetros m_z , m_{xy} , D_z , D_{xy} , D_θ y I_z , se utilizan las ecuaciones 2.1, 2.2, y 2.3 enviando una señal de una entrada conocida para cada movimiento, y ajustando la curva de respuesta mediante un algoritmo de mínimos cuadrados.

2.3 Control PID

Los controladores PID (proporcional, integral y derivativo) son indudablemente los controladores mas utilizados en la actualidad por su eficacia y versatilidad. Cerca del 95% de los lazos de control emplean un PID [16]. La estructura universal del controlador PID en tiempo continuo tiene la siguiente forma:

$$u(t) = k_p e(t) + k_i \int e(t) dt + k_d \frac{de(t)}{dt} \quad (2.6)$$

Donde k_p corresponde a la constante de proporcionalidad, k_i es la constante proporcional a la integración del error y k_d es la constante proporcional a la derivada del error. Estas 3 variables son parámetros a ajustar de acuerdo a la respuesta deseada en la dinámica del sistema. Obteniendo el diagrama de bloques de la figura 2.3.

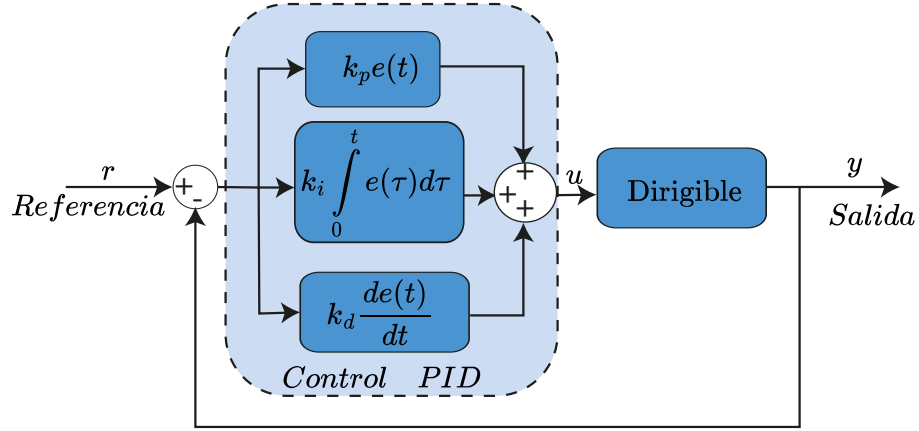


Figura 2.3: Diagrama de bloques de control PID.

Para comparar las respuestas obtenidas mediante el Control Predictivo Basado en Modelo desarrollado en este trabajo, se utilizarán los resultados obtenidos en el dirigible GT-MAB, desarrollado en el Georgia Institute of Technology [17], donde se aplicó un control PID para controlar velocidad horizontal, altitud y orientación en un dirigible con el mismo radio (91 cm) que el dirigible utilizado en este trabajo. las funciones de transferencia en lazo abierto determinadas para el GT-MAB son las siguientes:

$$P_2 = \frac{11.4912}{s^3 + 6.0088s^2 + 0.7453s} \quad (2.7)$$

$$P_3 = \frac{64.3324}{s^3 + 6.0676s^2 + 1.0912s} \quad (2.8)$$

Donde P_2 representa a la altura del dirigible respecto a la fuerza f_z , y P_3 representa el ángulo del dirigible respecto al torque aplicado al momento M_z . Los valores del controlador PID obtenidos en dicha investigación fueron sintonizados mediante Matlab, donde obtuvieron los siguientes valores:

Tabla 2.1: Ganancias del control PID, para altura y orientación del dirigible.

Dirección	k_p	k_i	k_d
Altura x_z	0.656	0.0087	0.7352
Orientación θ	0.1955	0	0.192

Cabe destacar que estos valores fueron simulados y validados para el GT-MAB, en el trabajo anteriormente mencionado.

2.4 Control Predictivo Basado en Modelo

El Control Predictivo Basado en Modelo o MPC (por sus siglas en inglés) es un sistema de control optimizado que tiene como objetivo realizar predicciones sobre la magnitud de la señal de control u que se debe aplicar a la planta para alcanzar una referencia deseada en la señal de salida y . Para determinar la señal de control se utiliza el modelo del sistema y se proyectan hacia el futuro las entradas óptimas para que las salidas (velocidad y posición) alcancen las referencias deseadas.

Para implementar el MPC en un dirigible, es necesario desarrollar un modelo matemático de la dinámica del dirigible, que debe incluir la respuesta a diferentes entradas de control como la propulsión y el ascenso. Dicho modelo debe estar representado en Espacio de Estados en su forma discretizada. La discretización del modelo en Espacio de Estados es un punto clave ya que el control se aplicará en un sistema digital, y esta representación permite predecir una muestra hacia el futuro en los estados del sistema de acuerdo a su modelo, como se puede apreciar en la ecuación 2.9.

$$\begin{aligned}\mathbf{x}_{k+1} &= \mathbf{A}\mathbf{x}_k + \mathbf{B}u_k \\ \mathbf{y}_k &= \mathbf{C}\mathbf{x}_k\end{aligned}\tag{2.9}$$

Al igual que en [18] se utiliza una representación incremental para la variable de entrada del sistema Δu_k en lugar de la muestra actual u_k . El incremento de la señal de control está dado por $\Delta u_k = u_k - u_{k-1}$. Realizando este cambio en la ecuación 2.9 se obtiene la siguiente representación del modelo aumentado:

$$\begin{bmatrix} \mathbf{x}_{k+1} \\ u_k \end{bmatrix} = \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{x}_k \\ u_{k-1} \end{bmatrix} + \begin{bmatrix} \mathbf{B} \\ \mathbf{I} \end{bmatrix} \Delta u_k\tag{2.10}$$

$$\mathbf{y}_k = [\mathbf{C} \quad \mathbf{0}] \begin{bmatrix} \mathbf{x}_k \\ u_{k-1} \end{bmatrix}$$

Se redefinen las matrices \mathbf{A} , \mathbf{B} , \mathbf{C} y el vector de estados \mathbf{x}_k de la ecuación 2.9 para el modelo aumentado, de la siguiente forma:

$$\mathbf{A} = \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{0} & \mathbf{I} \end{bmatrix}; \quad \mathbf{B} = \begin{bmatrix} \mathbf{B} \\ \mathbf{I} \end{bmatrix}; \quad \mathbf{C} = [\mathbf{C} \quad \mathbf{0}]; \quad \mathbf{x}_k = \begin{bmatrix} \mathbf{x}_k \\ u_{k-1} \end{bmatrix}\tag{2.11}$$

En el modelo aumentado de la ecuación 2.10 también se genera una predicción de una muestra de los estados futuros de la planta \mathbf{x}_{k+1} . Esta predicción se puede utilizar de forma recursiva sobre

la misma ecuación, generando una ventana de predicciones sobre los estados futuros. Esto se realiza considerando una predicción de N_p muestras en las salidas (hacia el futuro), y N_c señales de control aplicadas para alcanzar la referencia, obteniendo el siguiente sistema de ecuaciones:

$$\begin{aligned}
\mathbf{y}_{k_i+1|k_i} &= \mathbf{CA}\mathbf{x}_{k_i} + \mathbf{CB}\Delta u_{k_i} \\
\mathbf{y}_{k_i+2|k_i} &= \mathbf{CA}^2\mathbf{x}_{k_i} + \mathbf{CAB}\Delta u_{k_i} + \mathbf{CB}\Delta u_{k_i+1} \\
\mathbf{y}_{k_i+3|k_i} &= \mathbf{CA}^3\mathbf{x}_{k_i} + \mathbf{CA}^2\mathbf{B}\Delta u_{k_i} + \mathbf{CA}^2\mathbf{B}\Delta u_{k_i+1} + \mathbf{CB}\Delta u_{k_i+2} \\
\mathbf{y}_{k_i+N_p|k_i} &= \mathbf{CA}^{N_p}\mathbf{x}_{k_i} + \mathbf{CA}^{N_p-1}\mathbf{B}\Delta u_{k_i} + \mathbf{CA}^{N_p-2}\mathbf{B}\Delta u_{k_i+1} \\
&\quad + \dots + \mathbf{CA}^{N_p-N_c}\mathbf{B}\Delta u_{k_i+N_c-1}
\end{aligned} \tag{2.12}$$

Donde la variable k_i representa la muestra k en la iteración i . El sistema de ecuaciones 2.12 puede ser representado de forma matricial de acuerdo a [13]. Utilizando un vector de salidas \mathbf{Y} , y un vector de entradas $\Delta\mathbf{U}$ la representación matricial tiene la siguiente forma:

$$\mathbf{Y} = \begin{bmatrix} \mathbf{y}_{k_i+1|k_i} \\ \mathbf{y}_{k_i+2|k_i} \\ \mathbf{y}_{k_i+3|k_i} \\ \vdots \\ \mathbf{y}_{k_i+N_p|k_i} \end{bmatrix}, \quad \Delta\mathbf{U} = \begin{bmatrix} \Delta u_{k_i} \\ \Delta u_{k_i+1} \\ \Delta u_{k_i+2} \\ \vdots \\ \Delta u_{k_i+N_c-1} \end{bmatrix} \tag{2.13}$$

$$\mathbf{F} = \begin{bmatrix} \mathbf{CA} \\ \mathbf{CA}^2 \\ \mathbf{CA}^3 \\ \vdots \\ \mathbf{CA}^{N_p} \end{bmatrix}, \quad \mathbf{H} = \begin{bmatrix} \mathbf{CB} & 0 & 0 & \dots & 0 \\ \mathbf{CAB} & \mathbf{CB} & 0 & \dots & 0 \\ \mathbf{CA}^2\mathbf{B} & \mathbf{CAB} & \mathbf{CB} & \dots & 0 \\ \vdots & \vdots & \vdots & \dots & \vdots \\ \mathbf{CA}^{N_p-1}\mathbf{B} & \mathbf{CA}^{N_p-2}\mathbf{B} & \mathbf{CA}^{N_p-3}\mathbf{B} & \dots & \mathbf{CA}^{N_p-N_c}\mathbf{B} \end{bmatrix} \tag{2.14}$$

$$\mathbf{Y} = \mathbf{F}\mathbf{x}_{k_i} + \mathbf{H}\Delta\mathbf{U} \tag{2.15}$$

La ventana de predicción de estados hacia el futuro es llamada "horizonte de predicción" (representada por N_p). El control predictivo es planteado como un problema de optimización mediante el planteamiento de una función de costo. A través de la solución de la función de costo se obtendrá una cierta cantidad de señales de control futuras. La cantidad de señales de control futuras se expresan mediante el "horizonte de control" (denotado como N_c). Este proceso se repite para cada muestra y solo se considera la primera señal de control de en cada iteración, lo que se conoce como "horizonte móvil". El esquema de aplicación del control predictivo se muestra en la figura 2.4.

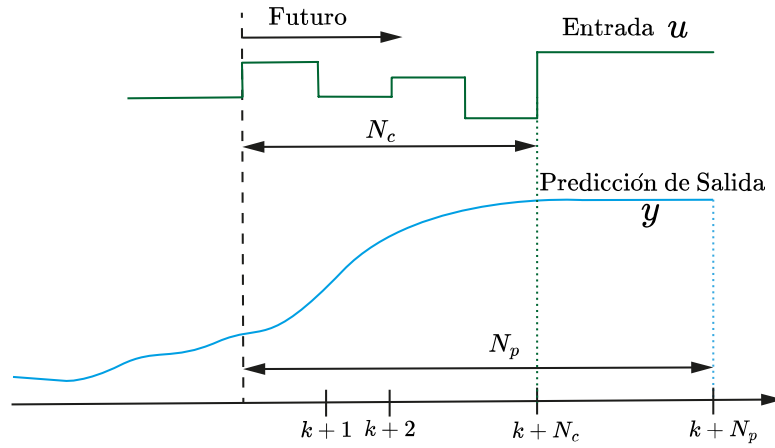


Figura 2.4: Esquema de predicciones del MPC.

2.4.1 Función de costo

La función de costo del MPC será una combinación lineal de los estados y las entradas del sistema.

$$J_{MPC} = J_y + J_{\Delta u} \quad (2.16)$$

La función J_y representa la función de costo para el error entre las referencias \mathbf{R} y las salidas \mathbf{Y} , y es definida de la siguiente forma:

$$\begin{aligned} J_y &= (\mathbf{R} - \mathbf{Y})^T \mathbf{W}_x (\mathbf{R} - \mathbf{Y}) \\ &= (\mathbf{R} - \mathbf{F}\mathbf{x}_{k_i} - \mathbf{H}\Delta\mathbf{U})^T \mathbf{W}_x (\mathbf{R} - \mathbf{F}\mathbf{x}_{k_i} - \mathbf{H}\Delta\mathbf{U}) \end{aligned} \quad (2.17)$$

Donde la matriz \mathbf{W}_x es una matriz de peso aplicada al error entre las salidas y sus referencias. Los valores de \mathbf{W}_x y el vector de referencias \mathbf{R} se representan de la siguiente forma:

$$\mathbf{R} = [r_x, r_{\dot{x}}, \dots, r_x, r_{\dot{x}}]^T, \quad \mathbf{W}_x = \begin{bmatrix} w_x & 0 & \dots & 0 \\ 0 & w_{\dot{x}} & 0 & \dots & 0 \\ \vdots & 0 & \ddots & & \vdots \\ 0 & \vdots & & w_x & 0 \\ 0 & 0 & \dots & 0 & w_{\dot{x}} \end{bmatrix} \quad (2.18)$$

En las expresiones de 2.18 los valores r_x y $r_{\dot{x}}$ representan las referencias para posición y velocidad respectivamente, y el vector \mathbf{R} es de dimensión $(2N_p, 1)$. Los valores w_x y $w_{\dot{x}}$ representan el peso que se asignará a cada variable de estado y son valores a ajustar dependiendo de cual es la variable a la cual se le asignará mayor importancia en alcanzar su referencia (posición o velocidad), la dimensión de la matriz \mathbf{W}_x es $(2N_p, 2N_p)$.

Por otro lado $J_{\Delta u}$ es la función de costo para el vector de entradas. Esta función de costo es definida por la siguiente expresión:

$$J_{\Delta u} = \Delta \mathbf{U}^T \mathbf{W}_{\Delta u} \Delta \mathbf{U} \quad (2.19)$$

Donde $\mathbf{W}_{\Delta u}$ es el vector de ponderación para las entradas del sistema es de dimensión $(N_c, 1)$, y tiene la siguiente forma:

$$\mathbf{W}_{\Delta u} = [w_u, \dots, w_u]' \quad (2.20)$$

Donde w_u es un valor de ponderación para la salida del sistema.

