



UNIVERSIDAD AUTÓNOMA
DEL ESTADO DE HIDALGO



INSTITUTO DE CIENCIAS BÁSICAS E INGENIERÍA
CENTRO DE INVESTIGACIÓN EN TECNOLOGÍAS DE LA INFORMACIÓN Y SISTEMAS

INSTITUTO DE CIENCIAS BÁSICAS E INGENIERÍA
MAESTRÍA EN CIENCIAS EN AUTOMATIZACIÓN Y CONTROL
CONTROL INTEGRAL DE UN UAV DE ALA ROTATORIA

T E S I S
QUE PARA OBTENER EL GRADO DE MAESTRO EN CIENCIAS
EN AUTOMATIZACIÓN Y CONTROL

PRESENTA:
GILDARDO GODÍNEZ GARRIDO

ASESORES:
DR. HUGO ROMERO TREJO
DR. OMAR JACOBO SANTOS SÁNCHEZ

PACHUCA HGO., MÉXICO ENERO DE 2016



UNIVERSIDAD AUTÓNOMA DEL ESTADO DE HIDALGO
Instituto de Ciencias Básicas e Ingeniería
Institute of Basic Sciences and Engineering
Área Académica de Computación y Electrónica
Computer Science and Electronics Department

Mineral de la Reforma, Hgo., a 13 de enero de 2016
Oficio No. MCAC01_2016

Ing. Gildardo Godínez Garrido
P R E S E N T E

Por medio de la presente y en mi calidad de coordinador de la Maestría en Ciencias en Automatización y Control, del Área Académica de Computación y Electrónica (AACyE) de la Universidad Autónoma del Estado de Hidalgo (UAEH), me es grato informarle que el Jurado asignado para la revisión de su trabajo de tesis titulado: "**Control Integral de un UAV de Ala Rotatoria**", dirigido por el Dr. Hugo Romero Trejo, y el Dr. Omar Jacobo Santos Sánchez que para obtener el grado de Maestro en Ciencias en Automatización y Control fue presentado por usted, ha tenido a bien en reunión de sinodales, autorizarlo para impresión. A continuación se integran las firmas de conformidad de los integrantes del Jurado:

Dr. Omar Arturo Domínguez Ramírez	(Presidente)	AACyE-UAEH
Dr. Jesús Patricio Ordaz Oliver	(Secretario)	AACyE-UAEH
Dr. Hugo Romero Trejo	(Vocal)	AACyE-UAEH
Dr. Omar Jacobo Santos Sánchez	(Suplente 1)	AACyE-UAEH

Atentamente
"Amor, Orden y Progreso"

Dr. Omar Jacobo Santos Sánchez
Coordinador de la Maestría en Ciencias en Automatización y Control
Universidad Autónoma del Estado de Hidalgo

c.c.p. Dr. Orlando Ávila Pozos, Director del Instituto de Ciencias Básicas e Ingeniería
c.c.p. Dr. Hugo Romero Trejo, Jefe del Área Académica de Computación y Electrónica
c.c.p. Expediente/ apl



Ciudad del Conocimiento
Carretera Pachuca - Tulancingo km. 4.5
Colonia Carboneras
Mineral de la Reforma, Hidalgo, México, C.P. 42184
Tel. +52 771 7172000 exts. 2250 y 2251
omarj@uaeh.edu.mx

www.uaeh.edu.mx



Gildardo Godínez Garrido

Control Integral de un UAV de Ala Rotatoria

Es muy hermosa la teoría, pero la práctica es un gran despertar, que nos lleva a soñar con nuevas teorías...

Agradecimientos

Doy gracias a Dios que me ha permitido culminar con este trabajo el cual no ha sido fácil, pero gracias a él contemplo lo complejo de la creación y me siento más pequeño, pero a la vez más grande por pertenecer y forma parte de ella.

A mis Padres Ines Garrido Ibarra y Arnulfo Godínez Cabrera, por ser las dos personas que más me han cuidado, enseñado y apoyado a lo largo de toda mi vida, por ser los mejores padres que me pudieron tocar.

A la mujer más especial en mi vida Griselda Monterrubio Balderrama, muchas gracias por todo tu apoyo por hacerme tan feliz y soportar conmigo retos como éste, te amo.

A mis hermanos Amalia y José Arnulfo por su apoyo, comprensión y sobre todo por ser tan buenos hermanos, los quiero mucho.

A mis compañeros por enseñarme tantas cosas por ayudarme y ser mis aliados en este reto tan grande.

Agradezco a mis asesores de tesis Dr. Omar Jacobo Santos Sánchez y Dr. Hugo Romero Trejo, por su paciencia, por compartir su conocimiento, por ayudarme incondicionalmente, por darme esta oportunidad que contribuye enormemente a mi crecimiento personal y profesional.

Doy las gracias por todo su apoyo y atención, a mis sinodales Dr. Omar Arturo Domínguez, Dr. Jesús Patricio Ordaz Oliver y Dr. Sergio Rosario Salazar Cruz.

Al M. en C. Orlando García Pérez por su gran apoyo y asesoría, en verdad muchas gracias.

Gracias a CONACYT por el apoyo económico a través de la beca de maestría otorgada al CVU 552521.

Abstract

In this thesis an optimal control strategy with integral action is synthesized to stabilize the altitude of a Quadrotor helicopter. The system is subdivided in four subsystems, to make easier the attitude and position stabilization tasks. The integral action is considered as an additional state variable, it means that the new system has an extended state vector. In order to achieve the altitude stabilization an exact linearization (feedback state) together with an optimal control with integral action are applied to aerial robot. The remained controllers are obtained under the assumption of pitch, roll and yaw angles are always inside of a bounded region and the whole vector state is available. So the optimal control law with integral action is used to drive the attitude and position on the x and y axes. The proposal is experimentally tested considering a specific routine composed by the takeoff phase and altitude stabilization task, using the Optitrack vision system to compute the whole state of the aerial vehicle. Satisfactory results have been obtained when the proposed control strategy is applied. Finally, in order to do an energy consumption analysis of the optimal control strategy with integral action, it was compared with a PID controller, which was heuristically tuned with the dynamic system under similar conditions to those experienced when the optimal control was tested. Simulations results using optimal control are also presented.

Resumen

En este trabajo de tesis se presenta la estabilización en altura para un vehículo aéreo no tripulado, mediante un control integral óptimo con el enfoque de programación dinámica. Lo anterior con el fin de ahorrar energía en función de un índice de desempeño preestablecido, además de absorber perturbaciones causadas por el viento y dinámicas no modeladas en el sistema. Se controlan tanto la posición y orientación del vehículo. Para lograr lo anterior, el sistema es dividido en cuatro subsistemas y una acción integral es agregada como una variable de estado adicional en dichos subsistemas. En el control en altura, una linealización exacta y un control óptimo con acción integral son propuestos. El resto de los controladores son sintetizados asumiendo que el alabeo, el cabeceo y la guiñada están dentro de una región acotada. El control integral es usado también en la orientación y posición del plano horizontal (x, y) , dicho control se aplica en una planta en donde se permite la recuperación del estado completo mediante un sistema de visión que captura en tres dimensiones a través del programa Motive de Optitrack. Además un controlador PID sintonizado empíricamente es aplicado a la misma plataforma, y se hace una comparación del consumo de energía entre ambos controladores. También son presentados resultados de simulación al aplicar el algoritmo del controlador óptimo integral.

Índice

1. Introducción	13
1.1. Revisión bibliográfica de resultados previos UAV	14
1.2. Hipótesis	15
1.3. Justificación	15
1.4. Objetivos	16
1.4.1. Objetivo general	16
1.4.2. Objetivos particulares	16
1.5. Planteamiento del problema	16
1.6. Limitaciones	17
2. Plataforma experimental	18
2.1. Modelo dinámico	20
2.2. Conclusiones	27
3. Estrategia de control	28
3.1. Modelo dinámico discretizado	31
3.2. Control integral aplicado al modelo dinámico discreto	33
3.3. Conclusiones y Comentarios	39
4. Resultados	40
4.1. Resultados de simulación	40
4.2. Controlador PID	44
4.3. Controlador Óptimo Integral	47
4.4. Comparación entre controlador PID y controlador óptimo integral	50
4.5. Conclusión General.	54
A. Métodos empleados	55
A.1. Control óptimo de horizonte infinito vía programación dinámica	55
A.2. Control PID	59
A.3. Regulación vía control Integral	61
A.4. Filtro Complementario	63
A.5. Filtro de Kalman	64
B. Herramientas	71
B.1. Central inercial	71
B.1.1. Acelerómetro	71
B.1.2. Giroscopio	72
B.2. Programa MATLAB	73
B.3. Programa Dynamic C	74
B.4. Programa Motive de OptiTrack	75
B.5. Visual C++	75
C. Diagramas de flujo	77
D. Programa para resolver las DARE para simulación	79

Índice de figuras

1.	Plataforma experimental para ambiente controlado.	19
2.	Quadrotor basado en Parrot.	19
3.	Esquema funcional de la plataforma.	20
4.	Distribución y sentido de giro en los motores del UAV.	21
5.	Movimientos del UAV.	21
6.	Esquema de helicóptero con cuatro rotores.	22
7.	Posiciones y ángulos en un UAV.	26
8.	Orientación del UAV.	27
9.	Diagrama a bloques del modelo del UAV cargado en SIMULINK.	41
10.	Esquema de control en SIMULINK.	42
11.	Posiciones $x(t)$, $y(t)$, $z(t)$ simuladas.	43
12.	Errores en posición $x(t)$, $y(t)$, $z(t)$ simulados.	43
13.	Velocidades en $x(t)$, $y(t)$, $z(t)$ simuladas.	43
14.	Posiciones angulares $\theta(t)$, $\phi(t)$, $\psi(t)$ simuladas.	43
15.	Velocidades angulares $\dot{\theta}(t)$, $\dot{\phi}(t)$, $\dot{\psi}(t)$ simuladas.	43
16.	Señales de control óptimas $\tau_{\theta}(t)$, $\tau_{\phi}(t)$, $\tau_{\psi}(t)$ simuladas.	43
17.	Control óptimo $u(t)$ simulado.	44
18.	Trayectoria del UAV en 3D simulada.	44
19.	Posiciones y referencias $x(t)$, $y(t)$, $z(t)$ aplicando PID.	44
20.	Errores en posición $x(t)$, $y(t)$, $z(t)$ aplicando PID.	44
21.	Velocidad en $x(t)$ aplicando PID.	45
22.	Velocidad en $y(t)$ aplicando PID.	45
23.	Velocidad en $z(t)$ aplicando PID.	45
24.	Posición angular $\theta(t)$ aplicando PID.	45
25.	Posición angular $\phi(t)$ aplicando PID.	45
26.	Posición angular $\psi(t)$ aplicando PID.	45
27.	Velocidad angular $\dot{\theta}(t)$ aplicando PID.	46
28.	Velocidad angular $\dot{\phi}(t)$ aplicando PID.	46
29.	Velocidad angular $\dot{\psi}(t)$ aplicando PID.	46
30.	Señal de control $\tau_{\theta}(t)$ aplicando PID.	46
31.	Señal de control $\tau_{\phi}(t)$ aplicando PID.	46
32.	Señal de control $\tau_{\psi}(t)$ aplicando PID.	46
33.	Señal de control $u(t)$ aplicando PID.	47
34.	Trayectoria del UAV en 3D aplicando PID.	47
35.	Posiciones y referencias $x(t)$, $y(t)$, $z(t)$ al aplicar control óptimo integral.	48
36.	Errores en posición $x(t)$, $y(t)$, $z(t)$ al aplicar control óptimo integral.	48
37.	Velocidad en $x(t)$ al aplicar control óptimo integral.	48
38.	Velocidad en $y(t)$ al aplicar control óptimo integral.	48
39.	Velocidad en $z(t)$ al aplicar control óptimo integral.	48
40.	Posición angular $\theta(t)$ al aplicar control óptimo integral.	48
41.	Posición angular $\phi(t)$ al aplicar control óptimo integral.	49
42.	Posición angular $\psi(t)$ al aplicar control óptimo integral.	49
43.	Velocidad angular $\dot{\theta}(t)$ al aplicar control óptimo integral.	49

44.	Velocidad angular $\dot{\phi}(t)$ al aplicar control óptimo integral.	49
45.	Velocidad angular $\dot{\psi}(t)$ al aplicar control óptimo integral.	49
46.	Señal de control $\tau_{\theta}(t)$ al aplicar control óptimo integral.	49
47.	Señal de control $\tau_{\phi}(t)$ al aplicar control óptimo integral.	50
48.	Señal de control $\tau_{\psi}(t)$ al aplicar control óptimo integral.	50
49.	Señal de control $u(t)$ al aplicar control óptimo integral.	50
50.	Trayectoria del UAV en 3D al aplicar control óptimo integral.	50
51.	UAV Conectado al sensor de corriente.	51
52.	Señales de corriente al aplicar el control integral óptimo de horizonte infinito.	51
53.	Señales de corriente al aplicar el control PID.	52
54.	Señal PWM que permite saber el tiempo exacto de adquisición empleado.	52
55.	Potencia consumida por los dos controladores desde el enfoque energía cinética y potencial.	53
56.	Esquema control integral.	63
57.	Esquema del filtro complementario.	64
58.	Esquema de sistema discretizado.	65
59.	Representaciones del filtro de Kalman.	66
60.	Algoritmo del filtro de Kalman.	67
61.	Acelerómetro basado en tecnología MEMS.	72
62.	Componentes de aceleración en el eje X	72
63.	Giroscopio tecnología MEMS referido al eje X	73
64.	Programa MOTIVE de OptiTrack.	76
65.	Diagrama de flujo del algoritmo programado en el controlador Rabbit.	77
66.	Diagrama de flujo del algoritmo programado en la PC.	78
67.	Circuito para adquirir corriente.	82
68.	PCB vista inferior.	82
69.	Esquema de circuito electrónico para medir corriente.	83

Índice de tablas

1.	Tabla de comparación de índices de desempeño entre el controlador PID y el controlador óptimo integral.	50
2.	Tabla de consumo de energía por cada experimento realizado.	53
3.	Tabla de ahorro de energía enfoque consumo eléctrico.	53
4.	Tabla de ahorro de energía desde el enfoque energía cinética y potencial.	54

Capítulo 1

1. Introducción

Los vehículos aéreos no tripulados (Unmanned Aerial Vehicle, UAV por sus siglas en inglés), también conocidos como drones, se han convertido recientemente en una plataforma viable para tareas de vigilancia, monitoreo y exploración donde la presencia humana es peligrosa, imposible o inadecuada [1]. El alto número de pilotos caídos durante misiones del tipo de rastreo de enemigos en la segunda guerra mundial alentó la idea de emplear vehículos aéreos no tripulados [2], por esta razón los primeros UAV fueron desarrollados a finales de dicha guerra y a partir de entonces un considerable número de investigadores han trabajado en el diseño, análisis, control y operación de estos dispositivos [3]-[4].

Dentro del conjunto de los UAV se encuentran los mini helicópteros de cuatro rotores mejor conocidos como cuadricópteros, éstos se caracterizan por ser capaces de despegar y aterrizar de forma vertical y poseer un diseño mecánico relativamente simple [1] en comparación con un helicóptero convencional cuyo diseño mecánico es más complejo. El primer cuadricóptero registrado fue construido por los hermanos Breguet en 1907, lo nombraron *The Breguet-Richet Gyroplane N.1*, este primer modelo presentó graves problemas de estabilidad por lo que fue volado a bajas alturas, además sólo se controló su ascenso y descenso. En 1922 Jerome de Bothezat contruyó uno de los cuadricópteros más grandes que se hayan hecho, controlando vuelos a bajas alturas y movimientos hacia adelante y hacia a tras, sin embargo debido a su bajo rendimiento el proyecto fue cancelado. Dos años antes el constructor francés Etienne Oemichen fabricó un cuadricóptero similar al realizado por Bothezat pero con un número adicional de rotores para controlar posición. El diseño inicial no era alimentado y tenía que usarse un globo de hidrógeno atado a él para generar fuerza de elevación y estabilidad [5].

Un cuadricóptero es un dispositivo que posee 6 grados de libertad: las tres posiciones en el espacio x , y , z y las tres orientaciones alabeo (ϕ), cabeceo (θ) y guiñada (ψ). De forma general puede decirse que es un arreglo de cuatro hélices en forma de cruz. Los dos pares de propelas (1,3) y (2,4) giran en dirección opuesta. Variando la velocidad de las hélices se puede modificar la fuerza de elevación y generar movimiento vertical. Cambiando la velocidad de las cuatro hélices puede lograrse modificar los ángulos de alabeo, cabeceo y guiñada, y desplazar por ende el dron en diferentes direcciones [4].

La principal ventaja de los cuadricópteros es la versatilidad que presentan para generar movimientos verticales y su fácil construcción, sin embargo el modelo dinámico que define su comportamiento es altamente complejo debido a que se trata de un modelo no lineal [1]. En el mundo real no es posible encontrar sistemas ideales que no presenten incertidumbres o perturbaciones que afecten su desempeño o comportamiento, por esta razón en las últimas décadas, alrededor del mundo los investigadores se han enfocado en crear leyes de control robustas que

permitan estabilizar UAVs [6]. Algunos enfoques de control aplicados a los cuadricópteros para controlar su vuelo son redes neuronales artificiales, controladores lineales, control basado en modos deslizantes, “backstepping”, control adaptable, linealización por retroalimentación, controles no lineales con entradas acotadas [3], control óptimo, etc.