Reemplazando las ecuaciones 2.17 y 2.19 en 2.16 y reagrupando terminos se obtiene la función de costo total del sistema en función de sus ponderaciones:

$$J_{MPC} = \Delta \mathbf{U}^T (\mathbf{H}^T \mathbf{W}_x \mathbf{H} + \mathbf{W}_{\Delta u}) \Delta \mathbf{U} - 2(\mathbf{R} - \mathbf{F}\mathbf{x})^T \mathbf{W}_x \mathbf{H} \Delta \mathbf{U} \quad (2.21)$$

El valor $\Delta \mathbf{U}_k$ se obtiene minimizando la función de costo de la ecuación 2.21 mediante programación cuadrática o QP (por sus siglas en inglés). Esto se realiza mediante la función "quadprog" en Matlab y es resuelto en cada muestra. Dicha función tiene una estructura definida por:

$$\min_{\Delta \mathbf{U}} \frac{1}{2} \Delta \mathbf{U}^T \mathbf{h} \Delta \mathbf{U} + \mathbf{f}^T \Delta \mathbf{U} \quad \text{tal que } \mathbf{N} \Delta \mathbf{U} \leq \mathbf{b} \quad (2.22)$$

Donde la expresión $\mathbf{N} \Delta \mathbf{U} \leq \mathbf{b}$ representa las restricciones del sistema representadas como desigualdad, y los valores de h y f se determinan de acuerdo a la función de costo de la ecuación 2.21. Obteniendo la siguiente estructura:

$$\begin{aligned} \mathbf{h} &= 2(\mathbf{H}^T \mathbf{W}_x \mathbf{H} + \mathbf{W}_{\Delta u}) \\ \mathbf{f} &= -2((\mathbf{R} - \mathbf{F}\mathbf{x})^T \mathbf{W}_x \mathbf{H})^T \end{aligned} \quad (2.23)$$

2.4.2 Restricciones

En la función quadprog de Matlab se pueden agregar las restricciones tanto en las salidas como en las entradas del sistema. Dichas restricciones deben ser expresadas como desigualdades. A partir de [13] se desarrollan las restricciones en salida y entrada.

Restricciones en señal de salida

Las salidas del sistema están limitadas por un valor mínimo \mathbf{Y}_{min} y un valor máximo \mathbf{Y}_{max} .

$$\mathbf{Y}_{min} \leq \mathbf{F}\mathbf{x}_{k_i} + \mathbf{H}\Delta \mathbf{U} \leq \mathbf{Y}_{max} \quad (2.24)$$

La desigualdad de la ecuación 2.24 puede ser representada por sus restricciones de mínimo y máximo por separado:

$$\begin{aligned} -\mathbf{H}\Delta\mathbf{U} &\leq -\mathbf{Y}_{min} + \mathbf{F}\mathbf{x}_{k_i} \\ \mathbf{H}\Delta\mathbf{U} &\leq \mathbf{Y}_{max} - \mathbf{F}\mathbf{x}_{k_i} \end{aligned} \quad (2.25)$$

Restricciones en señal de control

Las restricciones en la amplitud en la señal de control se pueden expresar de la siguiente forma:

$$u_{min} \leq u_k \leq u_{max} \quad (2.26)$$

Considerando que la entrada es representada en el modelo aumentado como $\Delta u_{k_i} = u_{k_i} - u_{k_i-1}$, la ecuación 2.26 puede ser reordenada como se muestra a continuación:

$$\begin{aligned} \Delta u_{k_i} &\leq u_{max} - u_{k_i-1} \\ -\Delta u_{k_i} &\leq -u_{min} + u_{k_i-1} \end{aligned} \quad (2.27)$$

Denotando a \mathbf{C}_t como una matriz triangular inferior con valores 1 de dimensión $N_c \times N_c$. Las restricciones en la amplitud de la salida se pueden definir como:

$$\begin{bmatrix} \mathbf{C}_t \\ -\mathbf{C}_t \end{bmatrix} \begin{bmatrix} \Delta u_{k_i+1} \\ \Delta u_{k_i+2} \\ \vdots \\ \Delta u_{k_i-1} \end{bmatrix} \leq \begin{bmatrix} u_{max} - u_{k_i-1} \\ \vdots \\ u_{max} - u_{k_i-1} \\ u_{min} + u_{k_i-1} \\ \vdots \\ u_{min} + u_{k_i-1} \end{bmatrix} \quad (2.28)$$

La desigualdad 2.28 puede ser representada de la siguiente forma:

$$\begin{bmatrix} \mathbf{C}_t \\ -\mathbf{C}_t \end{bmatrix} \Delta\mathbf{U} \leq [\mathbf{M}] \quad (2.29)$$

Donde \mathbf{M} representa el lado derecho de la desigualdad 2.28, con una dimension $2N_c \times 1$

Unión de Restricciones en salida y entrada

Utilizando las desigualdades 2.25 y 2.29 obtenidas para las entradas y salidas del sistema en función de $\Delta\mathbf{U}$, se obtiene la representación de las salidas con la misma estructura requerida por la función "quadprog", representada en 2.22.

$$\begin{bmatrix} -\mathbf{H} \\ \mathbf{H} \\ \mathbf{C}_t \\ -\mathbf{C}_t \end{bmatrix} \Delta \mathbf{U} \leq \begin{bmatrix} -\mathbf{Y}_{min} + \mathbf{F}\mathbf{x}_{k_i} \\ \mathbf{Y}_{max} - \mathbf{F}\mathbf{x}_{k_i} \\ \mathbf{M} \end{bmatrix} \quad (2.30)$$

En la figura 2.5 se muestra el diagrama de bloques del control MPC, con el bloque de optimización de la función de costo y el diagrama de flujo de las señales de entrada y salida.

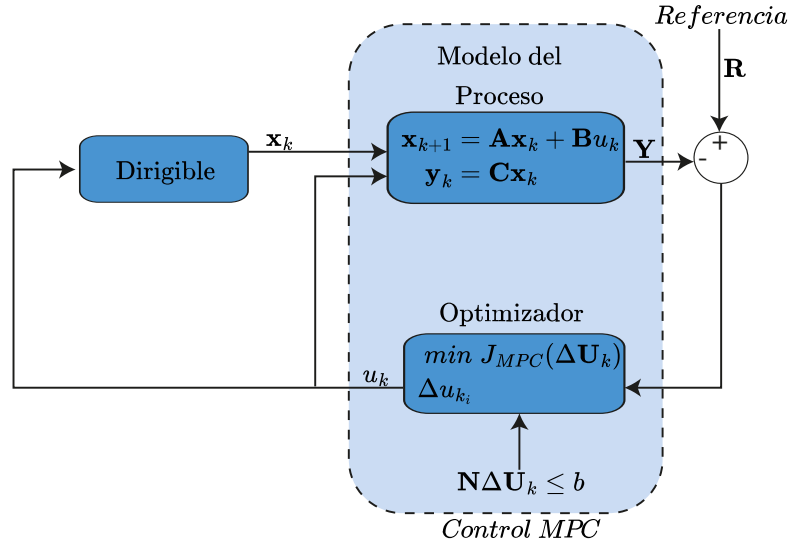


Figura 2.5: Diagrama de bloques de Control Predictivo Basado en Modelo.

2.5 Filtro de Kalman

Al diseñar un Control Predictivo Basado en Modelo se asume que los estados $\mathbf{x}(k)$ se pueden medir para cada muestreo k_i . En muchas aplicaciones reales esto no es así, ya que algunos de los estados podrían ser imposibles de medir directamente o surgen a partir de derivación o integración de otro estado medido de forma directa (como la velocidad a partir de la posición), lo que puede generar errores debido a problemas como ruido en la señal medida. Una solución a esto es la utilización del mismo modelo en Espacio de Estados para estimar o aproximar los estados del sistema en conjunto con la información de la variable medida. Esta aproximación es llamada Observador de Estados, y puede servir tanto para estimar los estados como para reducir el efecto del ruido en las mediciones.

El filtro de Kalman puede ser utilizado como un observador usando el modelo del sistema. Su principal ventaja es que permite agregar el efecto del ruido en las mediciones y el efecto de posibles perturbaciones al sistema. El filtro diseñado en esta investigación está basado en el trabajo realizado en [19]. A continuación se presenta su estructura general:

$$\begin{aligned} \hat{\mathbf{x}}_{k|k-1} &= \mathbf{A}\hat{\mathbf{x}}_{k-1} + \mathbf{B}u_{k-1} + \mathbf{w}_{k-1} \\ \mathbf{y}_k &= \mathbf{C}\hat{\mathbf{x}}_k + \mathbf{v}_k \end{aligned} \quad (2.31)$$

Donde $\hat{\mathbf{x}}_{k|k-1}$ corresponde a la predicción de los variables de estado en la iteración actual k considerando los valores pasados $k - 1$. \mathbf{w}_{k-1} es el ruido blanco en el proceso del sistema (posibles perturbaciones) y se puede expresar a través de su varianza \mathbf{Q} . \mathbf{v}_k es el ruido en las mediciones (es decir de los sensores) y se puede representar a través de su varianza \mathbf{V} .

De esta forma el filtro de Kalman entrega una estimación de los estados, utilizando las mediciones anteriores \mathbf{x}_{k-1} , El ruido de los sensores incluido a través de su matriz de covarianza \mathbf{V} y el error debido a posibles perturbaciones incluido mediante la matriz de covarianza \mathbf{Q} .

La selección de los valores de las matrices de covarianza pueden afectar de manera significativa en las estimaciones. Una forma de ver los valores de \mathbf{Q} y \mathbf{V} es como ponderaciones realizadas a las predicciones (confianza en el modelo) y a las mediciones reales (confianza en las mediciones). De esta manera mientras mayores sean los valores de \mathbf{Q} se estima las ecuaciones de estado o el modelo obtenido tiene incertidumbre y el filtro debe corregir más su estimación con la siguiente medición (se tendrá una mayor confianza en las mediciones de los sensores). Por otro lado si los valores de \mathbf{V} son grandes significa que las mediciones tienen una gran cantidad de ruido, por lo que se debe confiar en el modelo y se deberá corregir menos las estimaciones del filtro. El diagrama de la figura 2.6 representa las correcciones que se realizan a la estimación del filtro del Kalman de acuerdo a los valores de \mathbf{Q} y \mathbf{V} .

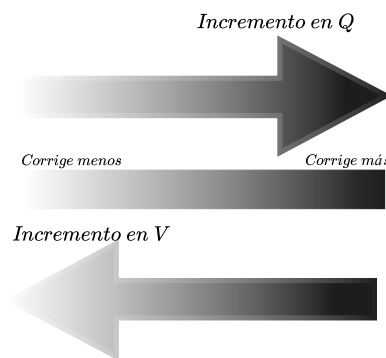


Figura 2.6: Diagrama de efecto de covarianzas \mathbf{Q} y \mathbf{V} en filtro de Kalman [19].

Algoritmo de aplicación del Filtro de Kalman:

La estimación de los estados inicial hace uso de la estimación pasada y la entrada pasada del sistema. Reemplazando estos valores en la ecuación de movimiento de la planta.

$$\begin{aligned}\hat{\mathbf{x}}_{k|k-1} &= \mathbf{A}\hat{\mathbf{x}}_{k-1} + \mathbf{B}u_{k-1} \\ y_k &= \mathbf{C}\hat{\mathbf{x}}_{k|k-1}\end{aligned}\tag{2.32}$$

Donde $\hat{\mathbf{x}}_{k|k-1}$ es la estimación de los estados actuales, basados en $\hat{\mathbf{x}}_{k-1}$ y u_{k-1} que son la predicción de estados anterior y la entrada anterior respectivamente. Posteriormente se determina la covarianza del error asociada a la estimación realizada inicialmente por medio de la ecuación 2.32.

$$\mathbf{P}_{k|k-1} = \mathbf{A}\mathbf{P}_{k-1}\mathbf{A}^T + \mathbf{Q}\tag{2.33}$$

Donde $\mathbf{P}_{k|k-1}$ es la matriz de covarianza asociada al error en los estados, \mathbf{P}_{k-1} es la matriz de covarianza determinada en la muestra anterior, y \mathbf{Q} es la matriz de covarianza asociada al ruido en la planta (posibles perturbaciones). Después de obtener las matrices de covarianza, se procede a obtener la matriz de ganancia de Kalman:

$$\mathbf{K}_k = \mathbf{P}_{k|k-1} \mathbf{C} (\mathbf{C} \mathbf{P}_{k|k-1} \mathbf{C}^T + \mathbf{V})^{-1} \quad (2.34)$$

Donde \mathbf{K}_k es la ganancia de Kalman y \mathbf{v} representa la covarianza del ruido en las mediciones de los estados.

La estimación del vector de estados que se entregará controlador se determina utilizando la estimación inicial $\hat{\mathbf{x}}_{k|k-1}$ y el residuo de la medición, que corresponde a la diferencia entre la medición del sensor y_k (mediante sensor con ruido) y la estimación inicial de la salida $\mathbf{C}\hat{\mathbf{x}}_{k|k-1}$.

$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k (y_k - \mathbf{C}\hat{\mathbf{x}}_{k|k-1}) \quad (2.35)$$

Finalmente, se actualiza la matriz de covarianza utilizando la ganancia de Kalman, y se repite el algoritmo para la siguiente medición.

$$\mathbf{P}_k = (\mathbf{I} - \mathbf{K}_k \mathbf{C}) \mathbf{P}_{k|k-1} \quad (2.36)$$

Las estimaciones de estados \mathbf{x}_k realizadas por el filtro de Kalman son entregadas al control MPC. En la figura 2.7 se muestra el diagrama de bloques del control predictivo basado en modelo en conjunto con el filtro de Kalman.

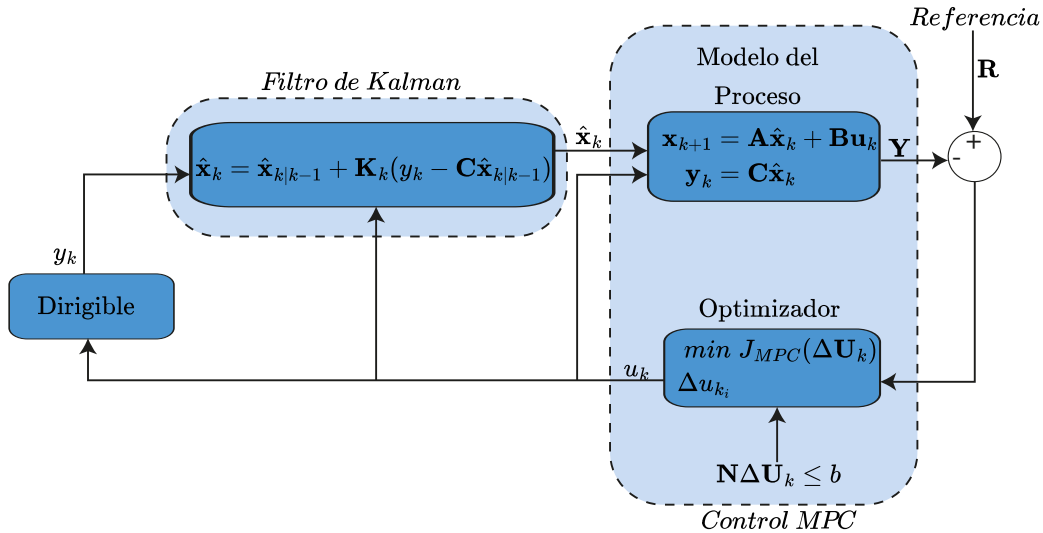


Figura 2.7: Diagrama de bloques de MPC con filtro de Kalman.

A continuación se presenta el algoritmo que se aplica en el dirigible y el computador para generar el control MPC.

Algorithm 1 Mediciones y MPC+KF.

Entrada: Medición de posición de altura x_z y posición angular θ .

Salida: Señal de control (voltaje a motores).

En el microcontrolador Arduino Fio

La tarjeta Arduino Fio espera por la señal de control, la cual recibirá a través del puerto serie por medio del modulo XBee.

Al recibir una señal, realiza la medición de sensor ultrasonido y/o medición de sensor IMU. Se envía la trama de datos mediante el transceptor Xbee a través del modo transparente.

En el computador

El modulo XBee recibe la trama de datos, y estos son enviados por comunicación serial a Matlab (en el PC).

Se estiman los estados de posición y velocidad $\hat{\mathbf{x}}$, mediante el filtro de Kalman en Matlab.

Se minimiza la función de costo, considerando las entradas entregadas por el filtro de Kalman.

La minimización de la función de costo entrega la señal de control a enviar a la tarjeta Arduino.

Se convierte la señal de control a señal PWM de acuerdo a la relación de fuerza y voltaje que tienen los motores.

El tiempo de muestreo es verificado mediante funciones tic y toc en Matlab. Si no se cumple, se entra a un ciclo while hasta que que cumpla.

Se envía la señal PWM a través del XBee a través de comunicación serial hacia el Arduino.

2.6 Localización mediante marcadores

La localización de robots para interiores es un problema ampliamente estudiado en robótica y existen diversos métodos, como la incorporación de marcadores al robot y medición mediante sensores o cámaras externas al robot, o triangulación mediante medición de intensidad de señales inalámbricas [20] [21]. Una de las posibles soluciones a este problema es la visión por computador utilizando marcadores en el espacio de trabajo como lo realizado en [22], [23] y [24]. Considerando lo expuesto anteriormente, la forma mas accesible y factible de determinar la localización es utilizando procesamiento de imágenes para detectar marcadores en el piso en el que se desplazará el dirigible. Al igual que el trabajo realizado en [25] se ubicarán marcadores en el piso, en localizaciones con coordenadas y orientaciones conocidas y fijas, como se puede observar en la figura 2.8.

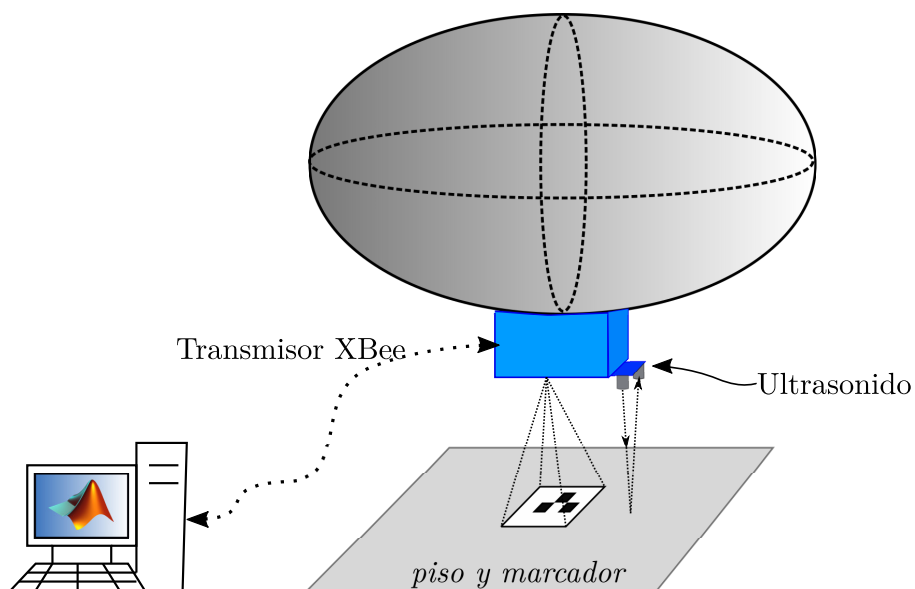


Figura 2.8: Esquema propuesto para localización mediante marcadores.

Mediante la matriz de proyección (que contiene las propiedades de la cámara) es posible determinar la posición y la orientación de la cámara respecto al marcador, y esto se puede convertir al marco de referencia inercial ubicado en un lugar fijo dentro de la habitación en la cual opera el dirigible. En la figura 2.9 se muestran los ejes coordenados del sensor de la cámara (u, v) , con su centro (c_x, c_y) , y las coordenadas del objeto medido (X, Y, Z) .

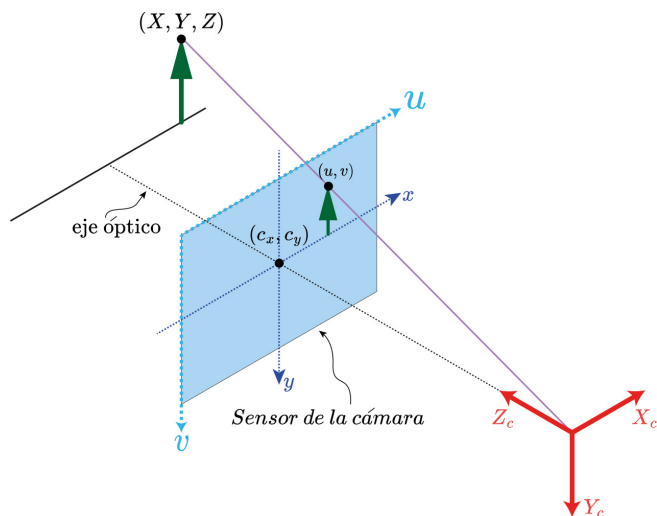


Figura 2.9: Representación de parámetros intrínsecos [25].

El modelo de la cámara es representado a través de sus parámetros intrínsecos, de la siguiente forma:

$$Z_c \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \quad (2.37)$$

Donde f_x y f_y corresponden a las distancias focales (medidas en píxeles), y c_x y c_y corresponden al centro del sensor óptico ccd.

Es necesario corregir posibles distorsiones presentes en la cámara. Para lograr esto es necesario rectificar la imagen a través de los coeficientes de distorsión, los que son representados como:

$$\text{Coeficientes de distorsiones} = [k_1 \quad k_2 \quad p_1 \quad p_2 \quad k_3] \quad (2.38)$$

Donde k_1 , k_2 y k_3 representan las distorsiones radiales, y p_1 , p_2 representan las distorsiones tangenciales. Las distancias focales y los coeficientes de distorsión son obtenidos mediante calibración utilizando OpenCV [26].

Además de la calibración de la cámara, se deben agregar los parámetros extrínsecos que corresponden a las matrices de rotación y traslación respecto al eje de coordenadas con el que se desea referenciar la posición del dirigible. Esto se realiza mediante una transformación homogénea. El modelo de la cámara será el siguiente:

$$Z_c \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \mathbf{r} & \mathbf{t} \\ \mathbf{0} & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \quad (2.39)$$

Donde \mathbf{r} representa la matriz de rotación y \mathbf{t} representa el vector de traslación.

Mediante el procesamiento de imágenes y la calibración de la cámara, se puede obtener la posición y orientación del marcador respecto a la cámara, obteniendo como resultado la transformada homogénea T_{ar}^c . Posteriormente, mediante la inversa de esta transformada homogénea T_{ar}^c , se puede obtener la posición y orientación de la cámara respecto al marcador, es decir, $T^{ar}c$. A su vez, esta transformada se puede convertir a un eje de coordenadas fijo en la habitación mediante la transformada $T^{ar}\eta$, como se muestra en la figura 2.10.

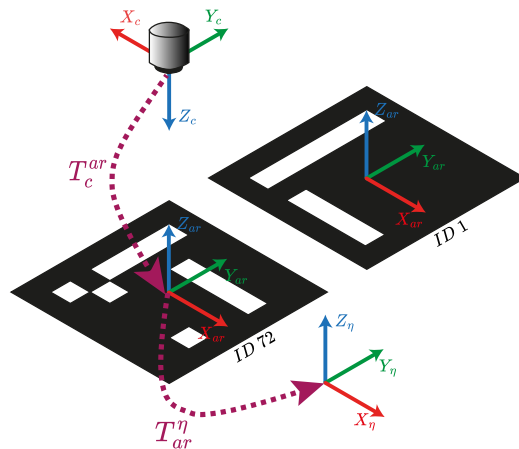


Figura 2.10: Transformación de ejes coordenados mediante transformadas homogéneas.

CAPÍTULO 3

Componentes del dirigible

La estructura de la góndola se debe construir considerando el tamaño de los sensores y el tipo de motores a utilizar, además del peso máximo que puede levantar el globo del dirigible. Como en este proyecto se desea controlar la posición en tres dimensiones y el ángulo de guiñada, se debe determinar los valores de desplazamiento vertical (distancia recorrida en eje Z), desplazamiento horizontal (distancia recorrida en ejes X e Y), y orientación respecto al eje Z. Los sensores y la cantidad de motores a utilizar se determinaron de acuerdo a los tipos de movimiento a controlar. A continuación se detallan cada uno de los componentes del dirigible y el motivo por el cual se escogió cada uno de ellos.

3.1 Globo

Se escogió el globo considerando que el dirigible debe trabajar en interiores, y a la vez debe levantar la mayor cantidad de carga posible. Se utilizó como referencia el trabajo realizado por [14], donde se desarrolló un dirigible para interiores con un globo de perfil simétrico respecto a cada eje. El hecho de que el perfil sea simétrico respecto a cada eje simplifica la dinámica del dirigible.

En el comercio se ofrece principalmente globos de latex y foil (nombre comercial). El globo debe mantener su forma al ser inflado de manera que no cambien sus propiedades de masa añadida y coeficiente de amortiguamiento, por lo que se descarta inmediatamente el latex. Por lo anterior se seleccionó un globo de foil redondo de 91 cm de diámetro, que levanta aproximadamente 100 gramos (adicionales a su propio peso) cuando está completamente inflado con helio.

3.2 Transmisión y recepción de datos

Las opciones para transmisión y recepción de datos disponibles para esta investigación son Bluetooth, WiFi, y módulos con protocolo Zigbee. En la investigación realizada por [27] se comparan los tres métodos de comunicación mencionados, concluyendo que el Bluetooth es el sistema mas adecuado para dispositivos portátiles (sensores corporales, aplicaciones en vehículos, etc.) por su corto alcance. La transmisión vía WiFi es la más adecuada para aplicaciones en hogares inteligentes, controlando dispositivos con protocolo TCP/IP incorporado. Los dispositivos con protocolo Zigbee serían los mas adecuados para aplicaciones de automatización industrial y robótica por su gran alcance y un bajo consumo energético.

Como se menciona en [28] la comunicación WiFi es la que presenta mayor consumo de energía por lo que queda descartada, ya que una baja tasa de consumo es primordial para obtener un

mayor tiempo de vuelo. Por otro lado el mayor rango de operación lo presenta el protocolo Zigbee con un mínimo de 100 metros versus 10 metros del Bluetooth. Por estos motivos se implementará el protocolo Zigbee.

Se utilizan dos módulos un XBee-S2C, los cuales se comunican mediante el el protocolo Zigbee. Estos módulos trabajan a una frecuencia de 2.4GHz con una velocidad máxima de transmisión de datos de 250.000 b/s, y un rango de señal de 1200 metros [29].

Comunicación UART

La comunicación se realiza utilizando el modo transparente, en el cual el módulo Xbee actúa como una línea de comunicación serial, específicamente utilizando comunicación UART (Universal Asynchronous Receiver Transmitter). En la transmisión UART se debe configurar la velocidad de transmisión de datos en baudios (bit por segundo), bit de inicio y parada, la longitud de los datos (8 bits), y se puede agregar opcionalmente bits de paridad que sirven para detectar errores en la transmisión de datos [30]. Dicha estructura de transmisión se muestra en la figura 3.1.

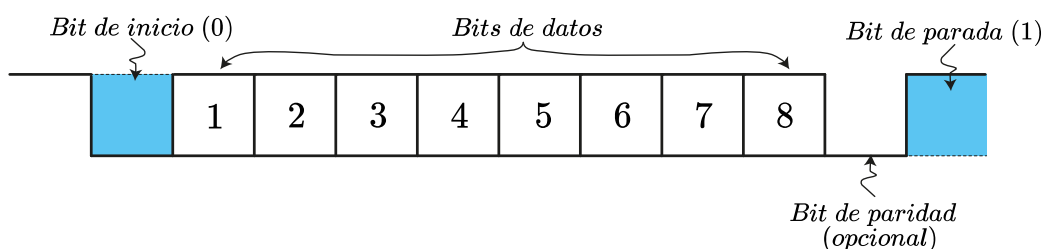


Figura 3.1: Estructura de comunicación serial UART [30].

Los parámetros de transmisión de datos del del modulo Xbee se muestran en la tabla 3.1.

Tabla 3.1: Parámetros de transmisión de sistema de transmisión de datos.

Parámetro	Configuración
Baudios	19200
Formato	ASCII
Tipo de dato	Flotante (%f)
Tiempo de muestreo	0.2 seg.
Paridad	impar

Para la velocidad de transmisión (Baudios) se utilizó el valor mas alto logrado sin que la transmisión presente problemas. Esto se determinó experimentalmente realizando pruebas enviando y recibiendo una trama con 4 numéricos mas sus separadores, considerando un tiempo de muestreo fijo. El tiempo de muestreo especificado es el que se utilizó en el control MPC. La paridad se establece como impar lo que significa que el total de bits transmitidos será impar (incluido el bit de paridad), de no ser así es porque se generó un error en la transmisión.

3.3 Tarjeta programable

En el mercado existe una gran variedad de tarjetas programables en las cuales se podría generar el código necesario para llevar a cabo el control MPC. Para la selección de la tarjeta se debe considerar su tamaño y peso, la compatibilidad con los sensores a utilizar y el lenguaje en el que será programada.

En esta investigación se utilizó microcontrolador Arduino, modelo Fio. Este microcontrolador incluye un módulo para la conexión de transmisores Xbee, lo que facilita las conexiones para la transmisión de datos. Es una tarjeta de un tamaño aceptable para esta aplicación (64X27 mm aproximadamente), con un peso de 45 gramos.

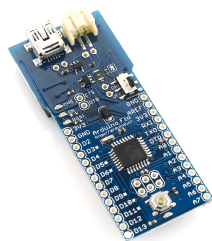


Figura 3.2: Arduino Fio [31].