Debido al buen desempeño de los cuadricópteros para generar movimientos verticales y horizontales estables, así como para realizar vuelos estáticos en un punto, se han aplicado en diferentes tareas de carácter civil, científico y militar, por ejemplo en Europa la empresa Endesa ha iniciado con el uso de drones para revisar el estado de la red eléctrica en toda España [7]. En catástrofes como lo ocurrido en Fukushima Japon, se emplearon drones para observar el reactor nuclear y poder planear el procedimiento de limpieza de la zona afectada [8]. Algunos países como Estados Unidos de Norte América emplean los drones con fines bélicos, realizando ataques a distancia en zonas de conflicto [9]. Estas son sólo algunas de las aplicaciones que se les ha dado a los drones en todo el mundo. En México por citar algunas otras aplicaciones se puede mencionar que, en Sinaloa bajo la supervisión del Centro de Investigación y Estudios Superiores en Antropología Social (CIESAS) se emplean drones para estudiar zonas arqueológicas [10]. En el Instituto Nacional de Astrofísica, Óptica y Electrónica (INAOE) ubicado en Puebla se desarrolló recientemente un sistema de visión y aprendizaje para control y navegación de drones sin depender de la señal GPS o de personal capacitado [11]. El Instituto de Investigaciones Estéticas de la Universidad Nacional Autónoma de México (UNAM), utiliza un drone con una cámara fotográfica integrada para hacer levantamiento topográfico y recolección de datos para el estudio de sitios arqueológicos [12]. La Unidad Mixta Internacional-Laboratorio Franco Mexicano de Informática y Automática (UMI-LAFMIA) del Centro de Investigación y de Estudios Avanzados del Instituto Politécnico Nacional (CINVESTAV), desarrolla drones y algoritmos de control avanzados para mejorar el desempeño de éstos [13]. En el CINVESTAV Guadalajara se ha desarrollado un sistema de monitoreo mediante el uso de un drone y una cámara para monitorear la temperatura del cableado en la red eléctrica e incendios [13].

1.1. Revisión bibliográfica de resultados previos UAV

En este trabajo de tesis se aplica un control subóptimo (este término se aplica debido a que para usar la técnica se considera una linealización exacta sobre la altura, la cual podría no darse si existen dinámicas no modeladas o perturbaciones) vía regulación integral a un mini helicóptero de cuatro rotores. El control óptimo aplicado a vehículos aéreos no tripulados (UAV) ha sido estudiado recientemente, sin embargo éste representa todo un desafío para la teoría de control, aunado a esto, en casi todas las aplicaciones en tiempo real, el controlador Proporcional Integral Derivativo (PID) es empleado ampliamente en los UAV. Una de las razones por la que el controlador PID es ampliamente usado, es que éste puede ser sintonizado de forma heurística lo que permite que sea aplicado fácilmente, además puede usarse tanto en sistemas lineales como en no lineales, para los primeros existen diversos métodos que permiten sintonizar este tipo de controlador, para el segundo caso no existen técnicas generales. La presencia de incertidumbres causa que no pueda asegurarse que el error sea cero, sin embargo, gracias a la parte integral el control PID puede reducir el error debido a las incertidumbres o dinámicas no modeladas [14].

El controlador PID ha demostrado ser una poderosa herramienta, a pesar de que para algunas tareas específicas no es eficiente, puede ser implementado modificando los parámetros de

control bajo un esquema adaptativo o agregando términos no lineales. En este trabajo se usa la ventaja de la acción integral del PID mediante la aplicación de un controlador integral, con el fin de mejorar el desempeño en la estabilización de altura de un vehículo aéreo no tripulado.

El control integral aplicado a UAV's ha sido explorado en algunos trabajos reportados. Pude citarse [15], donde se usa un controlador no lineal predictivo de horizonte infinito H_∞ para seguir una trayectoria y estabilizar los movimientos rotacionales de un mini robot aéreo de cuatro rotores, en dicho trabajo son reportados resultados experimentales exitosos. En [16] una estrategia de control "backstepping" integral y una planeación de trayectorias, son combinadas para controlar un helicóptero mediante la estabilización de dirección punto a punto, se presentan resultados de simulación con el fin de mostrar el desempeño del sistema y su robustez. La misma técnica de control es presentada en [17], con el objetivo de controlar la orientación, altura y posición, en despegue, *hover* y aterrizaje, resultados de simulación y experimentales son presentados sobre un dispositivo OS4. Una comparación entre un controlador PID, un Regulador Cuadrático Lineal (Linear Quadratic Regulator, LQR de sus siglas en inglés), y un control no lineal (Adaptive Integral Backstepping Controller) se presentó en [18], en este trabajo se empleó una aproximación para el control no lineal basada en una metodología recursiva mediante el enfoque de Lyapunov, para ello se empleó la técnica "backstepping" y un esquema adaptativo, se reportaron resultados de simulación y experimentales. En [19] con el objetivo de mejorar el desempeño de un UAV, es usado un Regulador Cuadrático Lineal (LQR) con un término integral agregado, los estados no medibles son estimados mediante un observador, resultados experimentales satisfactorios son presentados.

1.2. Hipótesis

Debido a que la acción integral da robustez a un lazo de control, la regulación en un sistema vía control integral permite reducir la afectación causada por perturbaciones externas, además de corregir aquellas producidas por dinámicas no modeladas en éste. Por lo anterior se puede decirse que aplicar un control que contemple una acción integral a un vehículo aéreo no tripulado, permitirá reducir las afectaciones causadas por el viento y por términos no contemplados en el modelo matemático de éste.

1.3. Justificación

La principal desventaja de la configuración de los cuadricópteros es que al poseer 4 motores consumen más energía que un helicóptero convencional, reduciendo el tiempo de vuelo. Aproximadamente el vuelo de un cuadricóptero dura 20 minutos. Actualmente los vehículos aéreos no tripulados (UAV) son empleados en diversas aplicaciones de carácter civil, militar y científico, como por ejemplo entretenimiento, vigilancia, exploración, reconstrucción de zonas arqueológicas, monitoreo ambiental, etc. En algunas de estas tareas los drones deben volar en ambientes donde el viento perturba su desempeño, además los modelos empleados para aproximar su comportamiento no son exactos por lo que algunas dinámicas son despreciadas o no modeladas. Por lo anterior es indispensable aplicar técnicas de control avanzadas, que permitan absorber perturbaciones en vuelos al exterior y aquellas causadas por dinámicas no modeladas, cuidando el ahorro de energía, para dar la robustez necesaria al sistema sin comprometer demasiado el tiempo de vuelo del UAV.

1.4. Objetivos

1.4.1. Objetivo general

Sintetizar un control óptimo integral en un vehículo aéreo no tripulado de cuatro rotores, empleando el enfoque de programación dinámica, para ahorrar energía al estabilizar su altura y absorber incertidumbres y dinámicas no modeladas.

1.4.2. Objetivos particulares

- Diseñar un control integral en la posición y orientación de un quadrotor en ambiente controlado empleando el sistema de visión Flex 3 para obtener todo el estado del sistema.
- Implementar el control integral al modelo discretizado del sistema en ambiente controlado, empleando visual C++ y librerías de Optitrack.
- Implementar un controlador Proporcional Integral Derivativo (PID) a un cuadricóptero en ambiente controlado usando lenguaje C++, con el fin de comparar el consumo de energía entre éste y el controlador integral.
- Diseñar y armar un circuito que permita adquirir el consumo de corriente en tiempo real de la plataforma en ambiente controlado empleando MATLAB, para comparar el consumo real de corriente en el UAV al aplicar el controlador PID y el control integral.
- Realizar un análisis de consumo de energía entre los controles aplicados a la plataforma.

1.5. Planteamiento del problema

El uso de los vehículos aéreos no tripulados (UAV) presenta principalmente cuatro problemas fundamentales:

- El modelado y la identificación de los parámetros del sistema es altamente complejo.
- Son sistemas MIMO no lineales cuyo control es complicado y requiere de la aplicación de técnicas avanzadas.
- El consumo de energía es alto, lo que implica un corto tiempo en la duración de la batería que alimenta al sistema.
- Los sistemas de navegación precisos son de un costo muy elevado, lo que genera la necesidad de emplear sistemas menos precisos y compensar esto mediante algoritmos complejos.

En este trabajo se debe aplicar un control óptimo integral y un controlador PID a un UAV. Se emplea una plataforma la cual se nombrará *plataforma en ambiente controlado* esto debido a que los vuelos se realizan en un salón cerrado y condiciones controladas, además se usa un sistema de visión que permite obtener todo el estado del sistema. El problema que se aborda es minimizar el consumo de energía empleando un control óptimo con acción integral para aumentar el tiempo de vuelo en el UAV, para demostrar esto se realiza una comparación entre los índices de desempeño del PID y el control óptimo.

1.6. Limitaciones

Este trabajo presenta algunas limitaciones que son importantes de mencionar:

- El sistema de visión empleado presenta un retardo en la transmisión de datos, sin embargo, respecto al control de posición es posible calcularlo en una Computadora Personal (PC) y enviarlo a un procesador a bordo del UAV, con respecto a la orientación, el retardo inducido por el sistema de visión es suficientemente importante y no es posible enviar la señal de control desde la PC, por lo que el control de orientación se realizó a bordo empleando un microprocesador Rabbit.
- La altura máxima a la que se puede trabajar en esta plataforma es de 60 cm, ya que a partir de valores superiores las cámaras no son capaces de detectar el vehículo correctamente.
- La distancia entre la PC y el UAV no debe ser mayor a los 100 m ya que se emplean modems X-Bee S1 RF para esta tarea.
- El tiempo de muestreo usado en esta plataforma es de 50 ms, para tiempos menores no es posible aplicar el algoritmo debido al retardo causado por el sistema de visión.
- En esta plataforma se emplea un cable para alimentar al sistema electrónico, dicho cable va de la fuente de alimentación al UAV por esta razón existe una perturbación no modelada que afecta su desempeño, además la fuente de alimentación siempre entrega una corriente constante sin importar el tiempo de vuelo, lo que no pasa con la baterías que van sufriendo una descarga conforme pasa el tiempo de vuelo, puede decirse que los resultados presentados en este trabajo dependen de que la fuente de alimentación entregue carga constante.
- Las pruebas son realizadas en una habitación con ambiente controlado, donde la temperatura, presión, humedad, intensidad luminosa, y velocidad del viento no sufren cambios drásticos.

Capítulo 2

2. Plataforma experimental

En este trabajo se emplea una plataforma para aplicar el control óptimo integral y el control PID, la cual se nombrará “Plataforma para ambiente controlado”, ésta cuenta con un sistema de visión que permite tener la medida de todas las variables de estado del sistema. En el presente capítulo se da una descripción general de dicha plataforma, además se explica el modelo matemático que la describe.

La plataforma experimental para vuelo en ambiente controlado emplea el sistema de visión Flex 3 de OptiTrack, el cual es capaz de sensar la posición y orientación de un objeto, mediante marcadores colocados en éste, 12 cámaras envían información al software Motive, dicho programa reconstruye la trayectoria del objeto. Mediante una librería creada por la empresa OptiTrack se comunican los softwares Visual C++ y Motive, para que éste último envíe la posición y orientación del UAV a Visual C++. Las leyes de control se programan en lenguaje C, las velocidades son calculadas numéricamente y se usa un filtro digital FIR pasa bajas para reducir afectaciones causadas por el ruido, a una velocidad de muestreo de 50 ms. Entre la PC y el UAV existe comunicación a través de un modem que usa un Xbee Pro S1 como transceiver mediante comunicación RS232 a una velocidad de 38400 bps (ver figura 1).

El UAV empleado está armado sobre un frame del Parrot y posee embarcados en él, un microprocesador Rabbit RCM4300, una IMU 3DM-GX3 de MicroStrain, un transceiver Xbee Pro S1 y un receptor de radio, como se muestra en la figura 2.

El sistema de visión de Optitrack envía las posiciones espaciales y angulares $x(k)$, $y(k)$, $z(k)$, $\theta(k)$, $\phi(k)$ y $\psi(k)$ del UAV a la PC, mediante el programa Motive se reconstruyen y envían al software Visual C++ dichas posiciones. En el programa Visual C++ son calculados los controles $u_{PC}(k)$ (control de altura calculado en la PC), $\tau_{\phi PC}(k)$ (control de alabeo calculado en la PC) y $\tau_{\theta PC}(k)$ (control de cabeceo calculado en la PC), dichos controles son enviados mediante el modem Xbee vía RS232 inalámbrico al modem que está embarcado en el UAV, para que éste los envíe al microprocesador Rabbit.

Se emplea un Radio RF Futaba como calibrador, mediante las señales de control (las cuales son fijas) enviadas a través de él, se modifican, la altura y las orientaciones en el UAV, esto para compensar errores en la orientación causados por lecturas de la IMU o el sistema de visión. Las señales de calibración enviadas por el Radio son: $u_c(k)$, $\tau_{\phi_c}(k)$, $\tau_{\theta_c}(k)$ y $\tau_{\psi_c}(k)$, la primera de ellas sirve para modificar la altura y las tres últimas para compensar variaciones en las orientaciones angulares del UAV, estas señales son recibidas por el receptor de radio que está embarcado en el UAV y son enviadas vía I^2C al microprocesador Rabbit.

El procesador Rabbit lee las orientaciones angulares de la IMU $\theta_{IMU}(k)$, $\phi_{IMU}(k)$ y $\psi_{IMU}(k)$, vía protocolo I^2C , para calcular controles en orientación de una forma más rápida que la PC ya que el sistema de visión causa un retardo que afecta la estabilidad en orientación. Por lo anterior

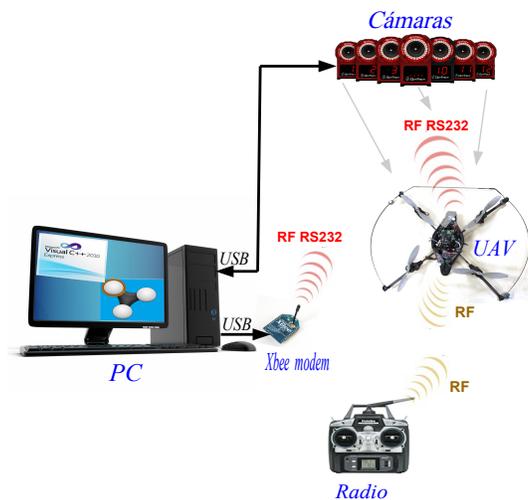


Figura 1: Plataforma experimental para ambiente controlado.

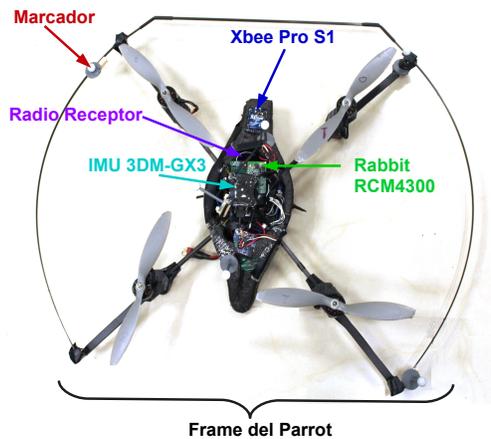


Figura 2: Quadrotor basado en Parrot.

las señales de control debidas a las orientaciones angulares son calculadas en el Rabbit, y sumadas a las señales de control calculadas en la PC (calculadas exclusivamente para las posiciones espaciales $x(t)$, $y(t)$, $z(t)$) y a las señales de calibración enviadas por el radio.

Al sumar las señales de control y de calibración se generan los controles $u(k)$, $\tau_\phi(k)$, $\tau_\theta(k)$ y $\tau_\psi(k)$ los cuales modifican las velocidades en los motores del UAV para modificar su posición y

orientación. El esquema funcional mostrado en la figura 3 ilustra lo anterior. En el Apéndice A son mostrados los diagramas de flujo de los algoritmos empleados en la PC y en el microprocesador embarcado en el UAV.

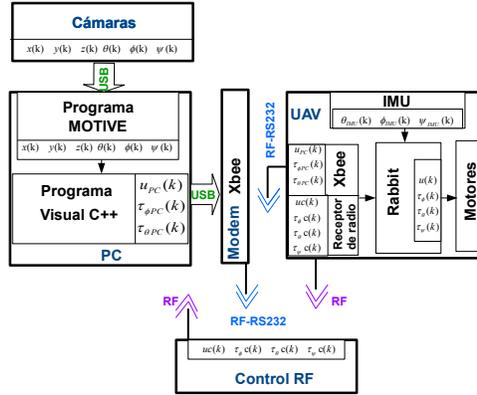


Figura 3: Esquema funcional de la plataforma.

Para la plataforma la distribución de los motores y sus sentidos de giro, son fijados como se muestra en la figura 4, dicha distribución es el estándar usado por 3D Robotics en el autopilot Pix Hawk. Para generar movimientos en el quadrotor se procede a variar la velocidad de los motores tal y como se muestra en la figura 5, en función de las combinaciones mostradas en dicha figura se logra crear las señales de control $u, \tau_{\theta}, \tau_{\phi}$ y τ_{ψ} , por tanto, por ejemplo, si se desea incrementar la fuerza de elevación en el UAV se procede a aumentar la velocidad de los cuatro motores forma homogénea, para decrementar dicha fuerza se reduce la velocidad de los cuatro motores de la misma manera, se procede igual para variar los torques, así que, para modificar la señal de control τ_{θ} (relacionado con el ángulo de cabeceo, es decir que permite avanzar hacia adelante y atrás) se aumenta la velocidad de los motores 2 y 4 y a decrementar la velocidad de los motores 1 y 3, con lo anterior se estaría generando un torque positivo en τ_{θ} , para generar un par negativo se realiza la acción contraria. Lo mismo ocurre con los otros dos pares τ_{ϕ} y τ_{ψ} .

2.1. Modelo dinámico

En este apartado el modelo dinámico de un helicóptero con cuatro rotores es presentado, dicho modelo ha sido obtenido aplicando el método de Euler-Lagrange. Sean \mathcal{R}_w y \mathcal{R}_b dos sistemas de coordenadas. \mathcal{R}_w es definido como el sistema de coordenadas inercial de referencia y \mathcal{R}_b es otro sistema de coordenadas ligado al cuerpo del robot volador. El vector de coordenadas generalizado para el robot es $v \in \mathbb{R}^6$, descrito como:

$$v = (x \ y \ z \ \psi \ \theta \ \phi)^T$$

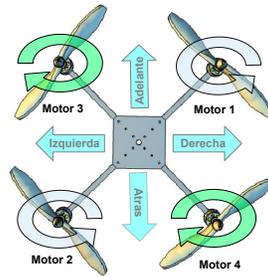


Figura 4: Distribución y sentido de giro en los motores del UAV.