La tarjeta Arduino fue utilizada para la aplicación del control de altura del dirigible. Sin embargo, para el control de orientación y desplazamiento fue insuficiente, ya que se requiere realizar procesamiento de imágenes para detectar marcadores, lo cual difícilmente se podría realizar con las capacidades que posee una Arduino Fio. Por este motivo se considera la utilización de un microprocesador Raspberry Pi 4. El cual tiene un procesador más poderoso, y puede ser programado mediante Python o C++ pudiendo incluir librerías de procesamiento de imágenes como OpenCV. Este microprocesador tiene dimensiones de 85X49 mm y un peso de 59 grs.

3.4 Sensores y actuadores

3.4.1 Medición de altura

Para la medir altura (posición en eje z), se debió seleccionar entre dos opciones disponibles, las cuales fueron un sensor ultrasonido y uno infrarrojo. El principio de funcionamiento de ambos sensores es el mismo, y consiste en emitir una onda (luz o sonido) y medir el tiempo que tarda esa onda en ser reflejada en algún objeto que se encuentre frente al sensor como se muestra en la figura 3.3.

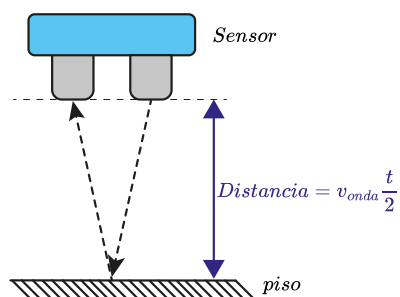


Figura 3.3: Medición de altura.

El sensor infrarrojo es no-lineal y existen sensores con rangos de operación acotados. Por este motivo se decidió utilizar un sensor ultrasonido modelo HC-SR04, el cual tiene un rango de operación de 2 a 400 cm, una precisión de 0.5 cm y un peso de 8 gramos de acuerdo al proveedor "MCI electronics".

3.4.2 Medición de orientación

Para determinar la orientación en las pruebas de identificación de sistemas, se consideró un sensor de medición inercial o IMU (por sus siglas en inglés). El más accesible de estos sensores es el MPU6050 el cual cuenta con acelerómetro en sus 3 ejes, y giroscopio también para sus 3 ejes, como se puede apreciar en la figura 3.4. Este sensor puede presentar problemas para determinar rotación en el eje z . Esto debido a que la determinación de la rotación proviene de la integración de la velocidad medida por el giroscopio, por lo que si hay un pequeño error en la calibración este se propagará en el tiempo. Dicho error puede ser corregido mediante la incorporación de la medición del acelerómetro. Por otro lado el acelerómetro también podría presentar desviaciones respecto al valor real de giro, debido a altas vibraciones en el sistema en el cual está midiendo. Con esto en consideración se decidió utilizar este sensor debido a que internamente el sensor cuenta con un procesamiento de señales llamado Digital Motion Processor o DMP, el cual limpia las señales y mitiga los efectos negativos de ambas mediciones (acelerómetro y giroscopio). En [32] se determinó que el error al medir orientación respecto al eje z o ángulo de guiñada es de aproximadamente 1.8 grados.

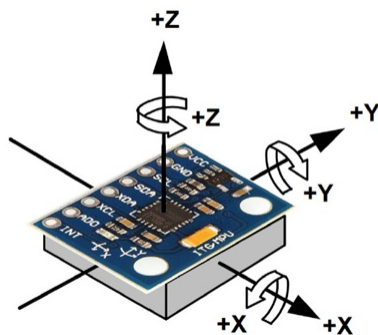


Figura 3.4: Sensor MPU 6050 con sus 3 ejes de medición.

Otra alternativa sería incorporar un magnetómetro, para corregir la orientación midiendo el norte magnético de la tierra.

3.4.3 Cámara utilizada para localización

De acuerdo a lo expresado en el capítulo 2.6 la localización en ejes (x, y) y el ángulo de guiñada (rotación en z) se realizará mediante procesamiento de imágenes, por lo que se requiere una cámara compatible con la Raspberry Pi 4. Existen cámaras oficiales para Raspberry Pi, y considerando que no se necesita una cámara con un gran aumento o lentes que puedan distorsionar la imagen (como un gran angular) y que también sea lo mas pequeña posible, se seleccionó la Raspberry Pi V2.1. Esta cámara tiene 8 Megapíxeles de resolución, y puede grabar entre 60 y 90 fps con una resolución de 640X480 píxeles. Sus dimensiones son 25X23X9mm y pesa 3 grs [33].



Figura 3.5: Cámara Raspberry Pi V2.1 [33].

3.4.4 Motores y Puente H

Motores

Para controlar el posicionamiento del dirigible es necesario poder invertir la dirección de giro de los motores es por esto que se optó por motores de corriente continua sin núcleo. Es necesario que los motores DC giren en sentido opuesto para anular entre sí el momento generado por los mismos motores sobre el dirigible. Se determinó la utilización de motores DC de minidrone, específicamente motores de un minidrone Parrot mambo, cuyo voltaje de alimentación es 3.7V, y sus dimensiones son 8.5X20mm.

Puente H

Para regular tanto la la magnitud como el sentido de aplicación del voltaje que se envía a los motores, se utiliza un Puente H que recibe señales de pulso modulado o PWM (por sus siglas en ingles). Dado que los motores DC a utilizar operan con un voltaje de 3.7V, se seleccionó un puente H modelo TB6612FNG. Este puente H tiene un voltaje de alimentación de 2.7 a 5.5V con una corriente máxima de 1.2A y un peso de 5 gramos. Debido a que se deben controlar 4 motores, es necesario utilizar 2 puentes H.

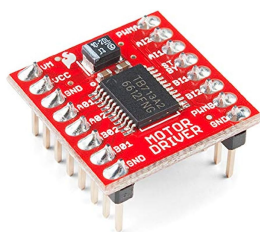


Figura 3.6: Puente H TB6612FNG.

3.5 Góndola

La góndola es la estructura que contiene todos los componentes electrónicos del dirigible, tales como sensores, motores, la batería y el microcontrolador. La estructura debe ser lo más liviana posible; por eso, fue diseñada para ser impresa en 3D y en partes separadas. De otra forma, se generarían demasiados soportes en el proceso de impresión, y al tener pilares tan delgados en su estructura, resultaría en una impresión de mala calidad. En la figura 3.7, se muestra la góndola con las ubicaciones designadas para los motores y el sensor de ultrasonido HC-SR04. La versión a) es la que se modeló inicialmente para trabajar solo con la tarjeta Arduino Fio. Posteriormente, este modelo fue modificado y adaptado para poder incorporar una Raspberry PI, como se puede observar en la versión b). En esta versión, también se puede utilizar una cámara, o en su lugar, un sensor HC-SR04, ya que es impresa por partes y la cavidad se puede reemplazar.

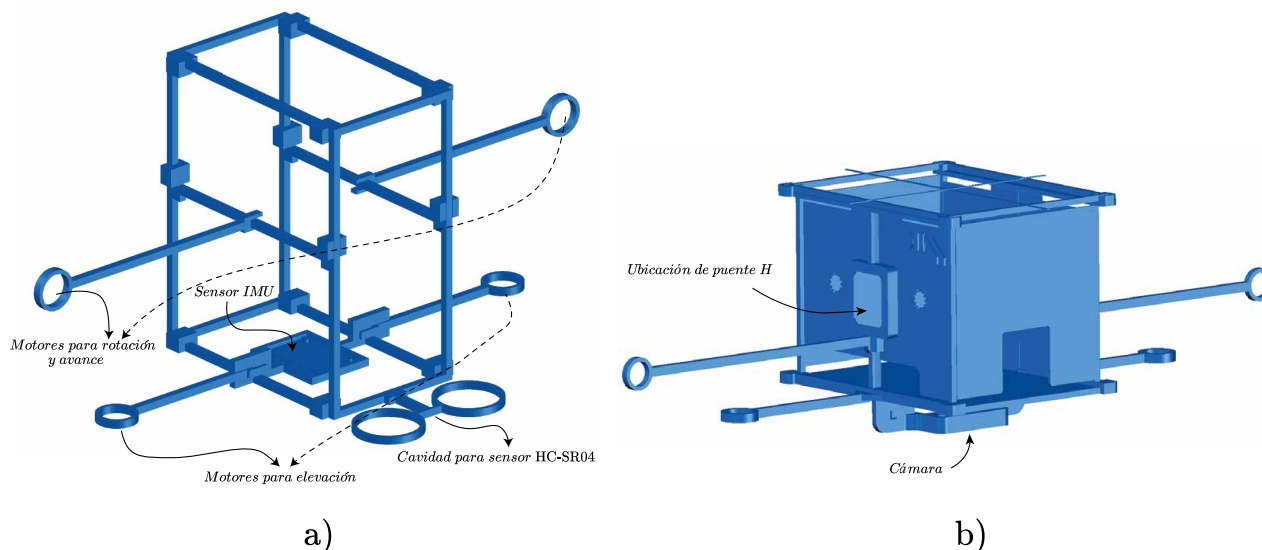


Figura 3.7: Góndola diseñada para el dirigible a) Prototipo inicial para Arduino Fio, b) prototipo mejorado para Raspberry Pi 4.

La impresión se realiza en PLA. Con una impresora Creality, modelo Ender3 Pro. La boquilla utilizada tiene un diámetro de 1.75 mm, y se utilizó una expansión horizontal de -0.2 para los las piezas en las que deben ir los motores, el ultrasonido y el sensor IMU.

En la figura 3.8 se muestra el dirigible armado, con cada uno de sus componentes.

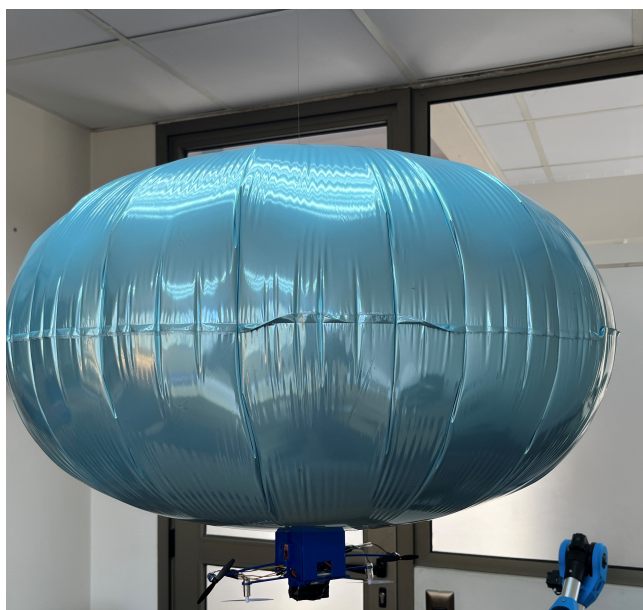


Figura 3.8: Dirigible con todos sus componentes.

3.5.1 Masa total de componentes

La masa de los componentes con los que se cuenta para armar controlar el dirigible se entregan en la tabla 3.2:

Tabla 3.2: Masa de los componentes del dirigible.

Componente	Masa (grs)
góndola	22.8
Arduino Fio+Xbee	7.2
Puente H	2.4
Batería de 3.7V y 400 mah	11
Raspberry Pi+camara	42.5
4 Motores dc de drone Parrot mambo	23.6 (5.9x4)
Sensor MPU6050	1.3

El globo adquirido tiene la capacidad de levantar 99.6 grs. (medido experimentalmente) cuando se encuentra completamente inflado con helio. En este trabajo se evaluó la posibilidad de trabajar con la Arduino Fio para implementar un sistema de control para la orientación (θ) y la altura (eje z), y posteriormente evaluar el control de desplazamiento mediante el control de orientación y velocidad, realizando la localización del dirigible mediante marcadores Aruco en una Raspberry Pi. Sin embargo, el peso total de los componentes para trabajar con la Raspberry Pi es de 115 gramos, sin contar los cables y el elevador de voltaje necesario, lo cual supera en 15 gramos la masa máxima que el dirigible puede levantar. Esto se considera teniendo en cuenta que el peso original de la Raspberry es de 59 gramos y se eliminó el puerto Ethernet y los puertos USB para reducir su masa.

También se intentó realizar la localización con una Raspberry Pi Zero, que pesa solo 9 gramos, pero la latencia fue de 0.5 segundos solo programando la localización mediante marcadores Aruco. Esto se realizó utilizando la librería en Python y tal vez se podría mejorar utilizando la librería de C++.

Por estos motivos, sólo se implementó el sistema de control para la orientación y la altura del dirigible utilizando el microcontrolador Arduino Fio.

CAPÍTULO 4

Evaluaciones experimentales

4.0.1 Fuerza de empuje de motores

La fuerza de empuje generada por las hélices de los motores presenta diferencias al cambiar el sentido de giro. Esto puede deberse al cambio en la eficiencia de las hélices (tienen un sentido de giro para el que fueron diseñadas) y la eficiencia propia de los motores en cada sentido de giro. Por este motivo se debe determinar la relación entre el voltaje (magnitud y sentido de aplicación) y la fuerza generada por los motores. Mediante el puente H se puede regular el sentido de giro, y también el voltaje aplicado a través de la siguiente relación:

$$V_{Motor} = V_{Bateria} \cdot C_{trabajo} \quad (4.1)$$

Donde $V_{bateria}$ es el voltaje entregado al puente H, y $C_{trabajo}$ es el ciclo de trabajo que se entrega al puente H mediante una señal de pulso modulado o PWM (por sus siglas en inglés).

Al igual que en el trabajo realizado en [34], se determinó la relación entre la fuerza de empuje ejercida por los motores y el voltaje entregado mediante el puente H utilizando una Balanza de alta precisión "Precisa 500M-2000C", para lo cual se imprimió una estructura para probar la fuerza de los motores, como se observa en la figura 4.1.



Figura 4.1: Balanza para medición de fuerza generada por motores DC.

El esquema utilizado para medir la fuerza generada en el eje z es el siguiente:

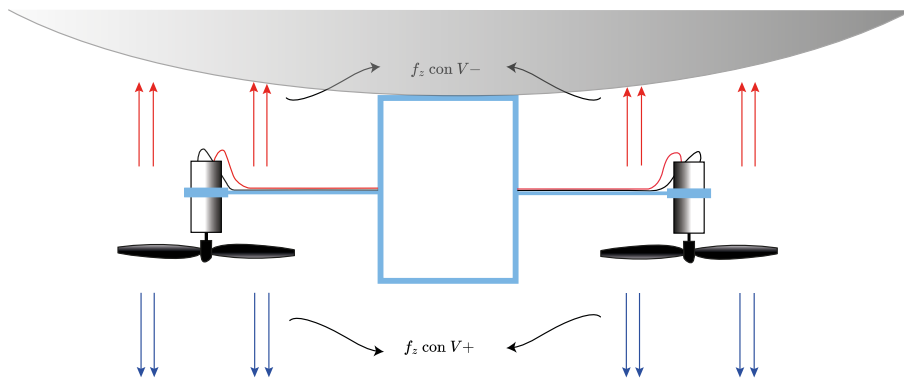


Figura 4.2: Esquema de medición de fuerza en motores en eje z.

En la figura 4.2 se muestra un esquema con las direcciones de aplicación de voltaje V y la fuerza generada para el eje z , con la cual se controlará la altitud del dirigible. A partir de las mediciones de fuerza obtenidas, se obtuvo el siguiente gráfico de fuerza (f_z) versus voltaje (V).

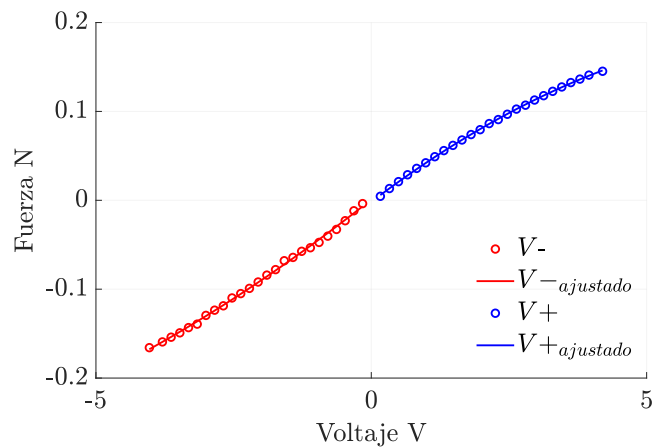


Figura 4.3: Fuerza de empuje vs voltaje aplicado a motores.

A través de las mediciones que se muestran en la figura 4.3 se determina la relación que existe entre el voltaje aplicado a los motores y la fuerza de empuje generada. Donde $V+$ y $V-$ corresponde a la dirección de aplicación del voltaje. Las curvas fueron ajustadas de acuerdo a su forma, con un polinomio de segundo grado. De esta manera, la fuerza generada por los motores y el voltaje tendrán la siguiente relación:

$$f_z = \begin{cases} -0.002806V^2 + 0.04699V - 0.001767 & \text{para } V+ \\ 0.001798V^2 + 0.04904V + 0.000849 & \text{para } V- \end{cases} \quad (4.2)$$

Para determinar la fuerza que ejercen los motores que controlaran la rotación y el desplazamiento del dirigible se siguió el mismo procedimiento.

4.0.2 Identificación de parámetros

Modelo de desplazamiento vertical y horizontal

Para aplicar el control predictivo basado en modelo, es necesario determinar los parámetros m_{xy} , m_z , D_{xy} , y D_z de los movimientos primitivos de nuestro modelo.

Considerando un movimiento desacoplado en eje xy y en eje z (un movimiento no afecta al otro), se puede representar el movimiento para cada eje a través de la siguiente expresión:

$$f_{xy} = m_{xy} \cdot \dot{v}_{xy} + D_{xy} \cdot v_{xy} \quad (4.3)$$

$$f_z = m_z \cdot \dot{v}_z + D_z \cdot v_z \quad (4.4)$$

Ambas ecuaciones tienen la misma estructura. La función de transferencia en el dominio de Laplace de las ecuaciones 4.3 y 4.4 considerando condiciones iniciales de posición y velocidad igual a cero se pueden representar de la siguiente forma:

$$\frac{Z_i}{F_i} = \frac{1}{(m_{T,i}) \cdot s^2 + (D_i) \cdot s} \quad (4.5)$$

Donde $m_{T,i}$, $i \in \{xy, z\}$ corresponde a la masa total del sistema para cada eje. D_i , $i \in \{xy, z\}$ es el coeficiente de amortiguamiento para cada eje.

Resolviendo la transformada de Laplace de la ecuación 4.5, para una entrada constante $F_i = f_0/s$ y condiciones iniciales cero, se obtiene que la ecuación de movimiento es:

$$Z_i(t) = \left(-\frac{m_{T,i}}{(D_i)^2} + \frac{t}{D_i} + \frac{m_{T,i}}{(D_i)^2} \cdot e^{\frac{-D_i}{m_{T,i}} \cdot t} \right) \cdot f_0 \quad (4.6)$$

Para determinar los parámetros del sistema, se desarrollaron ensayos en los cuales se mide la respuesta del dirigible ante una fuerza constante. El primer paso es determinar el peso para el cual se anula la fuerza de flotabilidad con la fuerza de gravedad. A partir esta condición se agrega una masa de 1.08 gramos al dirigible generando que este se desplace hacia abajo considerando su posición inicial como el cero (invirtiendo el eje z para simplificar la ecuación) como se muestra en la figura 4.4, midiendo la posición y el tiempo desde que el dirigible comienza a descender (partiendo con condiciones iniciales cero).

Para asegurar que las condiciones iniciales sean cero se utilizó un servomotor que libera el dirigible en el momento exacto en el que comienza el muestreo de datos, sincronizando el dirigible con el circuito del servomotor con un pulsador común que inicia la cuenta regresiva.

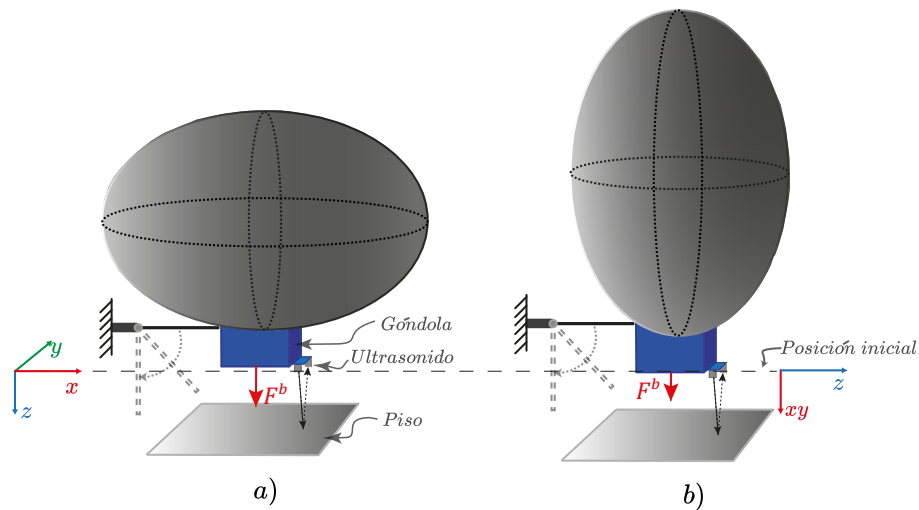


Figura 4.4: Experimento para determinar constantes de amortiguamiento y masa añadida a) ensayo para constantes m_z y D_z , b) ensayo para constantes m_{xy} y D_{xy} .

El ensayo se realiza 13 veces en cada orientación para generar regresiones que ajusten los parámetros de la ecuación 4.6 y posteriormente se realiza un ensayo con una masa agregada de 4.1 grs. para validar el modelo con los parámetros obtenidos.

Para lograr una tasa de muestreo más alta que la obtenida con el transmisor Xbee, las mediciones se almacenaron directamente en una tarjeta micro SD utilizando un módulo conectado a la tarjeta Arduino, con un tiempo de muestreo de 0.01 segundos.

Modelo para la rotación

Para determinar las variables de inercia I_z y el amortiguamiento D_θ , se realiza un ensayo en el cual se fija al dirigible en el eje vertical (eje z) mediante hilos de nylon de 0.05 mm de diámetro, para evitar desplazamientos y cualquier tipo de perturbación, como se puede observar en el esquema de la figura 4.5. En este ensayo se entregan fuerzas iguales a los motores, pero en direcciones opuestas, de forma que se genere un par de fuerzas. La distancia entre ejes de los motores es de 0.261m.

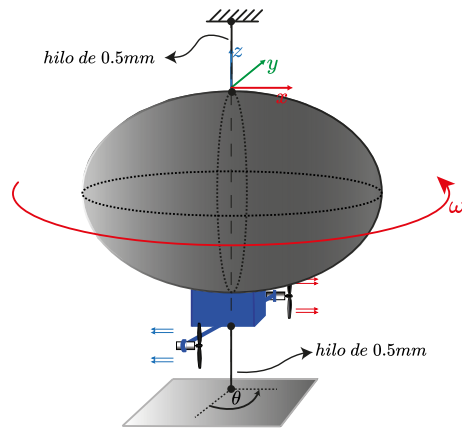


Figura 4.5: Esquema de experimento para determinar inercia respecto I_z y amortiguamiento D_θ del dirigible.

Se determina la ecuación de movimiento para la rotación el función del tiempo, resolviendo la ecuación diferencial 2.3. Realizando el procedimiento análogo al realizado para las ecuaciones de desplazamiento. La solución es la siguiente:

$$\theta(t) = \left(-\frac{I_z}{(D_\theta)^2} + \frac{t}{D_\theta} + \frac{I_z}{(D_\theta)^2} \cdot e^{\frac{-D_\theta}{I_z} \cdot t} \right) \cdot M_0 \quad (4.7)$$

Donde M_0 es el momento en Nm con el que se realizó la prueba, el cual fue de 0.0014 Nm. Al igual que para los desplazamientos, se realizaron 13 pruebas.

Resultados de identificación de parámetros

En la tabla 4.1 se entregan los parámetros obtenidos a partir de una regresión no lineal realizada a través de Matlab mediante el método de Levenberg-Marquard. Se seleccionó este método tomando como referencia el trabajo realizado en [35].

Tabla 4.1: Parámetros identificados experimentalmente.

Parámetro	Valor Promedio	Desviación Estándar
m_z	0.1896 kg	0.00117
D_z	0.0429 Ns/m	0.00357
m_{xy}	0.2288 kg	0.065219
D_{xy}	0.03945 Ns/m	0.002355
I_z	0.003648 kgm ²	0.00041315
D_θ	0.000457 Ns/m	0.0001632

A partir de los parámetros obtenidos se compara el modelo con la respuesta ante la fuerza ejercida por una masa de 4.1 gramos añadidos de lastre. Se realiza esta prueba en las dos

orientaciones para el desplazamiento, y también se realiza la prueba para determinar la ecuación de movimiento de rotación del dirigible en función del momento que se aplica. El resultado se muestra en los siguientes gráficos:

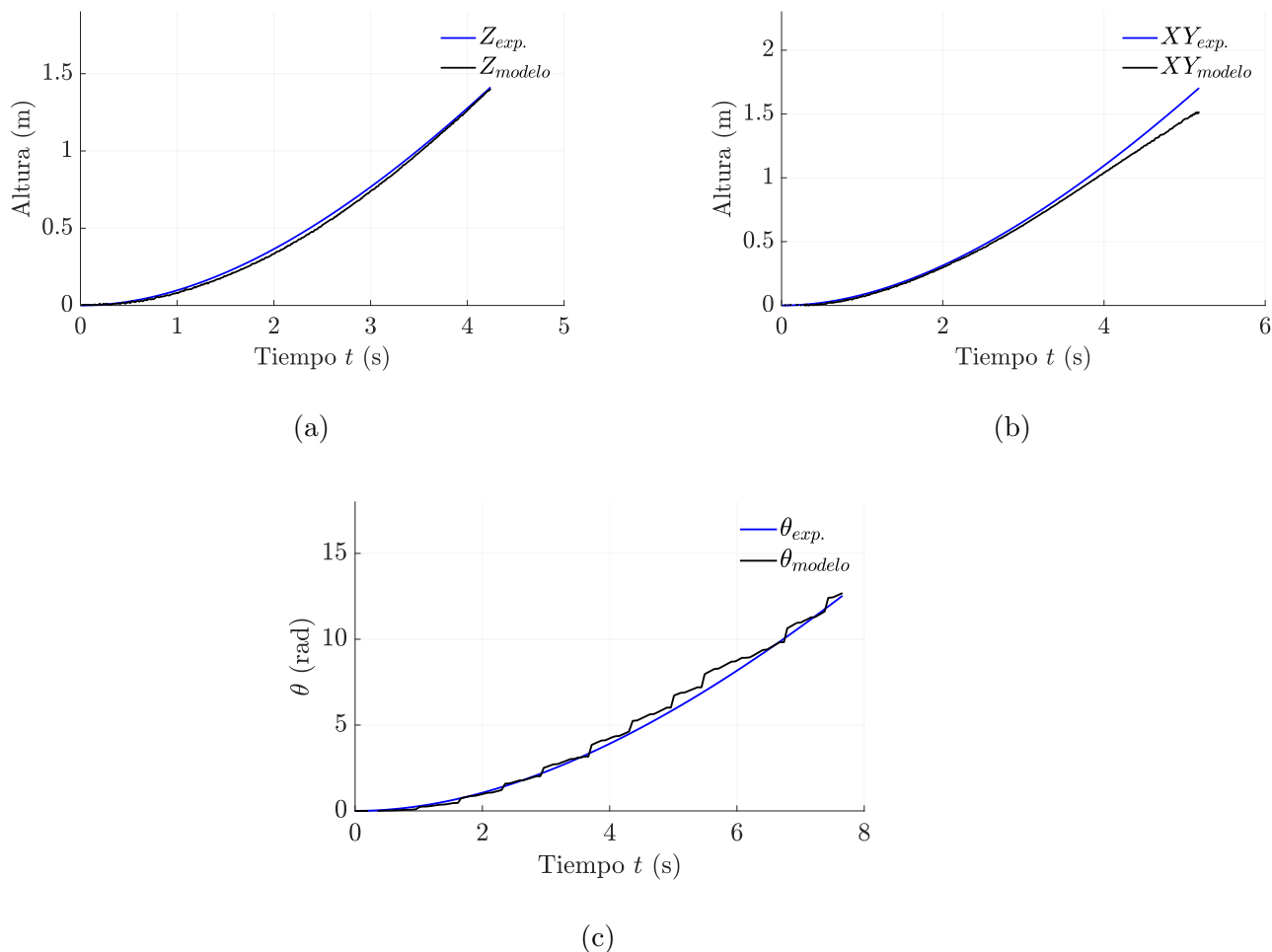


Figura 4.6: a) Validación de modelo para desplazamiento en eje z . b) Validación de modelo para desplazamiento horizontal en ejes xy . c) Validación de modelo para rotación respecto al eje z .

A través de Matlab se determina el Error Medio Cuadrático Normalizado o NRMS . Obteniendo que en dirección vertical z el modelo tiene una similitud del 94.72%, para el modelo de desplazamiento horizontal xy la similitud es del 87.75% y para el modelo de rotación respecto al eje z se obtuvo una similitud de 91.05% . Se considera que el porcentaje de similitud es lo suficientemente cercano para ser utilizado en el sistema de control.