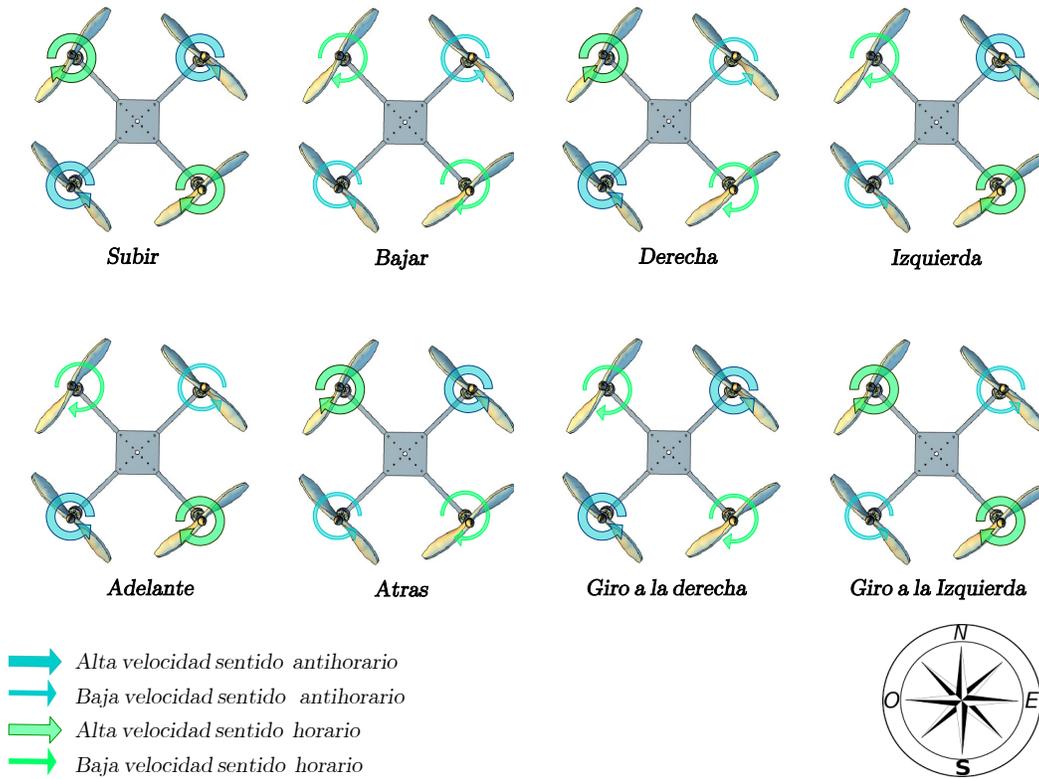


Figura 5: Movimientos del UAV.

Donde el vectores $(x \ y \ z)$ representa la posición del centro de masa del robot volador respecto

al sistema \mathcal{R}_w . el vector compuesto por $(\psi \ \theta \ \phi)$ define los tres ángulos de Euler, el cabeceo, el alabeo y la guiñada respectivamente. Con ellos es representada la orientación del robot.

De manera que se tienen dos subsistemas, un subsistema ξ referente a la parte traslacional y otro η para la parte rotacional, es decir, $\xi = (x \ y \ z)^T \in \mathbb{R}^3$, $\eta = (\psi \ \theta \ \phi)^T \in \mathbb{R}^3$, ver figura 6.

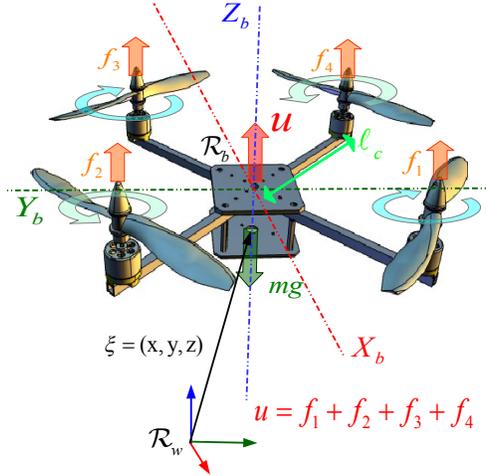


Figura 6: Esquema de helicóptero con cuatro rotores.

La energía cinética de traslación del helicóptero está dada por la expresión:

$$T_{tras} = \frac{m}{2} \dot{\xi}^T \dot{\xi}$$

Donde m representa la masa del helicóptero, contemplando ahora la parte rotacional la energía cinética es dada por:

$$T_{rot} = \frac{1}{2} \Omega^T \mathbb{I} \Omega \tag{2.1}$$

Donde Ω es el vector de velocidades angulares e \mathbb{I} es la matriz de inercia. Sea W el vector de velocidades angulares expresado respecto al marco de referencia \mathcal{R}_b , éste está ligado directamente con las velocidades generalizadas η por la relación cinemática estándar [20].

$$\Omega = W_\eta \dot{\eta} \tag{2.2}$$

Reemplazando (2.2) en (2.1) se tiene:

$$T_{rot} = \frac{1}{2} \dot{\eta}^T W_{\eta}^T \mathbb{I} W_{\eta} \dot{\eta} \quad (2.3)$$

Siendo.

$$W_{\eta} = \begin{bmatrix} -\text{sen } \theta & 0 & 1 \\ \text{sen } \phi \cos \theta & \cos \phi & 0 \\ \cos \phi \cos \theta & -\text{sen } \phi & 0 \end{bmatrix} \quad \mathbb{I} = \begin{bmatrix} I_{xx} & 0 & 0 \\ 0 & I_{yy} & 0 \\ 0 & 0 & I_{zz} \end{bmatrix}$$

Entonces el vector Ω está dado por:

$$\Omega = \begin{bmatrix} \dot{\phi} - \text{sen } \psi \\ \dot{\psi} \text{sen } \phi \cos \theta + \dot{\theta} \cos \phi \\ \dot{\psi} \cos \phi \cos \theta - \dot{\theta} \text{sen } \phi \end{bmatrix}$$

Definiendo:

$$\mathbb{J}(\eta) \triangleq W_{\eta}^T \mathbb{I} W_{\eta}$$

Implica que la ecuación (2.3) se puede escribir como:

$$T_{rot} = \frac{1}{2} \dot{\eta}^T \mathbb{J} \dot{\eta}$$

Donde \mathbb{J} es la matriz de inercia para la energía cinética de rotación, expresada en bajo las coordenadas generalizadas η .

La fuerza que contribuye a la energía potencial presentada en el sistema dinámico es la fuerza U debida a la aceleración gravitacional, expresada como:

$$U = mgz$$

Siendo z la altura del helicóptero. Considerando las ecuaciones anteriores el Lagrangiano está dado por:

$$\begin{aligned} \mathcal{L}(g, \dot{g}) &= T_{tras} + T_{rot} - U \\ &= \frac{m}{2} \dot{\xi}^T \dot{\xi} + \frac{1}{2} \dot{\eta}^T \mathbb{J} \dot{\eta} - mgz \end{aligned}$$

Para obtener el modelo dinámico del helicóptero, se consideran las ecuaciones de Euler-Lagrange, tomando en cuenta el vector de fuerzas externas generalizado, esto es:

$$\frac{d}{dt} \left(\frac{\partial \mathcal{L}}{\partial \dot{v}} \right) - \left(\frac{\partial \mathcal{L}}{\partial v} \right) = F$$

Donde $F = (F_\xi, \tau)$. F_ξ es el vector de fuerzas aplicado al robot aéreo. Dichas fuerzas son debidas a las entradas de control aplicadas para producir movimientos de traslación y τ es el vector de momentos generalizados producidos por las entradas de control aplicadas para efectuar movimientos de rotación. Todas las otras fuerzas aerodinámicas presentadas en el sistema no son tomadas en cuenta debido a que son despreciables comparadas con las fuerzas producidas por la entradas de control. Las fuerzas aplicadas al mini helicóptero respecto al marco de referencia \mathcal{R}_b son:

$$F_b = \begin{bmatrix} 0 \\ 0 \\ u \end{bmatrix}$$

donde:

$$u = f_1 + f_2 + f_3 + f_4$$

$$f_i = k_i \omega_i^2, \quad i = 1, \dots, 4$$

Donde $k_i > 0$ es un parámetro que depende de la densidad del aire ρ_a , del área A_h de la hélice, de su forma y del ángulo de sus palas. ω_i es la velocidad angular de cada motor i ($M_i, i=1, \dots, 4$), este conjunto de fuerzas expresado en \mathcal{R}_w se transforma en:

$$F_\xi = \mathbf{R}F_b$$

Donde $\mathbf{R} \in \mathbb{R}^{3 \times 3}$ es la matriz de rotación del mini helicóptero de cuatro rotores, ésta es:

$$\mathbf{R} = \begin{pmatrix} \cos \theta \cos \psi & \text{sen } \phi \text{ sen } \theta \cos \psi - \text{sen } \psi \cos \phi & \text{sen } \phi \text{ sen } \psi + \text{sen } \theta \cos \phi \cos \psi \\ \text{sen } \psi \cos \theta & \text{sen } \phi \text{ sen } \theta \text{ sen } \psi + \cos \phi \cos \psi & \text{sen } \theta \text{ sen } \psi \cos \phi - \text{sen } \phi \cos \psi \\ -\text{sen } \theta & \text{sen } \phi \cos \theta & \cos \phi \cos \theta \end{pmatrix} \quad (2.4)$$

Esta representación de la matriz \mathbf{R} es muy empleada en aeronáutica [20]. El vector de momentos generalizado sobre las variables de η es:

$$\tau = \begin{bmatrix} \tau_\psi \\ \tau_\theta \\ \tau_\phi \end{bmatrix}$$

siendo:

$$\begin{aligned} \tau_\psi &= \sum_{i=1}^4 \tau_{M_i} \\ \tau_\theta &= (f_2 - f_4)l_c \\ \tau_\phi &= (f_3 - f_1)l_c \end{aligned}$$

ℓ_c es la distancia entre los ejes de los motores y el centro de gravedad del helicóptero y τ_{M_i} es el momento producido por el motor M_i alrededor del centro de gravedad del vehículo.

Debido a que el Lagrangiano no contiene términos en la energía cinética combinando $\dot{\xi}$ con $\dot{\eta}$, las ecuaciones de Euler-Lagrange pueden ser divididas en las dinámicas para las coordenadas de ξ y las coordenadas de η .

La ecuación de Euler-Lagrange para el movimiento de traslación es:

$$\frac{d}{dt} \left[\frac{\partial \mathcal{L}_{tras}}{\partial \dot{\xi}} \right] - \left(\frac{\partial \mathcal{L}_{tras}}{\partial \xi} \right) = F_\xi$$

se obtiene:

$$m \ddot{\xi} + \begin{pmatrix} 0 \\ 0 \\ mg \end{pmatrix} = F_\xi$$

De manera similar, para las coordenadas de η se puede escribir:

$$\frac{d}{dt} \left[\frac{\partial \mathcal{L}_{rot}}{\partial \dot{\eta}} \right] - \left(\frac{\partial \mathcal{L}_{rot}}{\partial \eta} \right) = \tau$$

obteniendo:

$$\begin{aligned} \tau &= \frac{d}{dt} \left[\dot{\eta}^T \mathbb{J} \frac{\partial \dot{\eta}}{\partial \dot{\eta}} \right] - \left(\frac{1}{2} \frac{\partial}{\partial \eta} (\dot{\eta}^T \mathbb{J} \dot{\eta}) \right) \\ &= \mathbb{J} \ddot{\eta} + \dot{\mathbb{J}} \dot{\eta} - \frac{1}{2} \frac{\partial}{\partial \eta} (\dot{\eta}^T \mathbb{J} \dot{\eta}) \end{aligned} \tag{2.5}$$

Definiendo el vector de Coriolis como:

$$\mathcal{C}(\eta, \dot{\eta}) = \dot{\mathbb{J}} \dot{\eta} - \frac{1}{2} \frac{\partial}{\partial \eta} (\dot{\eta}^T \mathbb{J} \dot{\eta})$$

Reemplazando el vector $\mathcal{C}(\eta, \dot{\eta})$ en (2.5), se tiene:

$$\mathbb{J} \ddot{\eta} + \mathcal{C}(\eta, \dot{\eta}) = \tau$$

Reescribiendo (2.5), se obtiene:

$$\mathbb{J} \ddot{\eta} + \left(\dot{\mathbb{J}} - \frac{1}{2} \frac{\partial}{\partial \eta} (\dot{\eta} \mathbb{J}) \right) \dot{\eta} = \tau$$

$$\mathbb{J}\ddot{\eta} + \mathbb{C}(\eta, \dot{\eta})\dot{\eta} = \tau$$

Donde $\mathbb{C}(\eta, \dot{\eta})$ es la matriz de coriolis, en ella están comprendidos los términos giroscópicos y centrífugos relacionados a la dependencia de η sobre \mathbb{J} . Con el fin de simplificar el análisis se propone un cambio de variable en la entrada.

$$\dot{\eta} = \tilde{\tau}$$

lo que implica que:

$$\mathbb{J}\tilde{\tau} + \mathbb{C}(\eta, \dot{\eta})\dot{\eta} = \tau$$

$\tilde{\tau} = [\tilde{\tau}_\psi, \tilde{\tau}_\theta, \tilde{\tau}_\phi]^T$ es el vector con las nuevas entradas, las cuales son aplicadas directamente a los momentos angulares generalizados τ_ψ , τ_θ y τ_ϕ .

Por lo anterior el sistema puede ser reescrito como [21]:

$$\begin{aligned} F_\xi &= m\ddot{\xi} + mgE_z \\ \dot{\eta} &= \tilde{\tau} \end{aligned}$$

F_ξ está definido como:

$$F_\xi = \begin{bmatrix} -u \sin \theta \\ u \sin \phi \cos \theta \\ u \cos \phi \cos \theta \end{bmatrix}$$

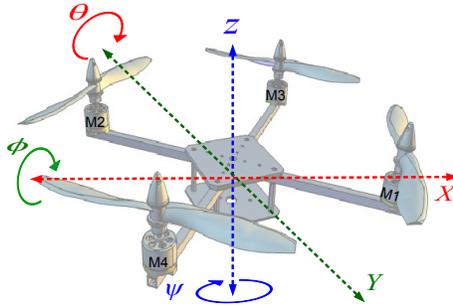


Figura 7: Posiciones y ángulos en un UAV.

Finalmente al multiplicar se obtiene que un robot aéreo puede ser representado por el siguiente sistema de ecuaciones no lineales:

$$\begin{aligned}m\ddot{x} &= -u \sin \theta \\m\ddot{y} &= u \cos \theta \sin \phi \\m\ddot{z} &= u \cos \theta \cos \phi - mg \\ \ddot{\phi} &= \tilde{\tau}_\phi \\ \ddot{\theta} &= \tilde{\tau}_\theta \\ \ddot{\psi} &= \tilde{\tau}_\psi\end{aligned}\tag{2.6}$$

Donde x , y son los desplazamientos en el plano horizontal y z es la posición vertical, ψ es el ángulo de guiñada en torno al eje z , θ es el ángulo de cabeceo respecto al eje y , ϕ es el ángulo de alabeo entorno al eje x . Las entradas de control son: u , $\tilde{\tau}_\phi$, $\tilde{\tau}_\theta$, y $\tilde{\tau}_\psi$. u es la fuerza generada por los motores para elevar el UAV. $\tilde{\tau}_\phi$, $\tilde{\tau}_\theta$, y $\tilde{\tau}_\psi$ son los torques generados en torno a cada eje respectivamente, como se muestra en las figuras 7 y 8.

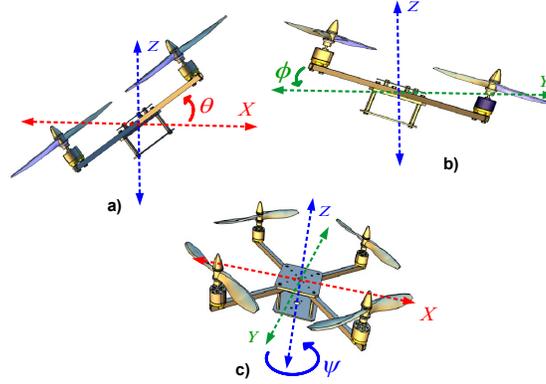


Figura 8: Orientación del UAV.

2.2. Conclusiones

En el presente capítulo fue descrita la plataforma experimental usada para aplicar el control PID y el control integral óptimo, además se da una descripción de el proceso realizado para llevar acabo dichas tareas. Es mostrado brevemente el modelo matemático empleado para describir el comportamiento dinámico de un mini helicóptero de cuatro rotores, mediante el enfoque Euler-Lagrange.

Capítulo 3

En este capítulo se describe la estrategia de control aplicada a la plataforma experimental previamente mostrada en el Capítulo 2. Para controlar el UAV se procede como en [22], donde el sistema es dividido en cuatro subsistemas y una linealización por retroalimentación en el control es aplicada en la dinámica z que representa matemáticamente el comportamiento en altura del minihelicóptero.