De acuerdo a la ecuación 4.8, la expresión en espacio de estados para el modelo que representa la altura del dirigible, es la siguiente:

$$\begin{pmatrix} \dot{x}_z \\ \ddot{x}_z \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 0 & -0.2262 \end{pmatrix} \begin{pmatrix} x_z \\ \dot{x}_z \end{pmatrix} + \begin{pmatrix} 0 \\ 5.2742 \end{pmatrix} (f_z) \quad (4.8)$$

La ecuación en espacio de estados para la orientación, reemplazando los parámetros obtenidos será la siguiente:

$$\begin{pmatrix} \dot{\theta} \\ \ddot{\theta} \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 0 & -0.1253 \end{pmatrix} \begin{pmatrix} \theta \\ \dot{\theta} \end{pmatrix} + \begin{pmatrix} 0 \\ 274.1 \end{pmatrix} (M_z) \quad (4.9)$$

Las expresiones 4.8 y 4.9 son discretizadas en Matlab a partir de la función "c2d" utilizando un tiempo de muestreo de 0.2 segundos (mismo que el utilizado en el MPC) para posteriormente ser transformadas a su forma aumentada, como se explicó en el capítulo 2.4.

Utilizando las funciones "ctrb(A,B)" y "obsv(A,C)" de Matlab, se determinaron las matrices de controlabilidad y observabilidad correspondientes a las ecuaciones 4.8 y 4.9. Se comprobó que, en ambos casos, las matrices presentan rango completo, lo cual indica que las dos salidas son observables y la entrada es controlable en ambas situaciones.

4.1 Aplicación de MPC

4.1.1 Parámetros del filtro de Kalman

Para implementar el control predictivo, es necesario ajustar el Filtro de Kalman, ya que este proporciona la realimentación de los estados. Los parámetros ajustados para la medición de la altura se muestran en la tabla 4.2.

Tabla 4.2: Especificaciones de Filtro de Kalman para estimación de z y \dot{z} .

Parámetro	Valor
\mathbf{Q}	$\begin{bmatrix} 0.0005 & 0 \\ 0 & 0.00005 \end{bmatrix}$
v	0.000025
P_0	$\begin{bmatrix} 0.00001 & 0 \\ 0 & 0.00001 \end{bmatrix}$
\mathbf{C}	$\begin{bmatrix} 1 & 0 \end{bmatrix}$
$\hat{\mathbf{x}}_0$	$\begin{bmatrix} x_z(1) & 0 \end{bmatrix}$

El valor del ruido en la medición (v) se determinó a partir del error en las mediciones del sensor ultrasonido que es de 0.005 m, por lo que la varianza en las mediciones es de $0.000025 m^2$. Por otro lado el dirigible es sensible a cualquier corriente de aire por mas mínima que sea, además la reacción de los motores no es inmediata (tiene un retraso) por lo que no se puede confiar un 100% en las estimaciones realizadas a través del modelo. Por esto se agregó la matriz \mathbf{Q} y sus parámetros fueron ajustados considerando las magnitudes de las mediciones, de manera que el filtro de Kalman predice principalmente la velocidad, y para la posición se ponderan mayormente las mediciones del sensor considerando . De esta forma el valor \mathbf{Q} para las mediciones de posición es de $0.0005m^2$ (error por perturbaciones de aprox. 2 cm) y para la velocidad es de $0.00005(m/seg)^2$, o un error por ruido del proceso de aprox. $0.007m/seg$.

Considerando la velocidad máxima que alcanza el dirigible es de aproximadamente $0.6m/seg$ se considera un error de predicción del modelo un 1.1% para esta variable.

Para el valor de P_0 se utilizó un error inicial en la covarianza de $0.0001 m^2$ en la posición y $0.00001 m^2/s^2$ en la velocidad, sin embargo de acuerdo a [19] el valor de P_0 es insignificante y se podría iniciar como una matriz identidad. El valor inicial de la predicción de estados $\hat{\mathbf{x}}_0$ se considera como la posición medida inicialmente, con una velocidad cero.

Para la orientación, se utilizó un valor de v de 0.00098. Este se determina a partir del error en la medición de la orientación, que es de aproximadamente 1.8 grados, según [32] y [36]. Estos estudios demostraron que el error no debería superar los dos grados para períodos de tiempo inferiores a un minuto. Esto corresponde a 0.0314 radianes y significa que el error tendrá una varianza de ≈ 0.00098 . Se ajustaron los valores de \mathbf{Q} de manera que se corrige principalmente la velocidad a través del modelo, asumiendo un error de predicción del modelo de 0.01 rad/seg. Este error es mínimo, considerando que la velocidad máxima alcanzada es de aproximadamente 1.2 rad/seg. Para la orientación, se consideró un error debido al ruido en el proceso de 0.031 radianes, o 1.8 grados.

Al igual que para la altura se consideraron valores de P_0 pequeños y aleatorios ya que este valor se autocorrigue de acuerdo a las predicciones. Los parámetros utilizados en el filtro de Kalman para las mediciones del sensor MPU6050 se entregan en la tabla 4.3.

Tabla 4.3: Especificaciones de Filtro de Kalman para estimación de θ y $\dot{\theta}$.

Parámetro	Valor
\mathbf{Q}	$\begin{bmatrix} 0.001 & 0 \\ 0 & 0.0001 \end{bmatrix}$
v	0.00098
P_0	$\begin{bmatrix} 0.00001 & 0 \\ 0 & 0.00001 \end{bmatrix}$
\mathbf{C}	$\begin{bmatrix} 1 & 0 \end{bmatrix}$
$\hat{\mathbf{x}}_0$	$\begin{bmatrix} \theta(1) & 0 \end{bmatrix}$

4.1.2 Aplicación de MPC para control de altura

Se determinaron los horizontes de control N_c y de predicción N_p a partir de las indicaciones entregadas en [37], las cuales son:

$$N_p > \frac{t_{respuesta}}{t_s} \quad (4.10)$$

$$N_c \ll N_p$$

Como se explicó en el capítulo 2.3 se comparará el control MPC con el control PID diseñado por [17]. Considerando que las pruebas experimentales para el MPC se realizan desde una posición inicial de 8.7 cm, se simuló la función de transferencia 2.7 como se muestra en la figura 6.2 de los Anexos. De acuerdo a esto se espera que el control MPC tenga un tiempo de levantamiento

de 1.56 segundos y un sobrepaso máximo de un 7%. El tiempo de muestreo seleccionado es de 0.2 segundos, y se espera un tiempo de asentamiento de aproximadamente 5 segundos. Con estos parámetros en mente, se procedió a seleccionar los valores de N_p y N_c .

Los parámetros de ponderación en la función de costo del MPC fueron seleccionados de acuerdo a simulaciones y posteriormente ajustados de forma experimental, considerando la respuesta con menor sobrepaso respecto a la referencia de posición. En la tabla 4.5 se muestran todos los parámetros con los que se aplicó el control MPC.

Tabla 4.4: Especificaciones de MPC para control de posición en eje z.

Parámetro	valor
N_p	30
N_c	10
Tiempo de muestreo t_s	0.2 seg
Ponderación en entrada w	2000
Ponderación en posición q_1	1
Ponderación en velocidad q_2	1

En la figura 4.7 se muestra el resultado obtenido luego de ajustar la ponderación w de forma experimental. Para lograrlo se implementó el control MPC para regular la altura del dirigible, estableciendo una referencia a un metro de altura desde el suelo y referencia de velocidad de 0 m/seg , sin considerar ninguna restricción. La ponderación w se configuró de manera que la fuerza en los motores no excediera los 0.18 N, que es la fuerza máxima que los motores de corriente continua utilizados pueden proporcionar. Como resultado, se logró un tiempo de levantamiento (desde el 10% hasta el 90% del valor final) de 1.9 segundos, un sobrepaso del 11%, y un tiempo de asentamiento de 10.8 segundos.

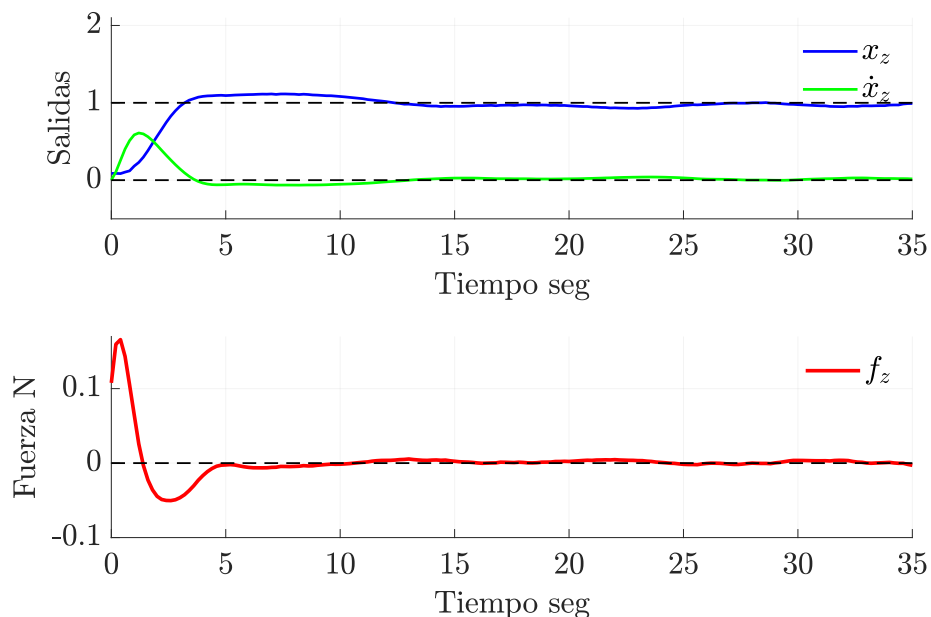


Figura 4.7: Control de altura z mediante control MPC sin restricciones.

Comparación con PID

En la comparación del control de la altura del dirigible, se observa que la respuesta del control PID es más rápida que la experimental, con un tiempo de levantamiento 0.6 segundos más corto. También se presenta un sobrepaso que es un 3% mayor en el control MPC y un tiempo de asentamiento que es 5 segundos mayor. Ambos parámetros podrían explicarse por la fuerza que el aire ejerce sobre el globo del dirigible en el movimiento descendente, la cual no se consideró al determinar la fuerza que generan los motores. Para corregir esto, se debería calcular la fuerza de los motores con el dirigible instalado en la estructura de prueba.

La aplicación experimental muestra un mayor tiempo muerto al comienzo, lo cual también contribuye a la diferencia en la respuesta. Aun así, esta diferencia es pequeña y se justifica adicionalmente por el hecho de que la fuerza de entrada en la simulación del PID es casi el triple de la que se aplica en el control MPC.

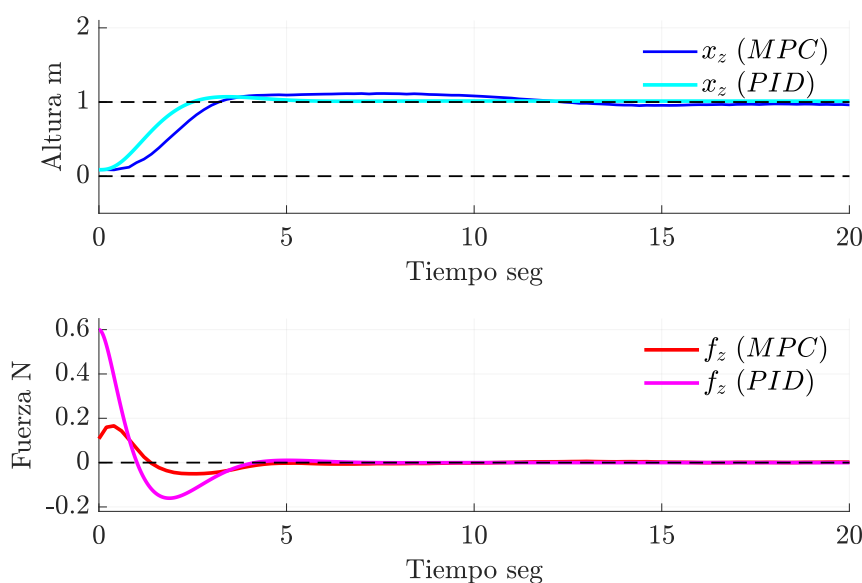


Figura 4.8: Comparación entre control predictivo basado en modelo y control PID, en control de altura z .

Aplicación de MPC para control de altura, con perturbaciones

Se puso a prueba el sistema de control mediante la introducción de perturbaciones, empujando el dirigible hacia arriba y hacia abajo. Los resultados obtenidos demostraron la eficacia del sistema de control para contrarrestar dichas perturbaciones, logrando retornar a la posición de referencia en un tiempo menor a 8 segundos en cada uno de los casos. Los resultados se pueden apreciar en la figura 4.9.

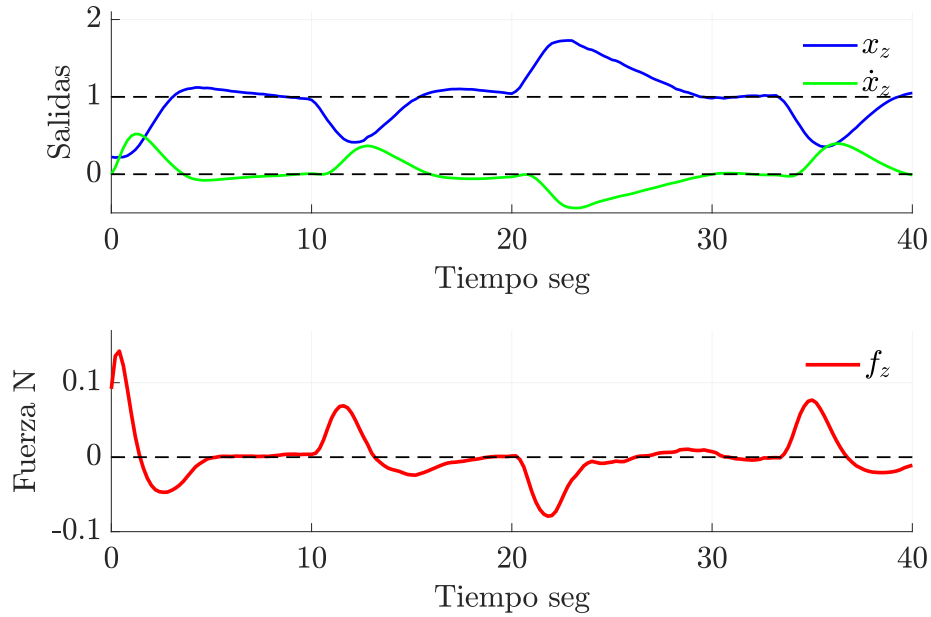


Figura 4.9: Aplicación de MPC en control de altura, sometiendo el dirigible a perturbaciones.

Como resultado de la primera perturbación, el dirigible descendió a una posición de 0.4 m y tardó aproximadamente 3 segundos en regresar a su posición original. Sin embargo, tras la segunda perturbación, el dirigible se elevó a 1.7 m y tardó aproximadamente 7 segundos en regresar a su punto de referencia, lo cual es más del doble del tiempo que tardó en la primera perturbación. Este patrón se observó en todas las pruebas realizadas, lo que podría atribuirse a que la fuerza generada por los motores fue calculada sin tener en cuenta la fuerza que se ejerce sobre el globo del dirigible cuando los motores lo impulsan hacia abajo (como se observa en la figura 4.2). Esta también podría explicar el retraso en el tiempo de asentamiento que se observa en la figura 4.7, ya que la fuerza real de empuje hacia abajo sería menor a la calculada por el MPC.

Aplicación de Restricciones en la fuerza f_z

Se incorporaron restricciones en la entrada del dirigible, específicamente, en la fuerza máxima que los motores pueden generar, limitando así el voltaje que puede aplicarse a estos, dado que están directamente correlacionados. Se estableció la limitación para el rango de fuerza de $-0.08N < f_z < 0.08N$, lo que dio como resultado la respuesta mostrada en la figura 4.10

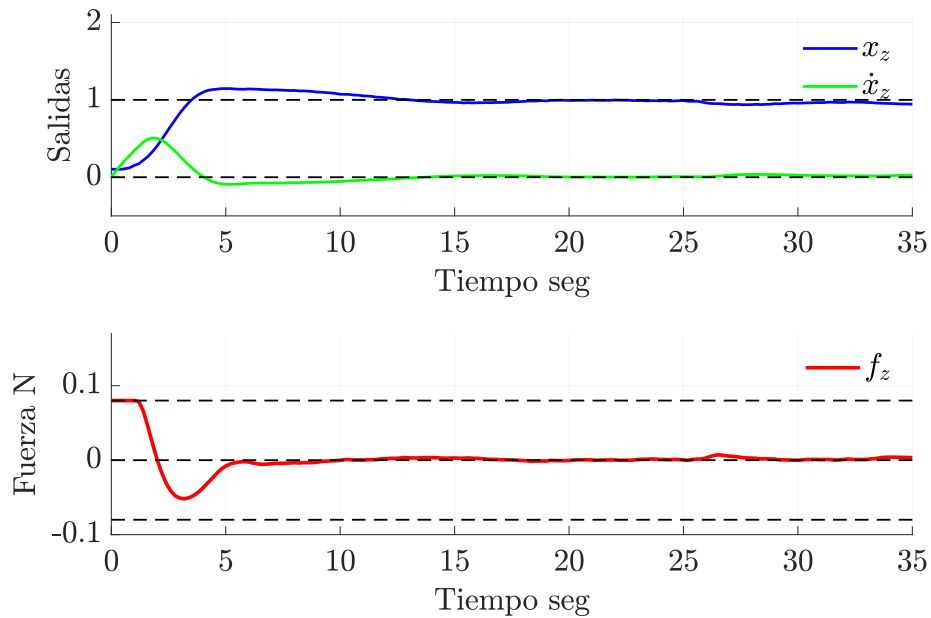


Figura 4.10: Aplicación de MPC para control de altura, con restricciones en entrada.

En la figura 4.10 se puede apreciar que se cumple la restricción impuesta a la señal de entrada, alcanzando el límite superior de la fuerza definido en las restricciones iniciales. Como consecuencia, el sobrepaso se incrementa a un 14%, y el tiempo de asentamiento se amplía a 11 segundos. Aunque este resultado representa un aumento en ambos parámetros, sigue siendo aceptable y consistente con las expectativas, dado que los posibles valores para la entrada de la señal de control están siendo restringidos, por lo que se obtiene un resultado subóptimo desde el punto de vista de la optimización de la función de costo.

4.1.3 Aplicación de MPC para control de Orientación

Para determinar la ponderación w en la señal de control del MPC de la orientación, se simuló el control PID de la función de transferencia 2.8 en Simulink. Esto permitió obtener parámetros de comparación respecto a los tiempos de respuesta que debería presentar el control de orientación.

La orientación de referencia utilizada en el MPC fue de 1.57 radianes (90 grados), y una velocidad de 0 rad/seg. Se consideró una posición angular inicial de 0.135 radianes en la simulación, que corresponde a la posición inicial de la prueba experimental. El diagrama de bloques de Simulink se muestra en la figura 6.2 en los Anexos. De esta simulación, se determinó que la respuesta del sistema debería tener un tiempo de levantamiento de 1.36 segundos y un tiempo de asentamiento de 2.05 segundos. Los valores de N_p y N_c , al igual que para la altura, se determinaron considerando que el tiempo de respuesta del sistema podría variar entre 0 y 5 segundos

Como resultado, la mejor respuesta obtenida con el MPC mostró un tiempo de levantamiento de 1.7 segundos y un tiempo de asentamiento de 2.8 segundos, utilizando una ponderación w de 5000. La respuesta del dirigible se puede observar en la figura 4.11. Con este valor de

ponderación, el valor máximo del momento obtenido fue de 0.015Nm, que fue convertido a fuerza para los motores teniendo en cuenta que la distancia entre estos es de 261.9 mm.

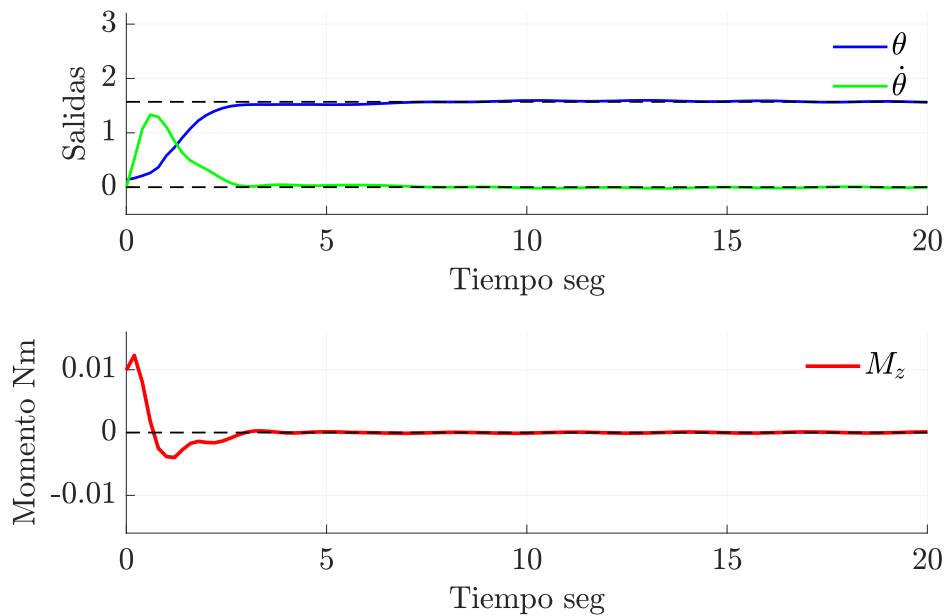


Figura 4.11: Aplicación de MPC en orientación sin restricciones en entrada.

El resumen de los parámetros utilizados en la evaluación experimental del sistema de control se entrega en la tabla 4.5.

Tabla 4.5: Especificaciones de MPC para control de orientación θ .

Parámetro	valor
N_p	30
N_c	10
Tiempo de muestreo t_s	0.2 seg
Ponderación en entrada w	5000
Ponderación en posición q_1	1
Ponderación en velocidad q_2	1

Comparación con PID

El tiempo de levantamiento del PID resultó ser 0.34 segundos más rápido que el control MPC, y 0.75 segundos más rápido en el tiempo de asentamiento. Esta diferencia se considera insignificante y se explica principalmente porque el momento generado por los motores al inicio de la simulación del control PID es de 0.016 Nm, y en la aplicación del MPC es de 0.0099 Nm, es decir, el momento del PID simulado es 1.6 veces mayor al inicio, y luego de 0.27 segundos los momentos son iguales.

Se puede observar en la figura 4.12 que el control PID tiene un tiempo de levantamiento menor al del MPC sin embargo requiere de un mayor momento para lograr dicha rotación.

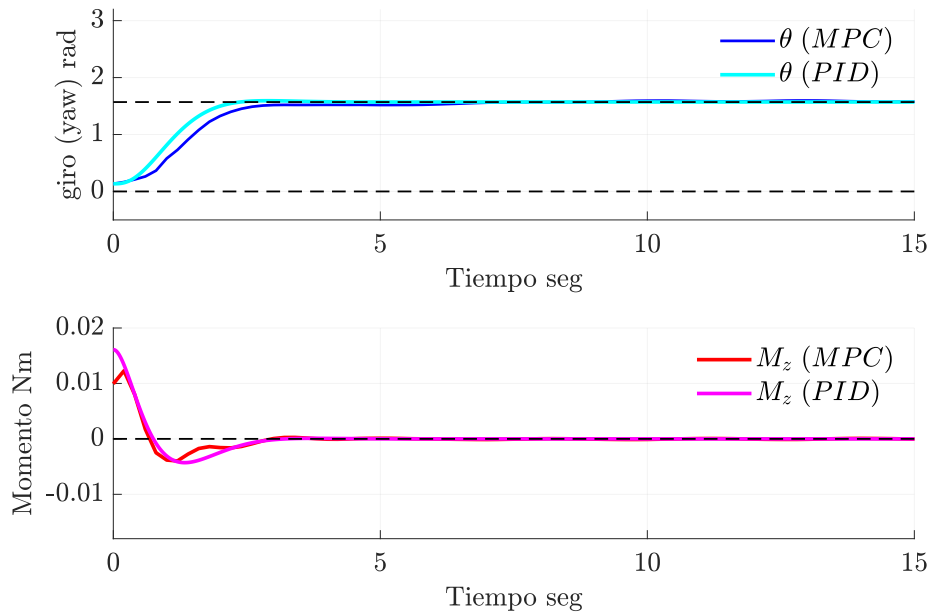


Figura 4.12: Comparación entre MPC y control PID aplicado a control de orientación θ .

Aplicación de MPC para control de orientación, con perturbaciones

Se aplicaron perturbaciones al dirigible para evaluar sus tiempos de respuesta, cuyos resultados se pueden apreciar en la figura 4.13. Durante la primera perturbación, el dirigible giró hasta los 0.44 radianes, lo que supone una desviación de 1.13 radianes respecto a la referencia (1.57 radianes o 90°), y demoró 3.2 segundos en regresar a la misma. Durante la segunda perturbación, el dirigible giró hasta los 2.41 radianes, implicando una desviación de 0.84 radianes respecto a la referencia, y tardó 3 segundos en retornar a ella. A partir de estos resultados, se puede inferir que los tiempos de respuesta son aproximadamente iguales en ambas direcciones de movimiento.

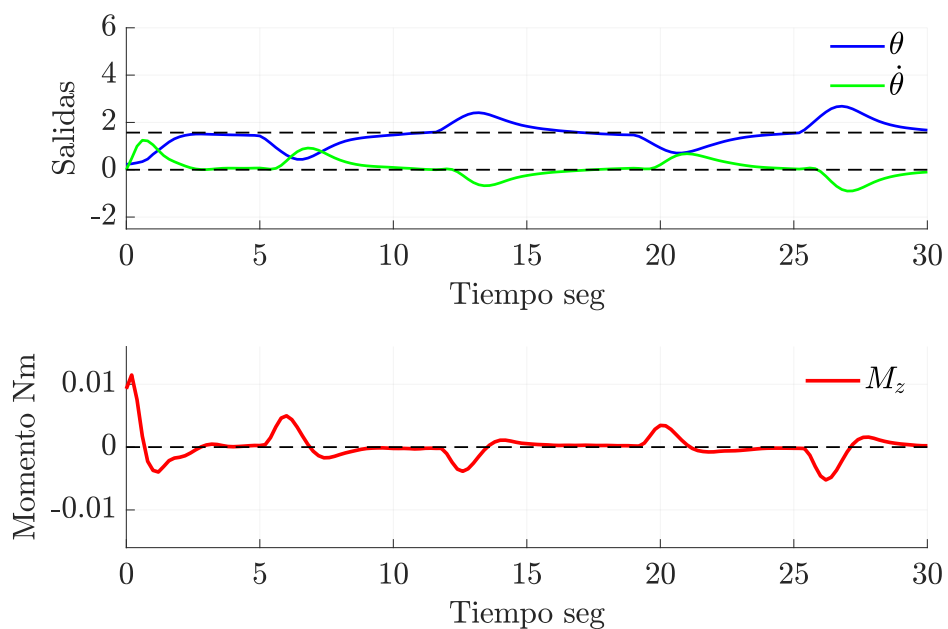


Figura 4.13: Aplicación de MPC en orientación sin restricciones en entrada con perturbaciones.

Aplicación de MPC para control de orientación con restricciones en momento M_z

Se agregaron restricciones al momento máximo que se puede entregar al dirigible para comprobar la capacidad del MPC de controlar la orientación del mismo, limitando la fuerza que los motores pueden proporcionar. La restricción impuesta fue de 0.0032 Nm. Los resultados de esta prueba se muestran en la figura 4.14.

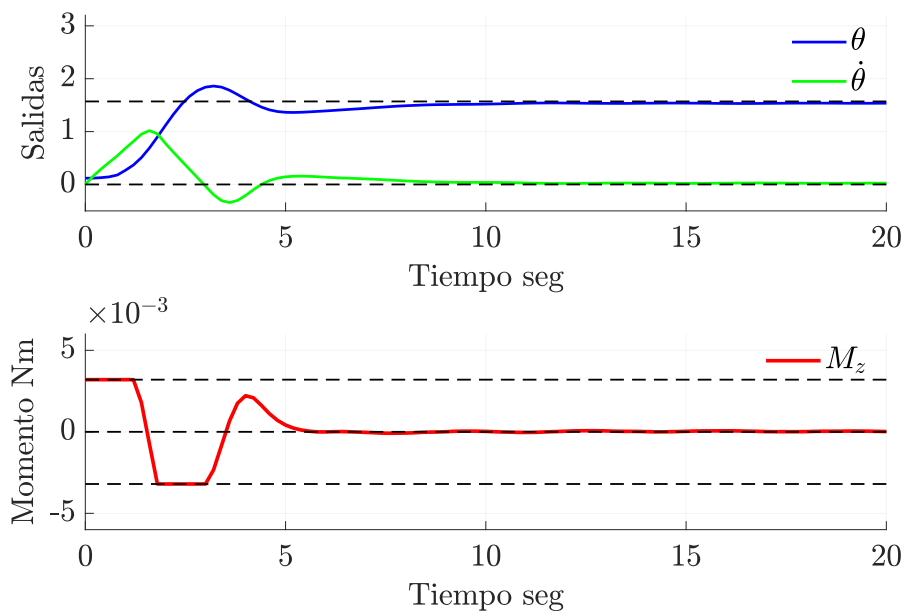


Figura 4.14: Aplicación de MPC para controlar orientación aplicando restricciones al momento que generan los motores

Al aplicar la restricción, se obtuvo una respuesta con un sobrepaso del 20% y el tiempo de asentamiento se incrementó a 8.4 segundos, lo cual es tres veces mayor que el tiempo que tarda sin restricciones. Para alcanzar la referencia, el sistema de control envía constantemente el límite superior del momento que se puede aplicar, seguido por el límite inferior. Dado que la solución del MPC (Model Predictive Control) no es óptima a nivel global debido a las restricciones impuestas, se obtiene una respuesta con un desempeño inferior, aunque sigue siendo un buen tiempo de respuesta

CAPÍTULO 5

Conclusiones y perspectivas a futuro

En cuanto al diseño, se logró diseñar un dirigible, al cual se puede aplicar control de altura y orientación utilizando componentes de bajo costo y una conexión inalámbrica para el envío y la recepción de señales necesarias para el lazo de control. Sin embargo, no se logró controlar el desplazamiento horizontal, ya que al agregar la Raspberry PI, se superó el límite de masa de aproximadamente 100 grs. que puede levantar el globo utilizado.