3. Estrategia de control

Para iniciar el proceso de síntesis de la estrategia de control a aplicar en este trabajo, se realiza la asignación de variables de estado sobre el modelo matemático del minihelicóptero descrito en la ecuación (2.6) dentro del Capítulo 2. Esta asignación de variables está descrita de la siguiente manera:

$$\begin{array}{l}
 x_1 = x \\
 x_2 = \dot{x} \\
 x_3 = y \\
 x_4 = \dot{y} \\
 x_5 = z \\
 x_6 = \dot{z} \\
 x_7 = \phi \\
 x_8 = \dot{\phi} \\
 x_9 = \theta \\
 x_{10} = \dot{\theta} \\
 x_{11} = \psi \\
 x_{12} = \dot{\psi}
 \end{array}
 \Rightarrow
 \left[\begin{array}{l}
 \dot{x}_1 = x_2 \\
 \dot{x}_2 = -\frac{u}{m} \text{sen}(x_9) \\
 \dot{x}_3 = x_4 \\
 \dot{x}_4 = \frac{u}{m} \cos(x_9) \text{sen}(x_7) \\
 \dot{x}_5 = x_6 \\
 \dot{x}_6 = \frac{u}{m} \cos(x_9) \cos(x_7) - g \\
 \dot{x}_7 = x_8 \\
 \dot{x}_8 = \tau_\phi \\
 \dot{x}_9 = x_{10} \\
 \dot{x}_{10} = \tau_\theta \\
 \dot{x}_{11} = x_{12} \\
 \dot{x}_{12} = \tau_\psi
 \end{array} \right] \quad (3.1)$$

Al utilizar esta asignación de variables de estado (3.1), evidentemente se obtiene la ecuación de estado siguiente:

$$\left[\begin{array}{l}
 \dot{x}_1 \\
 \dot{x}_2 \\
 \dot{x}_3 \\
 \dot{x}_4 \\
 \dot{x}_5 \\
 \dot{x}_6 \\
 \dot{x}_7 \\
 \dot{x}_8 \\
 \dot{x}_9 \\
 \dot{x}_{10} \\
 \dot{x}_{11} \\
 \dot{x}_{12}
 \end{array} \right]
 =
 \left[\begin{array}{l}
 x_2 \\
 0 \\
 x_4 \\
 0 \\
 x_6 \\
 -g \\
 x_8 \\
 0 \\
 x_{10} \\
 0 \\
 x_{12} \\
 0
 \end{array} \right]
 +
 \left[\begin{array}{cccc}
 0 & 0 & 0 & 0 \\
 -\frac{\text{sen}(x_9)}{m} & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 \\
 \frac{\cos(x_9) \text{sen}(x_7)}{m} & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 \\
 \frac{\cos(x_9) \cos(x_7)}{m} & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 \\
 0 & 1 & 0 & 0 \\
 0 & 0 & 0 & 0 \\
 0 & 0 & 1 & 0 \\
 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 1
 \end{array} \right]
 \left[\begin{array}{l}
 u \\
 \tau_\phi \\
 \tau_\theta \\
 \tau_\psi
 \end{array} \right] \quad (3.2)$$

A partir de la ecuación de estado (3.2) se realiza la síntesis de la ley de control óptima con acción integral. Como anteriormente se mencionó, la estrategia consiste en dividir el sistema en 4 subsistemas los cuales están definidos de la siguiente forma:

- Subsistema z .

$$\begin{aligned}\dot{x}_5 &= x_6 \\ \dot{x}_6 &= \frac{u}{m} \cos(x_9) \cos(x_7) - g\end{aligned}\quad (3.3)$$

- Subsistema ψ .

$$\begin{aligned}\dot{x}_{11} &= x_{12} \\ \dot{x}_{12} &= \tau_\psi\end{aligned}\quad (3.4)$$

- Subsistema y - ϕ .

$$\begin{aligned}\dot{x}_3 &= x_4 \\ \dot{x}_4 &= \frac{u}{m} \cos(x_9) \text{sen}(x_7) \\ \dot{x}_7 &= x_8 \\ \dot{x}_8 &= \tau_\phi\end{aligned}\quad (3.5)$$

- Subsistema x - θ .

$$\begin{aligned}\dot{x}_1 &= x_2 \\ \dot{x}_2 &= -\frac{u}{m} \text{sen}(x_9) \\ \dot{x}_9 &= x_{10} \\ \dot{x}_{10} &= \tau_\theta\end{aligned}\quad (3.6)$$

Apoyándose en el trabajo presentado en [22], se propone la aplicación de una ley de control $u(t)$ que permite realizar una linealización por retroalimentación de estado en el subsistema (3.3). Este control linealizante está definido como:

$$u = \frac{m(u_1 + g)}{(\cos\theta \cos\phi)}\quad (3.7)$$

Entonces lo que se busca primeramente es sintetizar las leyes de control τ_θ y τ_ϕ , que sean lo suficientemente robustas y de dinámica rápida de tal forma que las condiciones $\tan(\theta) \approx \theta$ y $\tan(\phi) \approx \phi$ se cumplan para un tiempo relativamente pequeños. Suponiendo, que las condiciones anteriores se cumplen, entonces cada uno de los subsistemas descritos en (3.3)-(3.6) pueden ser representados de la siguiente manera:

$$\begin{bmatrix} \dot{x}_5 \\ \dot{x}_6 \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}}_{A_z} \begin{bmatrix} x_5 \\ x_6 \end{bmatrix} + \underbrace{\begin{bmatrix} 0 \\ 1 \end{bmatrix}}_{B_z} [u_1]$$

$$\begin{bmatrix} \dot{x}_{11} \\ \dot{x}_{12} \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}}_{A_\psi} \begin{bmatrix} x_{11} \\ x_{12} \end{bmatrix} + \underbrace{\begin{bmatrix} 0 \\ 1 \end{bmatrix}}_{B_\psi} [\tau_\psi]$$

$$\begin{bmatrix} \dot{x}_3 \\ \dot{x}_4 \\ \dot{x}_7 \\ \dot{x}_8 \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & g & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}}_{A_{y\phi}} \begin{bmatrix} x_3 \\ x_4 \\ x_7 \\ x_8 \end{bmatrix} + \underbrace{\begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}}_{B_{y\phi}} [\tau_\phi]$$

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_9 \\ \dot{x}_{10} \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & -g & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}}_{A_{x\theta}} \begin{bmatrix} x_1 \\ x_2 \\ x_9 \\ x_{10} \end{bmatrix} + \underbrace{\begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}}_{B_{x\theta}} [\tau_\theta]$$

Es claro que los subsistemas presentados anteriormente son lineales y con representación genérica indicada a continuación respectivamente:

$$\begin{aligned} \dot{x}_z &= A_z x_z + B_z u_z \\ \dot{x}_\psi &= A_\psi x_\psi + B_\psi \tau_\psi \\ \dot{x}_{y\phi} &= A_{y\phi} x_{y\phi} + B_{y\phi} \tau_\phi \\ \dot{x}_{x\theta} &= A_{x\theta} x_{x\theta} + B_{x\theta} \tau_\theta \end{aligned} \tag{3.8}$$

Ahora bien, retomando la ecuación (3.7) se define a la señal u_1 como la ley de control óptimo. Para obtener la ley de control óptima se debe resolver de forma numérica una Ecuación Algebraica de tipo Riccati, este control está definido como:

$$u_1 = -R_z^{-1} B_z^T \mathbb{P}_z x_z(t)$$

donde u_1 está sujeto al índice de desempeño cuadrático J_z dado por:

$$J_z = \int_0^\infty (x_z^T Q_z x_z + u_1^2 R_z) dt$$

El control óptimo τ_ψ para la regulación del ángulo de guiñada ψ del cuadricóptero, está definido de la manera siguiente:

$$\tau_\psi = -R_\psi^{-1} B_\psi^T \mathbb{P}_\psi x_\psi(t)$$

el cual está sujeto al índice de desempeño J_ψ definido por:

$$J_\psi = \int_0^\infty (x_\psi^T Q_\psi x_\psi + \tau_\psi^2 R_\psi) dt$$

Para el caso del control óptimo τ_ϕ dedicado a regular el ángulo de alabeo ϕ del UAV, se encuentra definido como:

$$\tau_{y\phi} = -R_{y\phi}^{-1} B_{y\phi}^T \mathbb{P}_{y\phi} x_{y\phi}(t)$$

con índice de desempeño $J_{y\phi}$ asociado definido por:

$$J_{y\phi} = \int_0^\infty (x_{y\phi}^T Q_{y\phi} x_{y\phi} + \tau_{y\phi}^2 R_{y\phi}) dt$$

Finalmente la señal de control óptimo relacionado con el ángulo de cabeceo θ que está denotado por τ_θ se encuentra definido de la forma siguiente:

$$\tau_{x\theta} = -R_{x\theta}^{-1} B_{x\theta}^T \mathbb{P}_{x\theta} x_{x\theta}(t)$$

y su respectivo índice de desempeño $J_{x\theta}$ es:

$$J_{x\theta} = \int_0^\infty (x_{x\theta}^T Q_{x\theta} x_{x\theta} + \tau_{x\theta}^2 R_{x\theta}) dt$$

3.1. Modelo dinámico discretizado

En razón de que en la presente tesis se desarrollan validaciones experimentales de la estrategia de control propuesta para la estabilización del cuadricóptero, es importante considerar que estas validaciones involucran la utilización y aplicación de dispositivos de cómputo (microcontrolador). En función de esta reciente condición, se hace necesario encontrar la representación discreta tanto del modelo como de las leyes de control hasta ahora sintetizadas. Para tal fin se emplea la aproximación de Euler sobre los subsistemas definidos en la ecuación (3.8) con tiempo de muestreo T , obteniéndose lo siguiente:

- Subsistema $z(k)$.

$$\begin{aligned} x_5(k+1) &= Tx_6(k) + x_5(k) \\ x_6(k+1) &= T \left[\frac{u(k)}{m} \cos(x_9(k)) \cos(x_7(k)) - g \right] + x_6(k) \end{aligned}$$

- Subsistema $\psi(k)$.

$$\begin{aligned} x_{11}(k+1) &= Tx_{12}(k) + x_{11}(k) \\ x_{12}(k+1) &= T\tau_\psi(k) + x_{12}(k) \end{aligned}$$

- Subsistema $y\text{-}\phi(k)$.

$$\begin{aligned} x_3(k+1) &= Tx_4(k) + x_3(k) \\ x_4(k+1) &= T \left[\frac{u(k)}{m} \cos(x_9(k)) \text{sen}(x_7(k)) \right] + x_4(k) \\ x_7(k+1) &= Tx_8(k) + x_7(k) \\ x_8(k+1) &= T\tau_\phi(k) + x_8(k) \end{aligned}$$

- Subsistema $x\text{-}\theta(k)$.

$$\begin{aligned} x_1(k+1) &= Tx_2(k) + x_1(k) \\ x_2(k+1) &= T \left[-\frac{u(k)}{m} \text{sen}(x_9(k)) \right] + x_2(k) \\ x_9(k+1) &= Tx_{10}(k) + x_9(k) \\ x_{10}(k+1) &= T\tau_\theta(k) + x_{10}(k) \end{aligned}$$

Para eliminar las no linealidades se aplica un proceso análogo al aplicado para el caso continuo. Entonces, realizando una linealización exacta con la ley de control $u(k)$ propuesta, queda de la siguiente forma:

$$u(k) = m(u_1(k) + g)(\cos(x_9(k))\cos(x_7(k)))^{-1}$$

Lo cual causa que los subsistemas discretos queden definidos de la forma que a continuación se presenta:

$$\begin{bmatrix} x_5(k+1) \\ x_6(k+1) \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix}}_{A_{z(k)}} \begin{bmatrix} x_5(k) \\ x_6(k) \end{bmatrix} + \underbrace{\begin{bmatrix} 0 \\ T \end{bmatrix}}_{B_{z(k)}} [u_1(k)] \quad (3.9)$$

$$\begin{bmatrix} x_{11}(k+1) \\ x_{12}(k+1) \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix}}_{A_{\psi(k)}} \begin{bmatrix} x_{11}(k) \\ x_{12}(k) \end{bmatrix} + \underbrace{\begin{bmatrix} 0 \\ T \end{bmatrix}}_{B_{\psi(k)}} [\tau_\psi(k)] \quad (3.10)$$

$$\begin{bmatrix} x_3(k+1) \\ x_4(k+1) \\ x_7(k+1) \\ x_8(k+1) \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & gT & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{bmatrix}}_{A_{y\phi(k)}} \begin{bmatrix} x_3(k) \\ x_4(k) \\ x_7(k) \\ x_8(k) \end{bmatrix} + \underbrace{\begin{bmatrix} 0 \\ 0 \\ 0 \\ T \end{bmatrix}}_{B_{y\phi(k)}} [\tau_\phi(k)] \quad (3.11)$$

$$\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \\ x_9(k+1) \\ x_{10}(k+1) \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & -gT & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{bmatrix}}_{A_{x\theta(k)}} \begin{bmatrix} x_1(k) \\ x_2(k) \\ x_9(k) \\ x_{10}(k) \end{bmatrix} + \underbrace{\begin{bmatrix} 0 \\ 0 \\ 0 \\ T \end{bmatrix}}_{B_{x\theta(k)}} [\tau_\theta(k)] \quad (3.12)$$

Para que la linealización anterior de los subsistemas sea valida debe garantizarse que $\tau_\theta(k)$ es tal que $\tan(\theta(k)) \approx \theta(k)$ y que $\tau_\phi(k)$ es lo suficientemente robusta para que $\tan(\phi(k)) \approx \phi(k)$. Entonces si estas condiciones son satisfechas, los cuatro subsistemas linealizados están dados en la ecuación como a continuación:

$$\begin{aligned} x_z(k+1) &= A_z x_z(k) + B_z u_1(k) \\ x_\psi(k+1) &= A_\psi x_\psi(k) + B_\psi \tau_\psi(k) \\ x_{y\phi}(k+1) &= A_{y\phi} x_{y\phi}(k) + B_{y\phi} \tau_\phi(k) \\ x_{x\theta}(k+1) &= A_{x\theta} x_{x\theta}(k) + B_{x\theta} \tau_\theta(k) \end{aligned}$$

3.2. Control integral aplicado al modelo dinámico discreto

De lo abordado en el Apéndice A, se sabe que para considerar la acción integral, se define una nueva variable de estado $\dot{\sigma}$, la cual define el error de estado de la siguiente forma:

$$\dot{\sigma}(t) \triangleq e(t) = x(t) - x_R(t) \quad (3.13)$$

Donde $e(t)$ representa el error, $x(t)$ la variable de proceso y $x_R(t)$ la referencia fijada para dicha variable. Retomando T como el tiempo de muestreo y aplicando la aproximación de Euler se puede decir que:

$$\dot{\sigma}(t) \approx \frac{\sigma(k+1) - \sigma(k)}{T} \quad (3.14)$$

Sustituyendo la ecuación (3.14) en (3.13), se tiene:

$$\begin{aligned} \frac{\sigma(k+1) - \sigma(k)}{T} &= e(k) \\ \sigma(k+1) &= \sigma(k) + T e(k) \end{aligned}$$

Sustituyendo el error (3.13) en la ecuación anterior, ésta puede ser reescrita de la siguiente manera:

$$\sigma(k+1) = \sigma(k) + T x(k) - T x_R(t)$$

A continuación se procede a aplicar el algoritmo del control integral a cada uno de los subsistemas discretos (3.9)-(3.12) que representan al UAV. Para el subsistema relacionado con la altura ($z(k)$) del vehículo aéreo, se consideran las siguientes definiciones: $x_5(k) \triangleq x_{1z}(k)$ y $x_6(k) \triangleq x_{2z}(k)$, a partir de las definiciones anteriores dicho subsistema puede reescribirse como:

$$\begin{bmatrix} x_{1z}(k+1) \\ x_{2z}(k+1) \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix}}_{A_{z(k)}} \begin{bmatrix} x_{1z}(k) \\ x_{2z}(k) \end{bmatrix} + \underbrace{\begin{bmatrix} 0 \\ T \end{bmatrix}}_{B_{z(k)}} [u_1(k)]$$

Continuando con el proceso de síntesis de la acción integral se define $e_{1z}(k) \triangleq x_{1z}(k) - x_{1zR}(k)$ y $e_{2z}(k) \triangleq x_{2z}(k) - x_{2zR}(k)$ como los errores en los estados $x_{1z}(k)$ y $x_{2z}(k)$, donde $x_{1zR}(k)$ y $x_{2zR}(k)$ son las referencias dadas para $x_{1z}(k)$, $x_{2z}(k)$, respectivamente. Considerando $\sigma_{1z}(k+1) = e_{1z}(k)$ y $\sigma_{2z}(k+1) = e_{2z}(k)$ el vector del sistema aumentado para el subsistema $z(k)$ queda entonces definido por:

$$\xi_{z(k)}(k) = \begin{bmatrix} x_{1z}(k) \\ x_{2z}(k) \\ \sigma_{1z}(k) \\ \sigma_{2z}(k) \end{bmatrix}$$

Consecuentemente, la representación espacio estado para el sistema aumentado en $z(k)$ es:

$$\xi_{z(k)}(k+1) = \begin{bmatrix} x_{1z}(k+1) \\ x_{2z}(k+1) \\ \sigma_{1z}(k+1) \\ \sigma_{2z}(k+1) \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ T & 0 & 1 & 0 \\ 0 & T & 0 & 1 \end{bmatrix}}_{A_z} \begin{bmatrix} x_{1z}(k) \\ x_{2z}(k) \\ \sigma_{1z}(k) \\ \sigma_{2z}(k) \end{bmatrix} + \underbrace{\begin{bmatrix} 0 \\ T \\ 0 \\ 0 \end{bmatrix}}_{B_z} u_1(k) - \underbrace{\begin{bmatrix} 0 \\ 0 \\ x_{1zR}(k) \\ x_{2zR}(k) \end{bmatrix}}_{X_{Zref}}$$

$$\xi_{z(k)}(k+1) = \begin{bmatrix} A_{z(k)} & 0 \\ TC_{z(k)} & I_2 \end{bmatrix} \xi_{z(k)}(k) + \begin{bmatrix} B_{z(k)} \\ 0 \end{bmatrix} u_1(k)$$

Con respecto al subsistema relacionado con el ángulo de guiñada ψ , se tiene $x_{1\psi}(k) \triangleq x_{1\psi}(k)$ y $x_{2\psi}(k) \triangleq x_{2\psi}(k)$, por tanto el subsistema $\psi(k)$ está definido como:

$$\begin{bmatrix} x_{1\psi}(k+1) \\ x_{2\psi}(k+1) \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix}}_{A_{\psi(k)}} \begin{bmatrix} x_{1\psi}(k) \\ x_{2\psi}(k) \end{bmatrix} + \underbrace{\begin{bmatrix} 0 \\ T \end{bmatrix}}_{B_{\psi(k)}} [\tau_{\psi}(k)]$$

donde de manera similar al estado $z(k)$ se tiene que $e_{1\psi}(k) \triangleq x_{1\psi}(k) - x_{1\psi R}(k)$ y $e_{2\psi}(k) \triangleq x_{2\psi}(k) - x_{2\psi R}(k)$ son los errores en los estados $x_{1\psi}(k)$ y $x_{2\psi}(k)$, con $x_{1\psi R}(k)$ y $x_{2\psi R}(k)$ definiendo sus respectivas referencias. Nuevamente asignando $\sigma_{1\psi}(k+1) = e_{1\psi}(k)$ y $\sigma_{2\psi}(k+1) = e_{2\psi}(k)$, el vector de estado extendido para el subsistema $\psi(k)$ queda definido como:

$$\xi_{\psi(k)}(k) = \begin{bmatrix} x_{1\psi}(k) \\ x_{2\psi}(k) \\ \sigma_{1\psi}(k) \\ \sigma_{2\psi}(k) \end{bmatrix}$$