Mediante la identificación de sistemas, utilizando el método de caja gris, se determinaron los parámetros de masa, amortiguamiento e inercia necesarios para los tres movimientos primitivos del dirigible. Los modelos para el control de altura y orientación mostraron una similitud del 94.7% y 91.05% respectivamente, con la señal de prueba utilizada para la validación. Estos resultados se consideran suficientes, dadas las características de los sensores, y teniendo en cuenta que el dirigible es sensible a perturbaciones como corrientes de aire. Dicho modelo se aplicó tanto en el filtro de Kalman como en el control MPC.

A partir de los resultados del capítulo 4.1, se confirma que el control MPC podría ser una excelente alternativa al control PID. Aunque las simulaciones del control PID mostraron tiempos de respuesta levemente superiores, esto se debe principalmente a que en el control PID se tiene una señal de control más alta al inicio. Tal señal, aunque favorable en teoría, podría no ser factible de aplicar en la realidad, como en el caso del control de altura, debido a limitaciones en la fuerza máxima que pueden generar los motores.

El control MPC permite controlar el dirigible en condiciones en las que la entrada o la fuerza que pueden generar los motores esté limitada o se desee limitar. Esto quedó demostrado al aplicar restricciones a la fuerza en el control de altura z , y en el control de orientación θ . Esta capacidad representa una ventaja con respecto al control PID, ya que a pesar de que la respuesta fue subóptima con las restricciones en el MPC, en el control PID no se pueden considerar este tipo de restricciones en el proceso de cálculo de la señal de entrada.

Como perspectivas a futuro, se podría intentar la aplicación del control horizontal del dirigible utilizando una Raspberry PI Zero, la cual solo pesa 9 gramos, y emplear programación en C++ para mejorar el tiempo de ejecución del programa. De lograr esto, se abrirían posibilidades para aplicar métodos más avanzados, como el procesamiento de imágenes y redes neuronales para control y localización del dirigible.

REFERENCIAS

- [1] Yiwei Liu, Zengxi Pan, David Stirling, and Fazel Naghdy. Control of autonomous airship. In *2009 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pages 2457–2462. IEEE, 2009.
 - [2] Hong Zhang and James P Ostrowski. Visual servoing with dynamics: Control of an unmanned blimp. In *Proceedings 1999 IEEE International Conference on Robotics and Automation (Cat. No. 99CH36288C)*, volume 1, pages 618–623. IEEE, 1999.
 - [3] Anthony Colozza and James L Dolce. High-altitude, long-endurance airships for coastal surveillance. *NASA Technical Report, NASA/TM-2005-213427*, 2005.
 - [4] David St-Onge, Clément Gosselin, and Nicolas Reeves. Dynamic modelling and control of a cubic flying blimp using external motion capture. *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, 229(10):970–982, 2015.
 - [5] Sjoerd van der Zwaan, Alexandre Bernardino, and Jose Santos-Victor. Vision based station keeping and docking for an aerial blimp. In *Proceedings. 2000 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2000)(Cat. No. 00CH37113)*, volume 1, pages 614–619. IEEE, 2000.
 - [6] Michael Burri, L Gasser, M Käch, Matthias Krebs, S Laube, A Ledergerber, Daniel Meier, R Michaud, Lukas Mosimann, L Müri, et al. Design and control of a spherical omnidirectional blimp. In *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1873–1879. IEEE, 2013.
 - [7] Flavio Araripe d’Oliveira, Francisco Cristovão Lourenço de Melo, and Tessaleno Campos Devezas. High-altitude platforms—present situation and technology trends. *Journal of Aerospace Technology and Management*, 8:249–262, 2016.
 - [8] Fernando Ulloa Vásquez. Sistemas de radiocomunicación basados en plataformas aéreas eléctricas no tripuladas.
 - [9] Ningshi Yao, Emily Anaya, Qiuyang Tao, Sungjin Cho, Hongrui Zheng, and Fumin Zhang. Monocular vision-based human following on miniature robotic blimp. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3244–3249. IEEE, 2017.
 - [10] Ahmad Alsayed. *Pitch and Altitude Control of an Unmanned Airship with Sliding Gondola*. PhD thesis, Université d’Ottawa/University of Ottawa, 2017.
 - [11] Axel Rottmann, Christian Plagemann, Peter Hilgers, and Wolfram Burgard. Autonomous blimp control using model-free reinforcement learning in a continuous state and action space. In *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1895–1900. IEEE, 2007.
-

- [12] Hiroaki Fukushima, Ryosuke Saito, Fumitoshi Matsuno, Yasushi Hada, Kuniaki Kawabata, and Hajime Asama. Model predictive control of an autonomous blimp with input and output constraints. In *2006 IEEE Conference on Computer Aided Control System Design, 2006 IEEE International Conference on Control Applications, 2006 IEEE International Symposium on Intelligent Control*, pages 2184–2189. IEEE, 2006.
- [13] Liuping Wang. *Model predictive control system design and implementation using MATLAB®*. Springer Science & Business Media, 2009.
- [14] Sungjin Cho, Vivek Mishra, Qiuyang Tao, Paul Vamell, Matt King-Smith, Aneri Muni, Weston Smallwood, and Fumin Zhang. Autopilot design for a class of miniature autonomous blimps. In *2017 IEEE Conference on Control Technology and Applications (CCTA)*, pages 841–846. IEEE, 2017.
- [15] Lennart Ljung. Approaches to identification of nonlinear systems. In *Proceedings of the 29th Chinese Control Conference*, pages 1–5. IEEE, 2010.
- [16] Víctor M Alfaro Ruíz. Ecuaciones para controladores pid universales. *Ingeniería. Revista de la Universidad de Costa Rica*, 12(1-2):11–20, 2002.
- [17] Qiuyang Tao. *Design and control of an indoor miniature autonomous blimp*. PhD thesis, Georgia Institute of Technology, 2020.
- [18] Eduardo F Camacho and Carlos Bordons Alba. Control predictivo: Pasado, presente y futuro. *Revista Iberoamericana de Automática e Informática Industrial*, 1 (3), 5-28., 2004.
- [19] Matthew B Rhudy, Roger A Salguero, and Keaton Holappa. A kalman filtering tutorial for undergraduate students. *International Journal of Computer Science & Engineering Survey*, 8(1):1–9, 2017.
- [20] Anca Morar, Alin Moldoveanu, Irina Mocanu, Florica Moldoveanu, Ion Emilian Radoi, Victor Asavei, Alexandru Gradinaru, and Alex Butean. A comprehensive survey of indoor localization methods based on computer vision. *Sensors*, 20(9):2641, 2020.
- [21] Dragan Stojanović and Natalija Stojanović. Indoor localization and tracking: Methods, technologies and research challenges. *Facta Universitatis, Series: Automatic Control and Robotics*, 13(1):57–72, 2014.
- [22] Yutao Wang, Zongpeng Zheng, Zhiqi Su, Gang Yang, Zhong Wang, and Yu Luo. An improved aruco marker for monocular vision ranging. In *2020 Chinese Control And Decision Conference (CCDC)*, pages 2915–2919. IEEE, 2020.
- [23] Hisao Kadota, Hidenori Kawamura, Masahito Yamamoto, Toshihiko Takaya, and Azuraa Ohuch. Vision-based positioning system for indoor blimp robot. *IFAC Proceedings Volumes*, 37(8):478–483, 2004.
- [24] Andrej Babinec, Ladislav Jurišica, Peter Hubinský, and František Duchoň. Visual localization of mobile robot using artificial markers. *Procedia Engineering*, 96:1–9, 2014.
- [25] Huijuan Zhang, Chengning Zhang, Wei Yang, and Chin-Yin Chen. Localization and navigation using qr code for mobile robot in indoor environment. In *2015 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pages 2501–2506. IEEE, 2015.

- [26] OpenCVDevTeam. Camera calibration. https://docs.opencv.org/3.0-beta/doc/py_tutorials/py_calib3d/py_calibration/py_calibration.html#, November 2014.
- [27] Olaide O Kazeem, Olubiyi O Akintade, and Lawrence O Kehinde. Comparative study of communication interfaces for sensors and actuators in the cloud of internet of things. *Int. J. Internet Things*, 6(1):9–13, 2017.
- [28] Salim Jibrin Danbatta and Asaf Varol. Comparison of zigbee, z-wave, wi-fi, and bluetooth wireless technologies used in home automation. In *2019 7th International Symposium on Digital Forensics and Security (ISDFS)*, pages 1–5. IEEE, 2019.
- [29] IJorquera. Xbee zb s2c th. <https://xbee.cl/xbee-zb-s2c-th/>, August 2022.
- [30] Yi-yuan Fang and Xue-jun Chen. Design and simulation of uart serial communication module based on vhdl. In *2011 3rd International Workshop on Intelligent Systems and Applications*, pages 1–4. IEEE, 2011.
- [31] Arduino Fio kernel description. <https://xbee.cl/arduino-fio/>. Accessed: 09-11-2021.
- [32] Arif Ainur Rafiq, Wahid Nur Rohman, and Sugeng Dwi Riyanto. Development of a simple and low-cost smartphone gimbal with mpu-6050 sensor. *Journal of Robotics and Control (JRC)*, 1(4):136–140, 2020.
- [33] Raspberry Pi documentation. Camera. <https://www.raspberrypi.com/documentation/accessories/camera.html>, 2012-2022.
- [34] Jinjun Shan. Dynamic modeling and vision-based control for indoor airship. In *2009 International Conference on Mechatronics and Automation*, pages 2934–2939. IEEE, 2009.
- [35] Yue Wang, Gang Zheng, Denis Efimov, and Wilfrid Perruquetti. Altitude control for an indoor blimp robot. *IFAC-PapersOnLine*, 50(1):15990–15995, 2017.
- [36] Seung Yun Song, Yinan Pei, and Elizabeth T Hsiao-Wecksler. Estimating relative angles using two inertial measurement units without magnetometers. *IEEE Sensors Journal*, 22(20):19688–19699, 2022.
- [37] Matlab documentation. Choose sample time and horizons. <https://la.mathworks.com/help/mpc/ug/choosing-sample-time-and-horizons.html>, 2022.

CAPÍTULO 6

Anexos

6.1 Diagramas de simulaciones en Simulink

Las figuras 6.2 y 6.1 muestran las simulaciones realizadas para los sistemas de control de orientación y altura, respectivamente. En ambos casos, se añadió un desfase a la señal de salida, ya que las pruebas del MPC no inician en la posición cero. Por ello, se sumó una orientación de 0.135 radianes para la orientación y 0.087 metros para la altura.

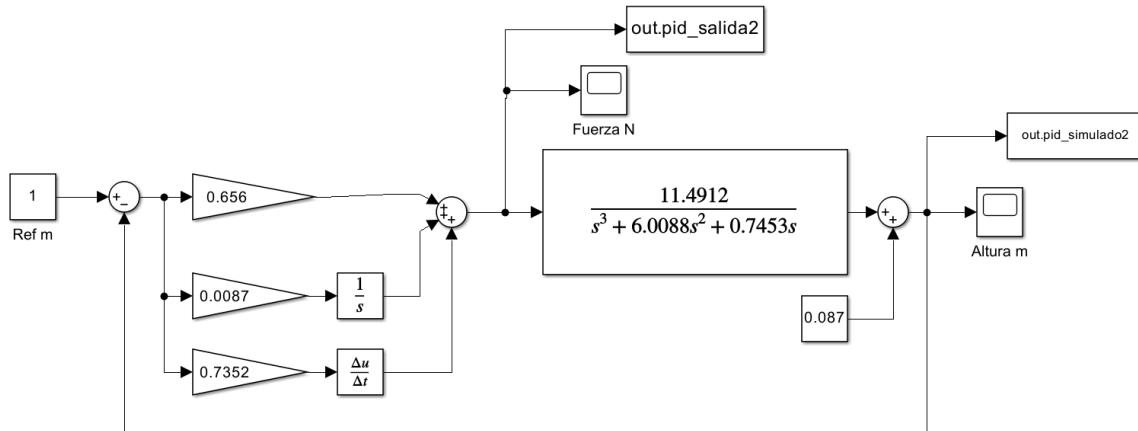


Figura 6.1: Diagrama de simulación en simulink para control PID de altura.

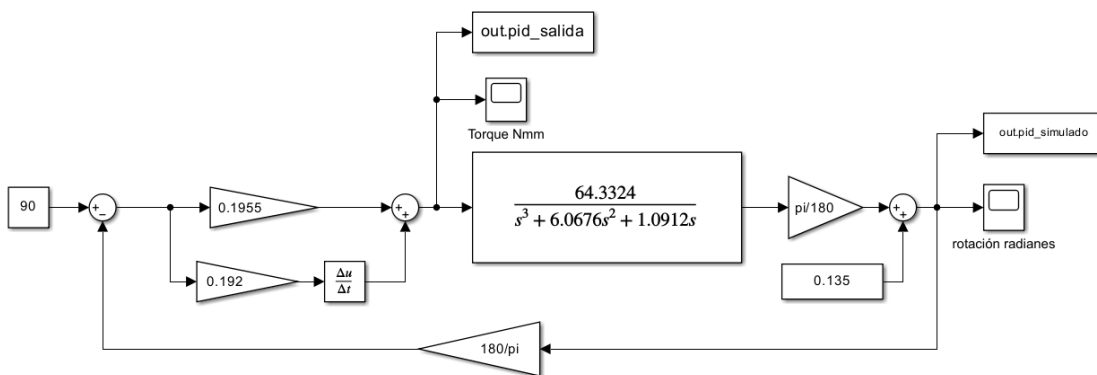


Figura 6.2: Diagrama de simulación en simulink para control PID de orientación.

6.1.1 Matriz de calibración de cámara

A pesar de no poder aplicar la localización mediante procesamiento de imágenes y marcadores Aruco, se entrega la calibración de la cámara utilizada, ya que podría ser de utilidad si se desea utilizar una Raspberry Pi Zero programada mediante C++.

La calibración de la cámara se lleva a cabo mediante el método del tablero de ajedrez utilizando OpenCV [26] con 24 capturas del tablero en diferentes ángulos y distancias. Obteniendo los siguientes parámetros :

$$\text{matriz de calibración} = \begin{bmatrix} 486.7031 & 0 & 324.3133 \\ 0 & 488.07867 & 242.5989 \\ 0 & 0 & 1 \end{bmatrix} \quad (6.1)$$

$$\text{Coeficientes de distorsiones} = [0.1115464 \quad -0.109831 \quad 0.004877 \quad 0.0032 \quad -0.3746] \quad (6.2)$$