Siendo su correspondiente representa en espacio de estado la siguiente:

$$\xi_{\psi(k)}(k+1) = \begin{bmatrix} x_{1\psi}(k+1) \\ x_{2\psi}(k+1) \\ \sigma_{1\psi}(k+1) \\ \sigma_{2\psi}(k+1) \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ T & 0 & 1 & 0 \\ 0 & T & 0 & 1 \end{bmatrix}}_{A_{\psi}} \begin{bmatrix} x_{1\psi}(k) \\ x_{2\psi}(k) \\ \sigma_{1\psi}(k) \\ \sigma_{2\psi}(k) \end{bmatrix} + \underbrace{\begin{bmatrix} 0 \\ T \\ 0 \\ 0 \end{bmatrix}}_{B_{\psi}} \tau_{\psi}(k) - \underbrace{\begin{bmatrix} 0 \\ 0 \\ x_{1\psi R}(k) \\ x_{2\psi R}(k) \end{bmatrix}}_{x_{\psi ref}}$$

$$\xi_{\psi(k)}(k+1) = \begin{bmatrix} A_{\psi(k)} & 0 \\ TC_{\psi(k)} & I_2 \end{bmatrix} \xi_{\psi(k)}(k) + \begin{bmatrix} B_{\psi(k)} \\ 0 \end{bmatrix} \tau_{\psi}(k)$$

El procedimiento se aplica y desarrolla de manera análoga para los dos subsistemas restantes $y - \phi(k)$ y $x - \theta(k)$. Para el subsistema $y - \phi(k)$ se realizan las siguientes definiciones:

$$\begin{aligned} x_3(k) &\triangleq x_{1y}(k) \\ x_4(k) &\triangleq x_{2y}(k) \\ x_7(k) &\triangleq x_{3\phi}(k) \\ x_8(k) &\triangleq x_{4\phi}(k) \end{aligned}$$

de tal manera que al ser aplicadas se llega a la siguiente representación en espacio de estado para este subsistema.

$$\begin{bmatrix} x_{1y}(k+1) \\ x_{2y}(k+1) \\ x_{3\phi}(k+1) \\ x_{4\phi}(k+1) \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & gT & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{bmatrix}}_{A_{y\phi(k)}} \begin{bmatrix} x_{1y}(k) \\ x_{2y}(k) \\ x_{3\phi}(k) \\ x_{4\phi}(k) \end{bmatrix} + \underbrace{\begin{bmatrix} 0 \\ 0 \\ 0 \\ T \end{bmatrix}}_{B_{y\phi(k)}} [\tau_{\phi}(k)]$$

Definiendo ahora las variables del sistema aumentado y los errores, se tiene que:

$$\begin{aligned} \sigma_{1y}(k+1) &\triangleq e_{1y}(k) = x_{1y}(k) - x_{1yR}(k) \\ \sigma_{2y}(k+1) &\triangleq e_{2y}(k) = x_{2y}(k) - x_{2yR}(k) \\ \sigma_{3\phi}(k+1) &\triangleq e_{3\phi}(k) = x_{3\phi}(k) - x_{3\phi R}(k) \\ \sigma_{4\phi}(k+1) &\triangleq e_{4\phi}(k) = x_{4\phi}(k) - x_{4\phi R}(k) \end{aligned}$$

Donde $x_{1yR}(k)$ es la referencia para $x_{1y}(k)$, $x_{2yR}(k)$ es la referencia para $x_{2y}(k)$, $x_{3\phi R}(k)$ es la referencia para $x_{3\phi}(k)$ y $x_{4\phi R}(k)$ es la referencia para $x_{4\phi}(k)$. El vector del subsistema $y - \phi(k)$ aumentado está dado por:

$$\xi_{y\phi}(k) = \begin{bmatrix} x_{1y}(k) \\ x_{2y}(k) \\ x_{3\phi}(k) \\ x_{4\phi}(k) \\ \sigma_{1y}(k) \\ \sigma_{2y}(k) \\ \sigma_{3\phi}(k) \\ \sigma_{4\phi}(k) \end{bmatrix}$$

lo que lleva a la siguiente ecuación de estado representando en forma discreta la dinámica del subsistema $y - \phi(k)$:

$$\xi_{y\phi(k)}(k+1) = \begin{bmatrix} x_{1y}(k+1) \\ x_{2y}(k+1) \\ x_{3\phi}(k+1) \\ x_{4\phi}(k+1) \\ \sigma_{1y}(k+1) \\ \sigma_{2y}(k+1) \\ \sigma_{3\phi}(k+1) \\ \sigma_{4\phi}(k+1) \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & T & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & gT & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & T & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ T & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & T & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & T & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & T & 0 & 0 & 0 & 1 \end{bmatrix}}_{A_{y\phi}} \begin{bmatrix} x_{1y}(k) \\ x_{2y}(k) \\ x_{3\phi}(k) \\ x_{4\phi}(k) \\ \sigma_{1y}(k) \\ \sigma_{2y}(k) \\ \sigma_{3\phi}(k) \\ \sigma_{4\phi}(k) \end{bmatrix} + \underbrace{\begin{bmatrix} 0 \\ 0 \\ 0 \\ T \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}}_{B_{y\phi}} \tau_{\phi}(k) - \underbrace{\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ x_{1yR}(k) \\ x_{2yR}(k) \\ x_{3\phi R}(k) \\ x_{4\phi R}(k) \end{bmatrix}}_{x_{y\phi ref}}$$

$$\xi_{y\phi(k)}(k+1) = \begin{bmatrix} A_{y\phi(k)} & 0 \\ TC_{y\phi(k)} & I_4 \end{bmatrix} \xi_{y\phi(k)}(k) + \begin{bmatrix} B_{y\phi(k)} \\ 0 \end{bmatrix} \tau_{\phi}(k)$$

Finalmente para el subsistema $x - \theta(k)$ se proponen los siguientes asignaciones de variables:

$$\begin{aligned} x_1(k) &\triangleq x_{1x}(k) \\ x_2(k) &\triangleq x_{2x}(k) \\ x_9(k) &\triangleq x_{3\theta}(k) \\ x_{10}(k) &\triangleq x_{4\theta}(k) \end{aligned}$$

Lo cual implica que la representación espacio estado para este subsistema está definida como:

$$\begin{bmatrix} x_{1x}(k+1) \\ x_{2x}(k+1) \\ x_{3\theta}(k+1) \\ x_{4\theta}(k+1) \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & -gT & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{bmatrix}}_{A_{x\theta(k)}} \begin{bmatrix} x_{1x}(k) \\ x_{2x}(k) \\ x_{3\theta}(k) \\ x_{4\theta}(k) \end{bmatrix} + \underbrace{\begin{bmatrix} 0 \\ 0 \\ 0 \\ T \end{bmatrix}}_{B_{x\theta(k)}} [\tau_{\theta}(k)]$$

Las variables auxiliares y los errores para construir el sistema aumentado son:

$$\begin{aligned} \sigma_{1x} &\triangleq e_{1x}(k) = x_{1x}(k) - x_{1xR}(k) \\ \sigma_{2x} &\triangleq e_{2x}(k) = x_{2x}(k) - x_{2xR}(k) \\ \sigma_{3\theta} &\triangleq e_{3\theta}(k) = x_{3\theta}(k) - x_{3\theta R}(k) \\ \sigma_{4\theta} &\triangleq e_{4\theta}(k) = x_{4\theta}(k) - x_{4\theta R}(k) \end{aligned}$$

Donde $x_{1xR}(k)$ es la referencia para $x_{1x}(k)$, $x_{2xR}(k)$ es la referencia para $x_{2x}(k)$, $x_{3\theta R}(k)$ es la referencia para $x_{3\theta}(k)$ y $x_{4\theta R}(k)$ es la referencia para $x_{4\theta}(k)$. El vector del subsistema $x-\theta(k)$ aumentado está dado por:

$$\xi_{x\theta}(k) = \begin{bmatrix} x_{1x}(k) \\ x_{2x}(k) \\ x_{3\theta}(k) \\ x_{4\theta}(k) \\ \sigma_{1x}(k) \\ \sigma_{2x}(k) \\ \sigma_{3\theta}(k) \\ \sigma_{4\theta}(k) \end{bmatrix}$$

Claramente se puede deducir que la ecuación de estado asociada a esta asignación y aumento en dimensión del vector de estado es la siguiente:

$$\xi_{x\theta(k)}(k+1) = \begin{bmatrix} x_{1x}(k+1) \\ x_{2x}(k+1) \\ x_{3\theta}(k+1) \\ x_{4\theta}(k+1) \\ \sigma_{1x}(k+1) \\ \sigma_{2x}(k+1) \\ \sigma_{3\theta}(k+1) \\ \sigma_{4\theta}(k+1) \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & T & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & -gT & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & T & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ T & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & T & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & T & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & T & 0 & 0 & 0 & 1 \end{bmatrix}}_{\mathcal{A}_{x\theta}} \begin{bmatrix} x_{1x}(k) \\ x_{2x}(k) \\ x_{3\theta}(k) \\ x_{4\theta}(k) \\ \sigma_{1x}(k) \\ \sigma_{2x}(k) \\ \sigma_{3\theta}(k) \\ \sigma_{4\theta}(k) \end{bmatrix} + \underbrace{\begin{bmatrix} 0 \\ 0 \\ 0 \\ T \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}}_{\mathcal{B}_{x\theta}} \tau_{\theta}(k) - \underbrace{\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ x_{1xR}(k) \\ x_{2xR}(k) \\ x_{3\theta R}(k) \\ x_{4\theta R}(k) \end{bmatrix}}_{\mathcal{X}_{x\theta ref}}$$

$$\xi_{x\theta(k)}(k+1) = \begin{bmatrix} \mathcal{A}_{x\theta(k)} & 0 \\ T\mathcal{C}_{x\theta(k)} & I_4 \end{bmatrix} \xi_{x\theta(k)}(k) + \begin{bmatrix} \mathcal{B}_{x\theta(k)} \\ 0 \end{bmatrix} \tau_{\theta}(k)$$

Los controles óptimos $u_1(k)$, $\tau_{\psi}(k)$, $\tau_{\phi}(k)$ y $\tau_{\theta}(k)$ han sido hallados utilizando la función e índice de desempeño asociado a cada uno de los subsistemas aumentados. Es claro que para que estos cuatro controles puedan ser aplicados es necesario verificar la controlabilidad de cada subsistema. Por lo tanto, se procede de la siguiente forma: si el par $\mathcal{A}_z(k), \mathcal{B}_z(k)$ es controlable y $\begin{bmatrix} \mathcal{A}_z(k) & \mathcal{B}_z(k) \\ \mathcal{C}_z(k) & 0 \end{bmatrix}$, es de rango $2 + 1$, donde 2 es el número de variables de estado y 1 es el número de controles, entonces $(\mathcal{A}_z, \mathcal{B}_z)$ es controlable. Suponiendo que esto se cumple, se puede calcular el control óptimo $u_1^*(k)$ está definido como:

$$u_1^*(k) = -(\mathcal{R}_z + \mathcal{B}_z^T \mathcal{P}_z \mathcal{B}_z)^{-1} \mathcal{B}_z^T \mathcal{P}_z \mathcal{A}_z x_z^*(k)$$

sujeto al índice de desempeño dado por la siguiente ecuación.

$$\mathcal{J}_z = \sum_{k=1}^{\infty} (x_z^T(k) \mathcal{Q}_z x_z(k) + u_1^2(k) \mathcal{R}_z)$$

donde $\mathcal{Q}_z \geq 0$, $\mathcal{R}_z > 0$ y \mathcal{P}_z es la solución a la ecuación algebraica del tipo de Riccati asociada definida como:

$$\mathcal{P}_z = \mathcal{A}_z^T \mathcal{P}_z \mathcal{A}_z - \mathcal{A}_z^T \mathcal{P}_z \mathcal{B}_z (\mathcal{R}_z + \mathcal{B}_z^T \mathcal{P}_z \mathcal{B}_z)^{-1} \mathcal{B}_z^T + \mathcal{Q}_z$$

De forma similar si el par $A_\psi(k), B_\psi(k)$ es controlable y $\begin{bmatrix} A_\psi(k) & B_\psi(k) \\ C_\psi(k) & 0 \end{bmatrix}$, es de rango $2+1$, donde 2 es el número de variables de estado y 1 es el número de controles, para este subsistema, entonces $(\mathcal{A}_\psi, \mathcal{B}_\psi)$ es controlable, si es así el control óptimo para el ángulo de guiñada ψ del UAV es:

$$\tau_\psi^*(k) = -(\mathcal{R}_\psi + \mathcal{B}_\psi^T \mathcal{P}_\psi \mathcal{B}_\psi)^{-1} \mathcal{B}_\psi^T \mathcal{P}_\psi \mathcal{A}_\psi x_\psi^*(k)$$

con índice de desempeño \mathcal{J}_ψ definido por:

$$\mathcal{J}_\psi = \sum_{k=1}^{\infty} (x_\psi^T(k) \mathcal{Q}_\psi x_\psi(k) + \tau_\psi^2(k) \mathcal{R}_\psi)$$

Donde $\mathcal{Q}_\psi \geq 0$, $\mathcal{R}_\psi > 0$ y \mathcal{P}_ψ es la solución a la ecuación algebraica del tipo de Riccati para este subsistema y definida como:

$$\mathcal{P}_\psi = \mathcal{A}_\psi^T \mathcal{P}_\psi \mathcal{A}_\psi - \mathcal{A}_\psi^T \mathcal{P}_\psi \mathcal{B}_\psi (\mathcal{R}_\psi + \mathcal{B}_\psi^T \mathcal{P}_\psi \mathcal{B}_\psi)^{-1} \mathcal{B}_\psi^T + \mathcal{Q}_\psi$$

Trabajando ahora con el subsistema $y\text{-}\phi(k)$ se tiene que si el par $A_{y\phi}(k), B_{y\phi}(k)$ es controlable y $\begin{bmatrix} A_{y\phi}(k) & B_{y\phi}(k) \\ C_{y\phi}(k) & 0 \end{bmatrix}$, es de rango $4 + 1$, donde 4 es el número de variables de estado y 1 es el número de controles, entonces $(\mathcal{A}_{y\phi}, \mathcal{B}_{y\phi})$ es controlable, lo que implica que el control óptimo para el alabeo del UAV es $\tau_{y\phi}^*(k)$ es:

$$\tau_{y\phi}^*(k) = -(\mathcal{R}_{y\phi} + \mathcal{B}_{y\phi}^T \mathcal{P}_{y\phi} \mathcal{B}_{y\phi})^{-1} \mathcal{B}_{y\phi}^T \mathcal{P}_{y\phi} \mathcal{A}_{y\phi} x_{y\phi}^*(k)$$

Sujeto al índice de desempeño $\mathcal{J}_{y\phi}$ mostrado en la siguiente ecuación:

$$\mathcal{J}_{y\phi} = \sum_{k=1}^{\infty} (x_{y\phi}^T(k) \mathcal{Q}_{y\phi} x_{y\phi}(k) + \tau_{y\phi}^2(k) \mathcal{R}_{y\phi})$$

Donde $\mathcal{Q}_{y\phi} \geq 0$, $\mathcal{R}_{y\phi} > 0$ y $\mathcal{P}_{y\phi}$ es la solución a la ecuación algebraica del tipo de Riccati, dada por:

$$\mathcal{P}_{y\phi} = \mathcal{A}_{y\phi}^T \mathcal{P}_{y\phi} \mathcal{A}_{y\phi} - \mathcal{A}_{y\phi}^T \mathcal{P}_{y\phi} \mathcal{B}_{y\phi} (\mathcal{R}_{y\phi} + \mathcal{B}_{y\phi}^T \mathcal{P}_{y\phi} \mathcal{B}_{y\phi})^{-1} \mathcal{B}_{y\phi}^T + \mathcal{Q}_{y\phi}$$

Finalmente para el subsistema $x\text{-}\theta(k)$ de igual manera, si el par $A_{x\theta}(k), B_{x\theta}(k)$ es controlable y $\begin{bmatrix} A_{x\theta}(k) & B_{x\theta}(k) \\ C_{x\theta}(k) & 0 \end{bmatrix}$, es de rango $4 + 1$, donde 4 es el número de variables de estado y 1 es el número de controles, entonces $(\mathcal{A}_{x\theta}, \mathcal{B}_{x\theta})$ es controlable, lo que implica que el control óptimo para el ángulo de cabeceo θ del UAV es:

$$\tau_{x\theta}^*(k) = -(\mathcal{R}_{x\theta} + \mathcal{B}_{x\theta}^T \mathcal{P}_{x\theta} \mathcal{B}_{x\theta})^{-1} \mathcal{B}_{x\theta}^T \mathcal{P}_{x\theta} \mathcal{A}_{x\theta} x_{x\theta}^*(k)$$

con índice de desempeño $\mathcal{J}_{x\theta}$ definido como:

$$\mathcal{J}_{x\theta} = \sum_{k=1}^{\infty} (x_{x\theta}^T(k) \mathcal{Q}_{x\theta} x_{x\theta}(k) + \tau_{x\theta}^2(k) \mathcal{R}_{x\theta})$$

Donde $\mathcal{Q}_{x\theta} \geq 0$, $\mathcal{R}_{x\theta} > 0$ y $\mathcal{P}_{x\theta}$ es la solución a la ecuación algebraica del tipo de Riccati dada por:

$$\mathcal{P}_{x\theta} = \mathcal{A}_{x\theta}^T \mathcal{P}_{x\theta} \mathcal{A}_{x\theta} - \mathcal{A}_{x\theta}^T \mathcal{P}_{x\theta} \mathcal{B}_{x\theta} (\mathcal{R}_{x\theta} + \mathcal{B}_{x\theta}^T \mathcal{P}_{x\theta} \mathcal{B}_{x\theta})^{-1} \mathcal{B}_{x\theta}^T + \mathcal{Q}_{x\theta}$$

3.3. Conclusiones y Comentarios

En este Capítulo se presentó de una manera detallada la estrategia de control empleada, la cual consiste en dividir el sistema en cuatro subsistemas, agregando una acción integral, dicha estrategia, ha sido sintetizada directamente de cada uno de los subsistemas discretizados. Se muestra el algoritmo aplicado a dichos subsistemas para la síntesis del control, tanto en tiempo continuo como en discreto. Además se expresan los controles óptimos hallados para cada subsistema en función de su representación espacio estado, la cual es obtenida al aplicar inicialmente una linealización exacta.

Capítulo 4

En este capítulo son mostrados los resultados de simulación y experimentales obtenidos durante la realización del presente trabajo de tesis. Primero es mostrada la simulación la cual permitió verificar la viabilidad de aplicar a un vehículo aéreo de cuatro rotores el algoritmo de control propuesto en el Capítulo 3. A continuación se muestran los resultados que arrojó el uso de un controlador PID en la planta, de igual forma los resultados obtenidos cuando se aplicó el control óptimo integral. El utilizar un control PID es para establecer un criterio de comparación entre este controlador y el algoritmo de control óptimo con acción integral, el criterio comparativo utilizado es el consumo de energético medido en un total de cinco experimentos para cada caso y promediando los resultados obtenidos. Lo anterior se hizo con el fin de comprobar el porcentaje de ahorro de energía presentado por el control integral óptimo respecto a un PID. Este último fue sintonizado empíricamente lo que implicó una ardua tarea, mientras que el control óptimo se sintonizó mediante la resolución numérica de las ecuaciones algebraicas del tipo de Riccati empleando el código mostrado en el Apéndice D. Al final las conclusiones generales del trabajo son abordadas.

4. Resultados

4.1. Resultados de simulación

La simulación del sistema se hace en el programa SIMULINK de MATLAB 2013 utilizando la función "S-function". Para realizar las tareas de simulación se construyó el modelo dinámico del robot aéreo utilizando los bloques disponibles en SIMULINK, el resultado es mostrado en la figura 9. Así también, las Ecuaciones Algebraicas de Riccati son resueltas en MATLAB utilizando el comando DARE (Discrete Algebraic Riccati Equation), lo cual puede ser consultado dentro del programa desarrollado y que se muestra en el Apéndice D. Para simular el vuelo del UAV se emplea un tiempo de muestreo $T = 10$ ms, la referencia se fija en: $x_{ref}(t) = 10$ cm, $y_{ref}(t) = 20$ cm, $z_{ref}(t) = 30$ cm, $\theta_{ref}(t) = \phi_{ref}(t) = \psi_{ref}(t) = 0^\circ$. Las matrices de penalización utilizadas para el control y el estado son las siguientes:

$$\mathcal{Q}_z = \mathcal{Q}_\psi = \begin{bmatrix} 1000 & 0 & 0 & 0 \\ 0 & 1000 & 0 & 0 \\ 0 & 0 & 1000 & 0 \\ 0 & 0 & 0 & 1000 \end{bmatrix}$$

$$\mathcal{R}_z = \mathcal{R}_\psi = 1$$

$$Q_{y\phi} = Q_{x\theta} = \begin{bmatrix} 1000 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1000 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1000 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1000 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1000 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1000 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1000 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1000 \end{bmatrix}$$

$$R_{y\phi} = R_{x\phi} = [1]$$

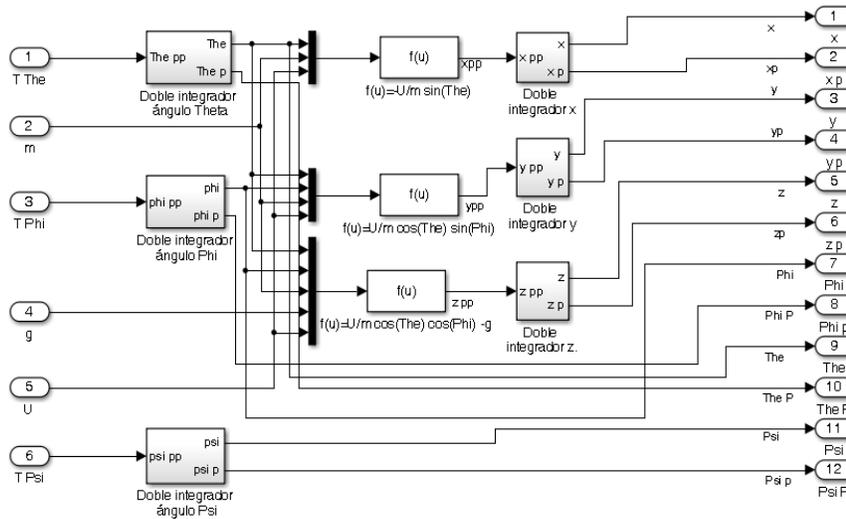


Figura 9: Diagrama a bloques del modelo del UAV cargado en SIMULINK.

En la figura 10 se muestra el esquema de control cargado al simulador, los bloques de ganancias son hallados mediante la ejecución del código mostrado en el Apéndice D. Como puede apreciarse en la figura mencionada, existe un bloque denominado “cuadrirrotor” dentro del cual se encuentra el diagrama ilustrado en la figura 9, de dicho bloque se obtienen las 12 variables de estado del sistema. Como se explicó en los capítulos precedentes, éstas son divididas en cuatro subsistemas y se aplican las ganancias óptimas halladas para el controlador de cada subsistema. Es así que se obtienen los 4 controles óptimos que sirven para cerrar el lazo en el modelo del UAV. Todo lo anterior se realiza una vez establecidas cada una de las referencias que son mostradas en los bloques “SP”.

4 RESULTADOS

Las gráficas de los resultados generados por la simulación son mostradas de la figura 11 a la figura 18. Pueden observarse los comportamientos de las variables que definen las 3 posiciones y las 3 velocidades traslacionales y angulares del UAV. Todas ellas alcanzan la respectiva referencia previamente establecida. Así también, se observa el comportamiento de los cuatro controles aplicados para la manipulación del robot. Es de resaltar que la trayectoria seguida por el UAV es libre, esto en el sentido que no obedece a alguna previamente fijada y es sólo el resultado de intentar alcanzar el estado deseado.

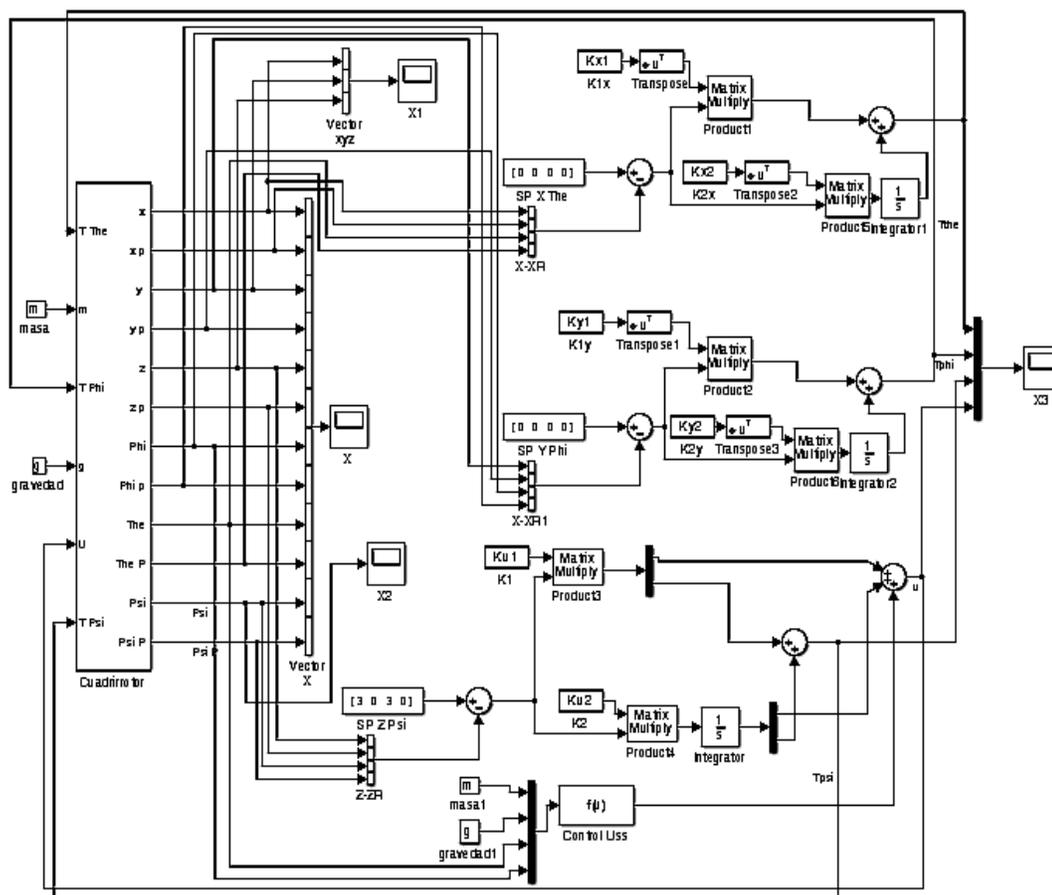


Figura 10: Esquema de control en SIMULINK.

4 RESULTADOS

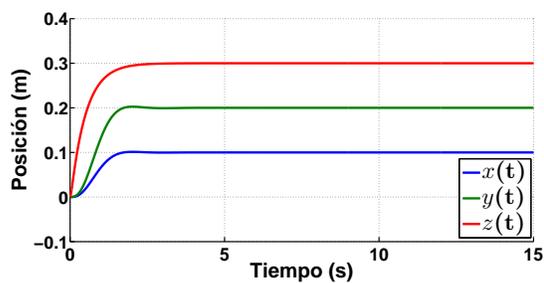


Figura 11: Posiciones $x(t)$, $y(t)$, $z(t)$ simuladas.

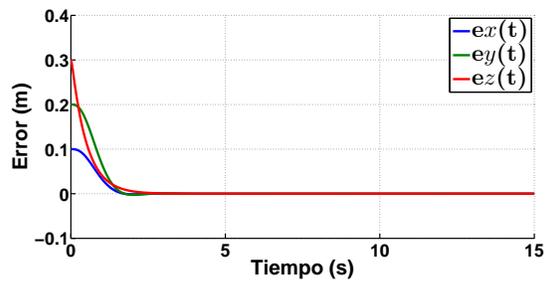


Figura 12: Errores en posición $x(t)$, $y(t)$, $z(t)$ simulados.

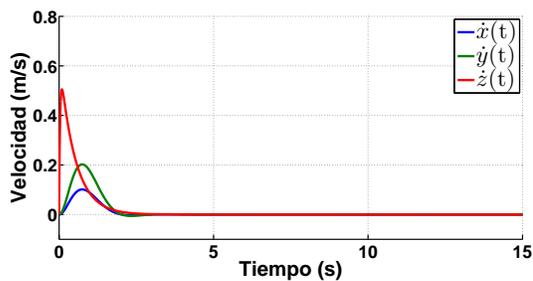


Figura 13: Velocidades en $x(t)$, $y(t)$, $z(t)$ simuladas.

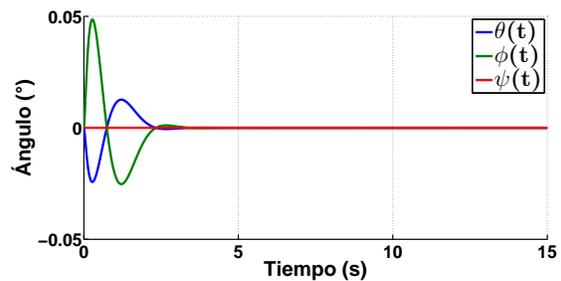


Figura 14: Posiciones angulares $\theta(t)$, $\phi(t)$, $\psi(t)$ simuladas.

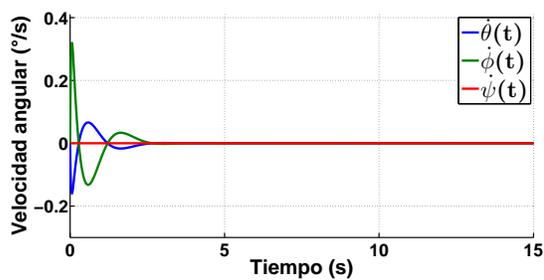


Figura 15: Velocidades angulares $\dot{\theta}(t)$, $\dot{\phi}(t)$, $\dot{\psi}(t)$ simuladas.

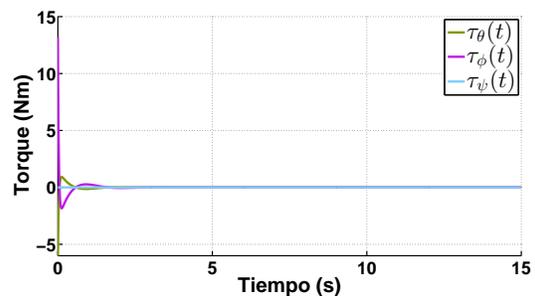


Figura 16: Señales de control óptimas $\tau_\theta(t)$, $\tau_\phi(t)$, $\tau_\psi(t)$ simuladas.

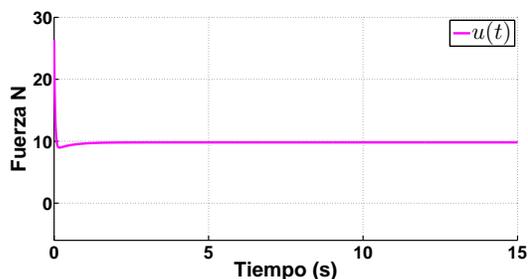


Figura 17: Control óptimo $u(t)$ simulado.

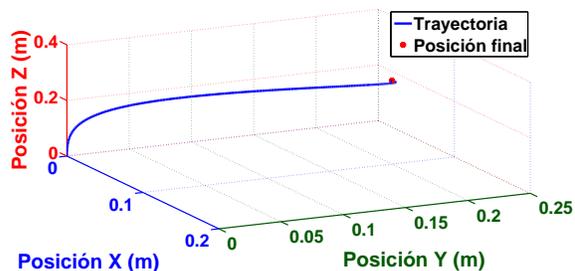


Figura 18: Trayectoria del UAV en 3D simulada.

4.2. Controlador PID

En esta subsección se presentan los resultados obtenidos al aplicar el controlador PID sintonizado heurísticamente. El algoritmo usado es el mostrado en el Apéndice A dado por la ecuación (A.19). El tiempo de muestreo usado es $T = 50\text{ms}$, el set point es fijado en $x(t)$ y $y(t)$: $x_{ref}(t) = y_{ref}(t) = 0\text{cm}$. Para la dinámica $z(t)$ la referencia es variante en el tiempo en el intervalo $0 \leq t < 5\text{s}$ se tiene $z_{ref} = 0\text{cm}$, para $t \geq 5\text{s}$ se tiene $z_{ref} = t * 0.45\text{cm}$ saturado a 30cm , es decir una vez alcanzada esta altura el valor deseado permanece constante. Para $\theta(t)$, $\phi(t)$, $\psi(t)$: $\theta_{ref}(t) = \phi_{ref}(t) = \psi_{ref}(t) = 0^\circ$. Se realizaron 5 experimentos el tiempo de vuelo es de 180 segundos para cada uno de ellos, de la figura 19 a la figura 34 se muestran los resultados para uno de los experimentos realizados al emplear el controlador PID en el UAV. Para aplicar el controlador PID el sistema fue dividido en 4 subsistemas como se indica en el Capítulo 3.

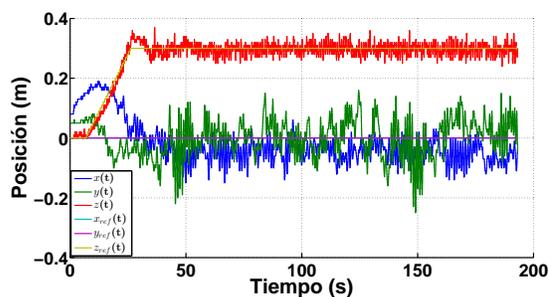


Figura 19: Posiciones y referencias $x(t)$, $y(t)$, $z(t)$ aplicando PID.

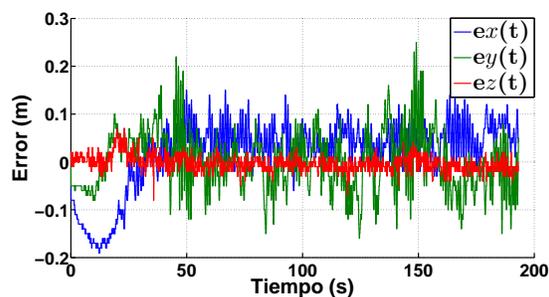


Figura 20: Errores en posición $x(t)$, $y(t)$, $z(t)$ aplicando PID.

4 RESULTADOS

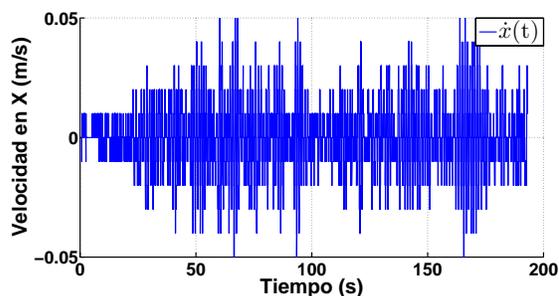


Figura 21: Velocidad en $x(t)$ aplicando PID.

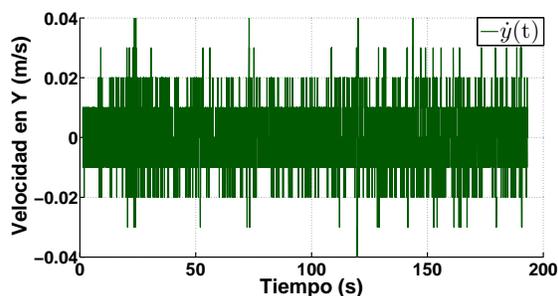


Figura 22: Velocidad en $y(t)$ aplicando PID.

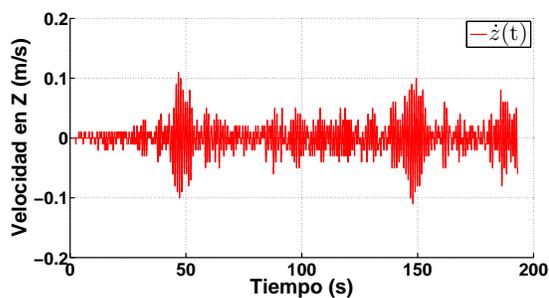


Figura 23: Velocidad en $z(t)$ aplicando PID.

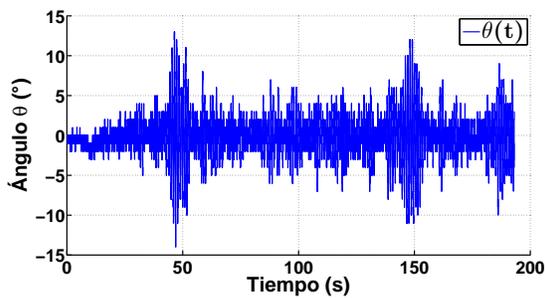


Figura 24: Posición angular $\theta(t)$ aplicando PID.

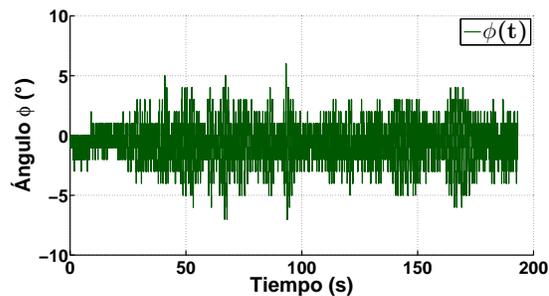


Figura 25: Posición angular $\phi(t)$ aplicando PID.

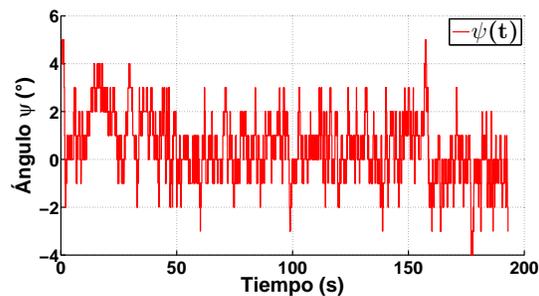


Figura 26: Posición angular $\psi(t)$ aplicando PID.

4 RESULTADOS

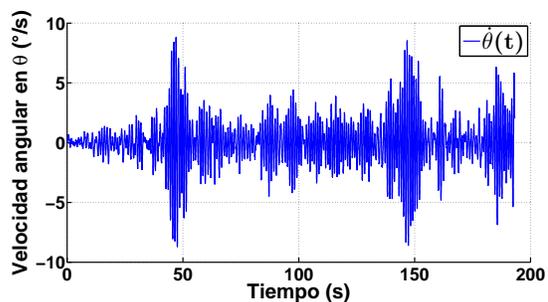


Figura 27: Velocidad angular $\dot{\theta}(t)$ aplicando PID.

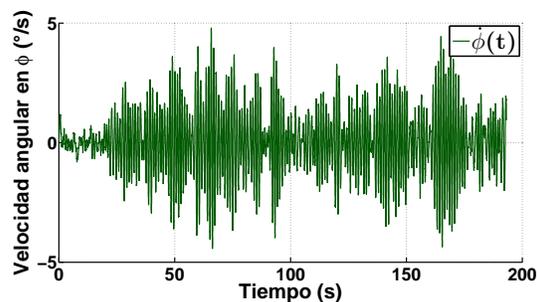


Figura 28: Velocidad angular $\dot{\phi}(t)$ aplicando PID.

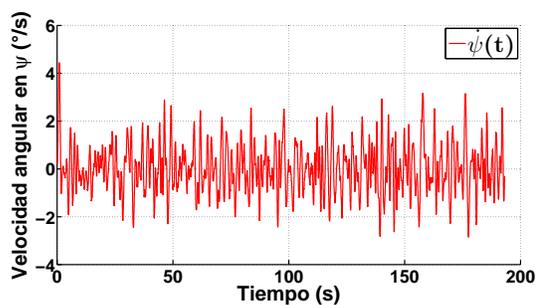


Figura 29: Velocidad angular $\dot{\psi}(t)$ aplicando PID.

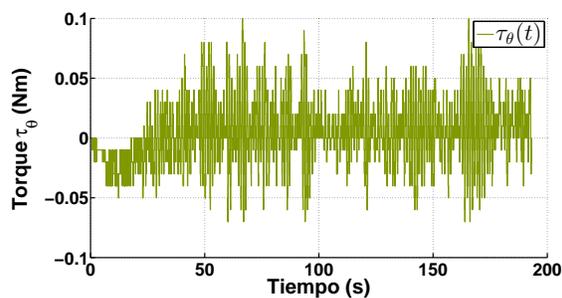


Figura 30: Señal de control $\tau_{\theta}(t)$ aplicando PID.

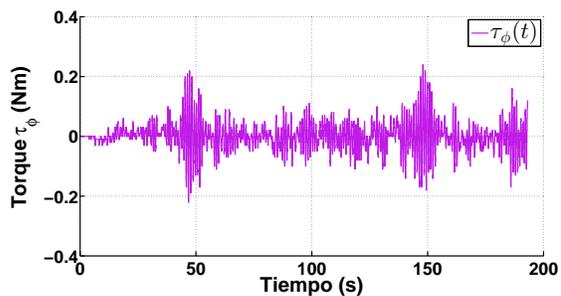


Figura 31: Señal de control $\tau_{\phi}(t)$ aplicando PID.

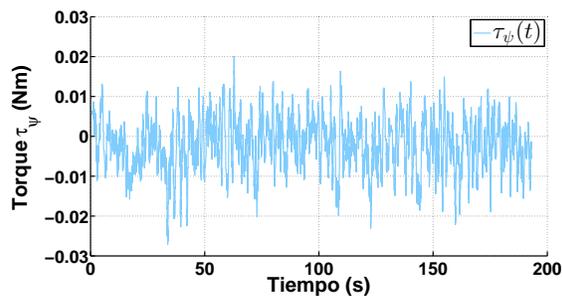


Figura 32: Señal de control $\tau_{\psi}(t)$ aplicando PID.

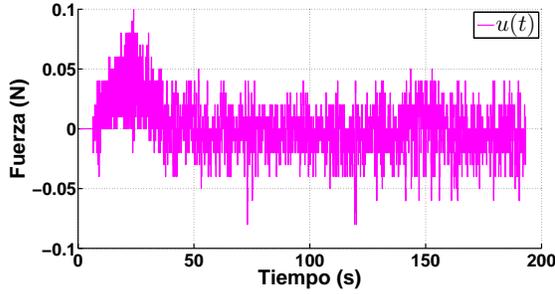


Figura 33: Señal de control $u(t)$ aplicando PID.

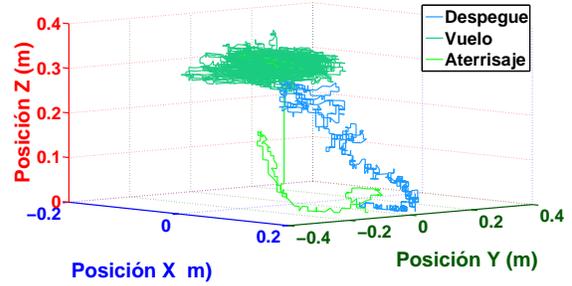


Figura 34: Trayectoria del UAV en 3D aplicando PID.

4.3. Controlador Óptimo Integral

Al igual que en los experimentos para el caso de la aplicación del controlador PID, también para el control óptimo integral se realizaron 5 experimentos, el tiempo de muestreo usado para calcular los controles óptimos fue seleccionado en $T = 50\text{ms}$, y la referencia fue establecida en $x(t)$ y $y(t)$: $x_{ref}(t) = y_{ref}(t) = 0\text{cm}$, mientras que para la dinámica $z(t)$ su referencia es variante en el tiempo, en el intervalo $0 \leq t < 5\text{s}$: $z_{ref} = 0\text{cm}$, con comportamiento regido por la función $z_{ref} = t * 0.45\text{cm}$ para $t \geq 5\text{s}$, hasta alcanzar 30cm donde es saturado. Es decir para $t \geq 5\text{s}$ la altura permanece constante en 30cm . Para $\theta(t)$, $\phi(t)$, $\psi(t)$: $\theta_{ref}(t) = \phi_{ref}(t) = \psi_{ref}(t) = 0^\circ$. Las matrices usadas para penalizar los controles están definidas como:

$$Q_z = \begin{bmatrix} 50 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}; \quad \mathcal{R}_z = 490;$$

$$Q_\psi = \begin{bmatrix} 18 & 0 & 0 & 0 \\ 0 & 171 & 0 & 0 \\ 0 & 0 & 0.1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}; \quad \mathcal{R}_\psi = 180;$$

$$Q_{y\phi} = Q_{x\theta} = \begin{bmatrix} 9 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 9 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 \times 10^{-6} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \times 10^{-7} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 4 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 10 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \times 10^{-5} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \times 10^{-5} \end{bmatrix}; \quad \begin{matrix} \mathcal{R}_{x\theta} = 40; \\ \mathcal{R}_{y\phi} = 31; \end{matrix}$$

Los Resultados experimentales obtenidos al aplicar el control óptimo de horizonte infinito, para una de las 5 pruebas realizadas, son mostrados en las figuras (35 a la 50). Como puede apreciarse el sistema es estable en toda la región de operación, lo que hace al control adecuado para la tarea de estabilización encomendada, pero además los resultados obtenidos son muy aceptables con respecto al comportamiento dinámico de la plataforma.

4 RESULTADOS

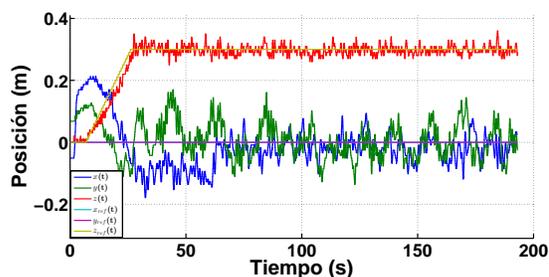


Figura 35: Posiciones y referencias $x(t)$, $y(t)$, $z(t)$ al aplicar control óptimo integral.

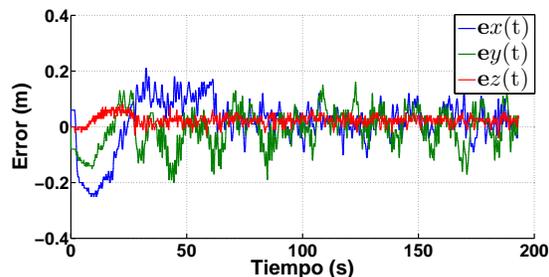


Figura 36: Errores en posición $x(t)$, $y(t)$, $z(t)$ al aplicar control óptimo integral.

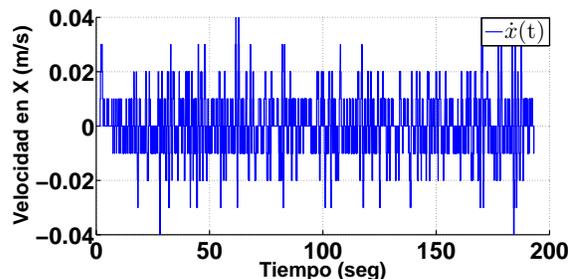


Figura 37: Velocidad en $x(t)$ al aplicar control óptimo integral.

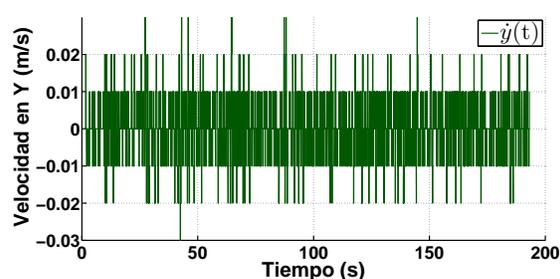


Figura 38: Velocidad en $y(t)$ al aplicar control óptimo integral.

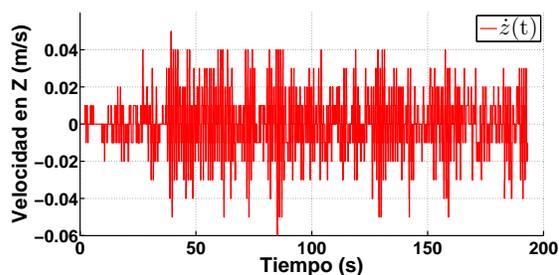


Figura 39: Velocidad en $z(t)$ al aplicar control óptimo integral.

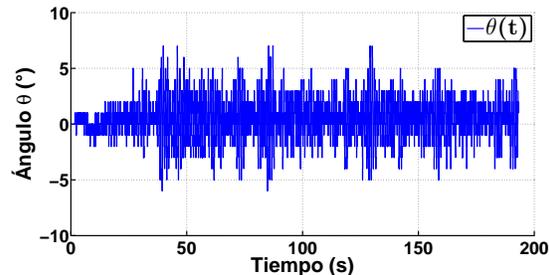


Figura 40: Posición angular $\theta(t)$ al aplicar control óptimo integral.

4 RESULTADOS

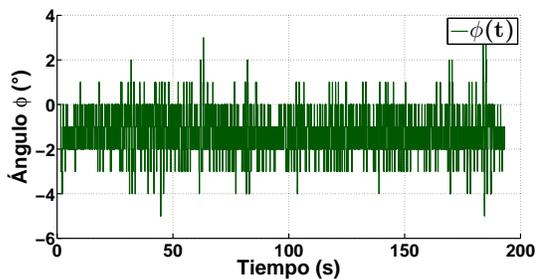


Figura 41: Posición angular $\phi(t)$ al aplicar control óptimo integral.

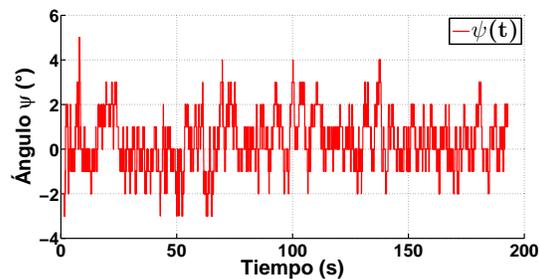


Figura 42: Posición angular $\psi(t)$ al aplicar control óptimo integral.

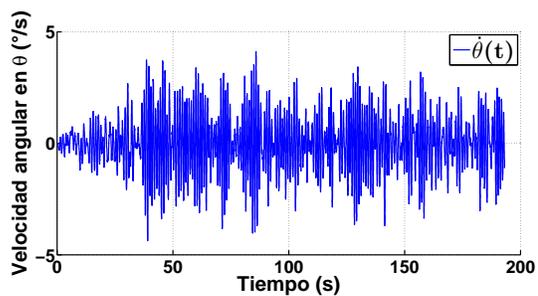


Figura 43: Velocidad angular $\dot{\theta}(t)$ al aplicar control óptimo integral.

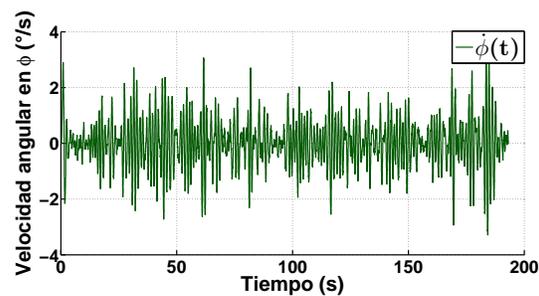


Figura 44: Velocidad angular $\dot{\phi}(t)$ al aplicar control óptimo integral.

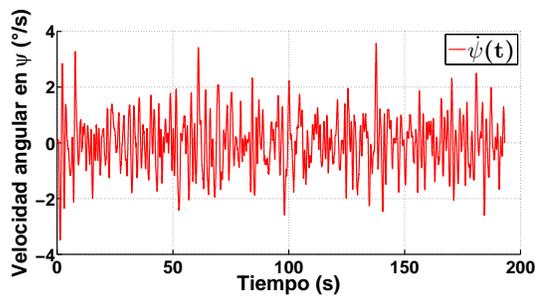


Figura 45: Velocidad angular $\dot{\psi}(t)$ al aplicar control óptimo integral.

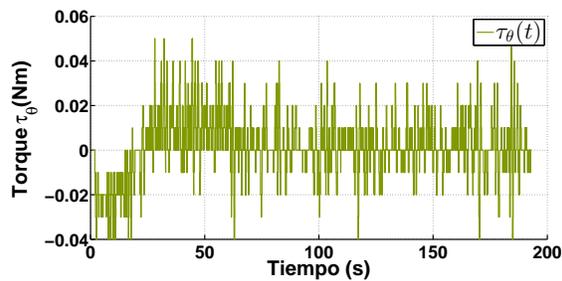


Figura 46: Señal de control $\tau_\theta(t)$ al aplicar control óptimo integral.

4 RESULTADOS

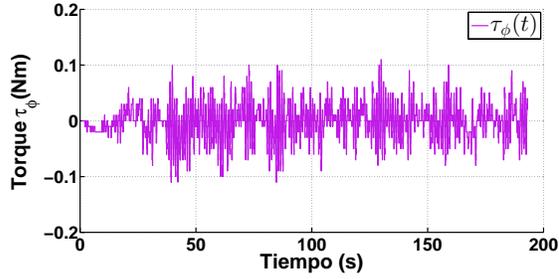


Figura 47: Señal de control $\tau_\phi(t)$ al aplicar control óptimo integral.

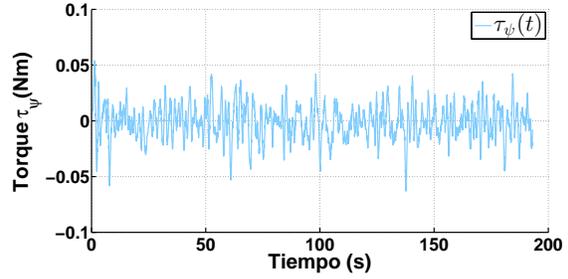


Figura 48: Señal de control $\tau_\psi(t)$ al aplicar control óptimo integral.

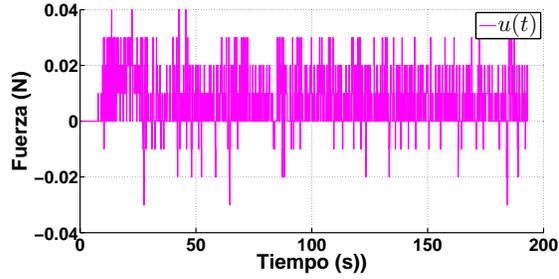


Figura 49: Señal de control $u(t)$ al aplicar control óptimo integral.

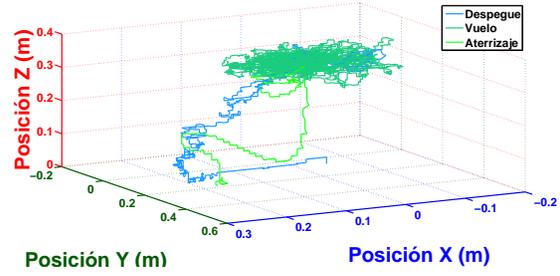


Figura 50: Trayectoria del UAV en 3D al aplicar control óptimo integral.

4.4. Comparación entre controlador PID y controlador óptimo integral

La comparación de porcentaje de ahorro de energía entre el controlador PID y el controlador óptimo integral de horizonte infinito es hallada en función de la integración de los índices de desempeño para cada uno de los 4 subsistema dinámicos, lo anterior es mostrado en la tabla 1.

Índice de desempeño	PID	Integral	Ahorro PID %	Ahorro Integral %
$\mathcal{J}_{ \tau_\theta }$	10849	4026	—	62.89
$\mathcal{J}_{ \tau_\phi }$	1545	3966	61.04	—
$\mathcal{J}_{ \tau_\psi }$	6238.80	3847.85	—	38.32
$\mathcal{J}_{ u }$	3930.01	2028.20	—	48.39

Tabla 1: Tabla de comparación de índices de desempeño entre el controlador PID y el controlador óptimo integral.

4 RESULTADOS

Adicionalmente, para obtener otra característica comparativa entre los controladores aplicados se ha empleado el sensor de corriente ACS712 acoplado a un microcontrolador PIC18F4550 de Microchip (para más detalles ver Apéndice E). Al utilizar este sensor se es capaz de medir la corriente consumida por los motores que proporcionan sustentación al helicóptero (ver figura 51). Los datos adquiridos fueron guardados en SIMULINK de MATLAB, durante las 5 ocasiones en que se desarrollaron los experimentos tanto para el controlador PID y el controlador óptimo integral de horizonte infinito. Los resultados obtenidos son mostrados en las figuras 52-53 (figura 52 para el control integral óptimo y figura 53 para el controlador PID).

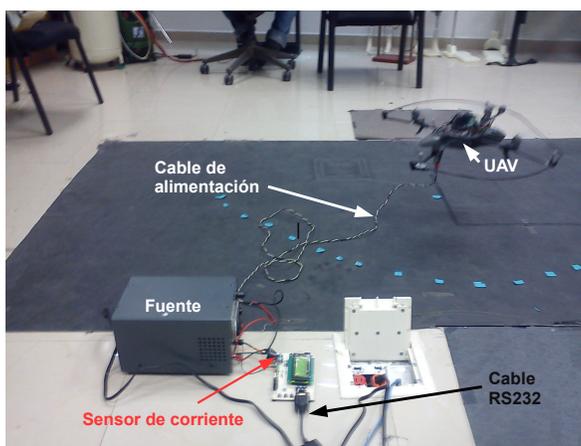


Figura 51: UAV Conectado al sensor de corriente.

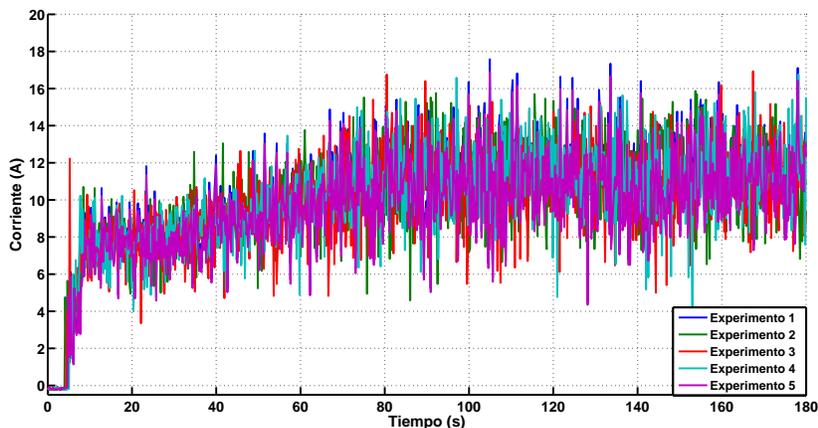


Figura 52: Señales de corriente al aplicar el control integral óptimo de horizonte infinito.

4 RESULTADOS

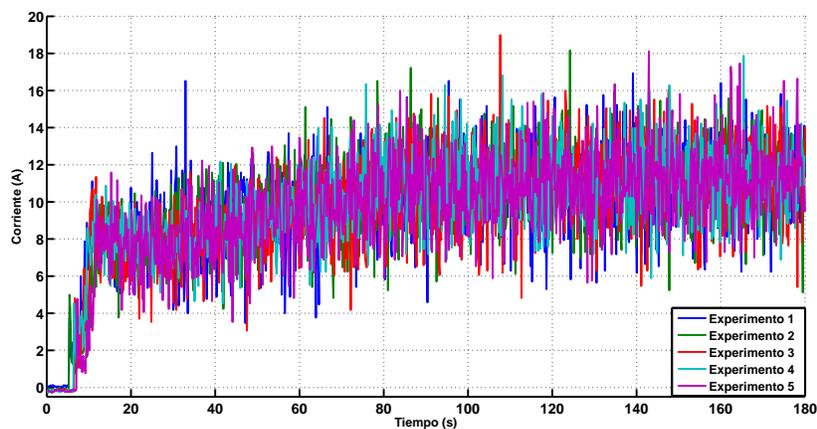


Figura 53: Señales de corriente al aplicar el control PID.

Para determinar el tiempo de adquisición de los datos referentes al consumo energético se realizó el siguiente procedimiento, se desactiva durante 100ms una salida del microcontrolador y solamente se activa durante la adquisición y envío de datos, realizando lo anterior y con la ayuda de un osciloscopio DSO-2090 el tiempo de adquisición se estimó en 98ms (como se muestra en la figura 54). Con el dato anterior y sabiendo que el UAV fue alimentado a 11.5V, se procede a calcular la potencia eléctrica. La tabla 2 muestra la potencia consumida por cada controlador durante los experimentos realizados.

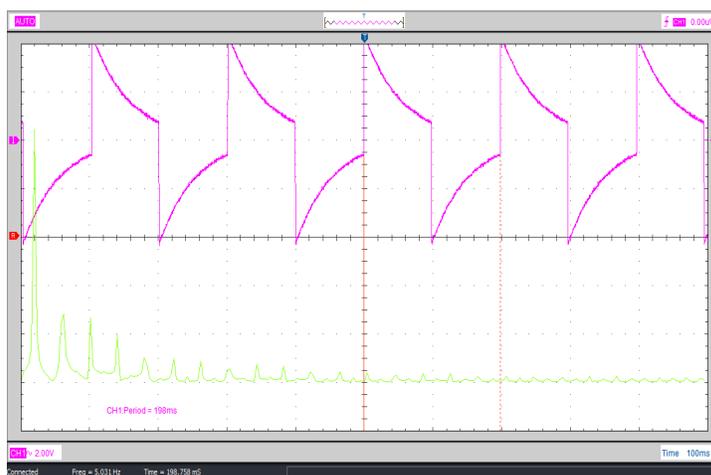


Figura 54: Señal PWM que permite saber el tiempo exacto de adquisición empleado.

4 RESULTADOS

N. de Prueba	Controlador PID P(Watts)	C. Óptimo Integral P(Watts)
1	11240.57538	10881.30559
2	11277.81777	10980.20998
3	11105.32321	10917.09009
4	11272.88084	10999.95773
5	11530.29733	10915.11559

Tabla 2: Tabla de consumo de energía por cada experimento realizado.

Obteniendo la media de los consumos energéticos estimados en cada uno de los experimentos realizados para cada uno de los 2 controladores, se puede establecer una comparación del ahorro de energía real del controlador óptimo integral con respecto al controlador PID. Dicho análisis es mostrado en la tabla 3.

Control	P. promedio (Watts)	Ahorro %
C. PID	11285.37891	-
C. Óptimo Integral	10938.73580	3.071612514

Tabla 3: Tabla de ahorro de energía enfoque consumo eléctrico.

A continuación es presentado un tercer análisis el cual está basado en el enfoque de energía cinética y potencial, para los 5 experimentos realizados se hace el cálculo de la energía total en función de los valores medidos y se obtiene el promedio. El consumo de energía es mostrado en la figura (55), en la tabla 4 se muestra el ahorro energético obtenido desde esta perspectiva.

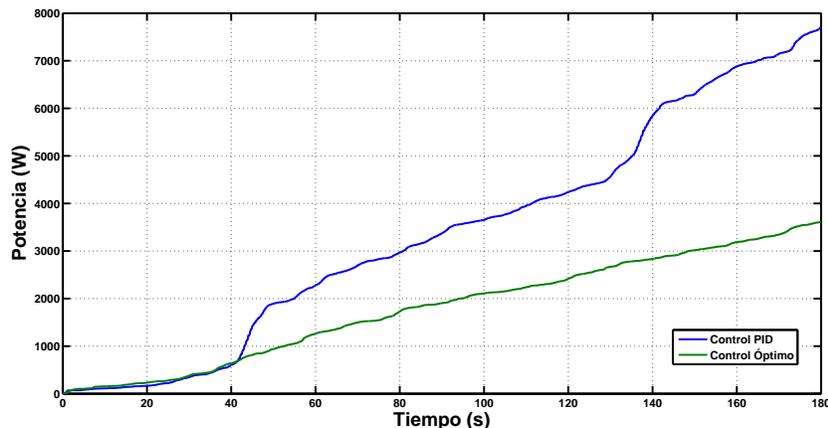


Figura 55: Potencia consumida por los dos controladores desde el enfoque energía cinética y potencial.

Control	P. promedio (Watts)	Ahorro %
C. PID	7721.82443	-
C. Óptimo Integral	3608.97545	53.26265

Tabla 4: Tabla de ahorro de energía desde el enfoque energía cinética y potencial.

4.5. Conclusión General.

Con los resultados obtenidos se demuestra que las técnicas de control óptimo avanzadas, como es el caso que se está abordando que además tiene agregado una acción integral permiten ahorrar energía sin tener que buscar heurísticamente las ganancias del controlador. Este no es el caso del controlador PID, ya que no existe un método directo para sintonizarlo sobre sistemas no lineales, y el hallar sus ganancias que le permitan operar correctamente puede resultar una tarea muy complicada, laboriosa y en ocasiones con un éxito muy limitado.

Al analizar el ahorro de energía comparando los índices de desempeño, éste pareciera ser considerablemente grande, pero al realizar las mediciones en la potencia consumida por el UAV, y de acuerdo a los resultados obtenidos el ahorro no es muy significativo, sin embargo este ahorro existe. Considerando un vuelo promedio estándar de 20min para un UAV de talla pequeña y aplicando este algoritmo se incrementaría el tiempo de vuelo 36.95s más. El análisis de energía desde el enfoque cinética-potencial muestra un ahorro del más del 50 % sin embargo el análisis de energía desde la perspectiva de consumo eléctrico denota que esto no es un valor efectivo, lo anterior se debe a que en el modelo no se contempla el consumo energético del controlador electrónico (microprocesador) y los demás componentes que forman parte del UAV (motores, regulador, etc.), lo que demuestra que el mayor consumo energético se concentra en dichos elementos.

Es importante resaltar que la linealización exacta aplicada permite garantizar la estabilidad del sistema no sólo en una región de operación sino en toda la trayectoria del sistema como se demuestra en [22]. Además, se demuestra que el controlador óptimo es lo suficientemente robusto ya que al aplicar perturbación al UAV se observó que el control integral es capaz de mantenerlo en una zona de operación, absorbiendo las perturbaciones causadas por viento y por las dinámicas que no hayan sido modeladas.

Apéndices

Apéndice A

A. Métodos empleados

Para el desarrollo de este trabajo se emplean las siguientes metodologías.

A.1. Control óptimo de horizonte infinito vía programación dinámica

Considere el siguiente sistema no lineal variante en el tiempo:

$$\dot{x}(t) = f(x(t), u(t)), \quad x(t_0) = x_0 \quad (\text{A.1})$$

donde $x(t) \in \mathbb{R}^n$, $u(t) \in \mathbb{R}^m$ y f es una función continuamente diferenciable en un dominio incluido en $\mathbb{R}^n \times \mathbb{R}^m$. El control $u(\cdot)$ pertenece a un conjunto \mathbb{U} , donde \mathbb{U} denota el conjunto de funciones continuas a tramos, subconjunto de \mathbb{R}^m , al aplicar el control $u(\cdot)$ se genera una trayectoria $x(\cdot)$ con condiciones iniciales x_0 , en el instante t_0 . Sea $x_{ref}(t) \in \mathbb{R}^n$, la señal de referencia establecida para A.1.

Suponga que se desea hallar un control $u(t) \in \mathbb{U}$, tal que el sistema en lazo cerrado en (A.1) genera una trayectoria que converja lo más rápido posible a la referencia fijada, en función de un índice de desempeño establecido. Para hallar una solución a este planteamiento se procede a definir dicho índice de desempeño, el cual, debe penalizar al estado del sistema y al control. Dicho índice se define como sigue:

$$\mathbf{J}^{u(\cdot)}(t_0, x_0) = \int_{t_0}^{\infty} \mathbf{L}(x(t), u(t)) dt \quad (\text{A.2})$$

donde L es una función escalar definida positiva que representa una penalización en el estado $x(\cdot)$ y en el control aplicado $u(\cdot)$. Para penalizar únicamente la energía aplicada al sistema, se puede definir $\mathbf{L}(x(t), u(t)) = u^T(t)u(t)$.

Suponga que $u^*(t)$ es un control que permite minimizar $\mathbf{J}^{u(\cdot)}(t_0, x_0)$, esto implica que:

$$\mathbf{J}^{u^*(\cdot)}(t_0, x_0) \leq \mathbf{J}^{u(\cdot)}(t_0, x_0), \quad u^*(\cdot) \in \mathbb{U}$$

Si lo anterior se cumple, $u^*(t)$ se dice control óptimo. El científico estadounidense Richard Bellman presentó el método de programación dinámica para resolver problemas de optimización con restricciones que involucran ecuaciones diferenciales. Una aplicación particular de este método, es el problema del control óptimo, en este problema las restricciones se interpretan como la ecuación de estado que representa al sistema a controlar. Este método consiste en reemplazar el problema de optimización dado por (A.1) y (A.2), el cual contiene una minimización en el espacio de controles admisibles \mathbb{U} , por una ecuación diferencial matricial en derivadas parciales,

A MÉTODOS EMPLEADOS

llamada ecuación de programación dinámica o ecuación de Hamilton-Jacobi-Bellman, definida como:

$$0 = \min_{u \in \mathbb{U}} \{ \mathbf{L}(x(t), u(t)) + \nabla_x V(x) \cdot f(x(t), u(t)) \}, \quad \forall t \in [t_0, \infty), x(t) \in \mathbb{R}^n \quad (\text{A.3})$$

donde $\frac{dV(x)}{dt} = \nabla_x V(x) \cdot f(x(t), u(t))$. La ecuación de Hamilton-Jacobi-Bellman se satisface por una función de Bellman $V(x)$, la cual se define como:

$$V(x_0) = \min_{u \in \mathbb{U}} \mathbf{J}^{u(\cdot)}(t_0, x_0) \quad (\text{A.4})$$

Para que la función de Bellman satisfaga la ecuación (A.3) es necesario que la función $V(\cdot)$ sea continuamente diferenciable a lo largo de las trayectorias del sistema (A.1). A continuación se explica brevemente el uso de estas ecuaciones para resolver el problema del control óptimo de horizonte infinito para el caso de los sistemas lineales.

Considere el siguiente sistema lineal:

$$\dot{x} = Ax(t) + Bu(t) \quad (\text{A.5})$$

donde el vector de estado $x(t) \in \mathbb{R}^n$, la matriz $A \in \mathbb{R}^{n \times n}$, $u(t) \in \mathbb{R}^m$, $B \in \mathbb{R}^{n \times m}$, sea \mathcal{J} un índice de desempeño cuadrático dado por:

$$\mathcal{J} = \int_0^\infty (x^T(t)Qx(t) + u^T(t)Ru(t))dt \quad (\text{A.6})$$

$Q \in \mathbb{R}^{n \times n}$ y $R \in \mathbb{R}^{m \times m}$ ambas son matrices constantes siendo Q semidefinida positiva y R definida positiva. Sea \mathbb{U} el conjunto de controles admisibles para el sistema (A.5).

$\mathbb{U} = \{ \text{todas las funciones } u(t) \mid u(t) \text{ es una función continua a tramos de forma que el sistema (A.5) es estable en lazo cerrado} \}$, lo anterior implica que el par (A, B) es controlable o al menos estabilizable. $u(t)$ es una función del estado $x(t)$, es decir $u(t) = u(x(t))$. Suponga que existe un control $u^*(x^*(t))$ admisible tal que se minimice el índice de desempeño (A.6).

$$u(x(t)) = u^*(x^*(t)) \quad (\text{A.7})$$

Con el fin de sintetizar la ley de control (A.7) se busca una función $V(x(t))$ definida positiva de manera que su derivada a lo largo de las trayectorias del sistema (A.5) sea:

$$\frac{dV(x^*(t))}{dt} = -L^*(x(t), u^*(t)) \quad (\text{A.8})$$

siendo :

$$L^*(x(t), u^*(t)) = x^T(t)Qx(t) + u^{*T}(x(t))Ru^*(x(t))$$

Integrando ambos miembros de la ecuación (A.8) desde 0 hasta ∞ se tiene:

$$\lim_{t \rightarrow \infty} V(x(t)) - V(x_0) = - \int_0^\infty L^*(x^*(t, x_0), u^*(t)) dt$$

Se supone que u^* es un control admisible, de esta forma estabiliza al sistema por lo que:

$$V(x_0) = \int_0^\infty L^*(x^*(t, x_0), u^*(t)) dt$$

$x^*(t, x_0)$ es la solución del sistema en lazo cerrado aplicando el control $u^*(t)$, dado que $L^*(x^*(t), u^*(t))$, es definida positiva, se puede probar que la función de Bellman $V(x(t))$ es una función de Lyapunov para el sistema (A.5) en lazo cerrado con el control $u^*(t)$. A partir de la ecuación (A.8) se puede deducir que:

$$\frac{dV(x(t))}{dt} + L^*(x^*(t), u^*(t)) = 0 \quad (\text{A.9})$$

Lo anterior puede ser escrito como:

$$\min_{u \in \mathbb{U}} \left(\left. \frac{dV(x(t))}{dt} \right|_{(\text{A.5})} + x^T(t)Qx(t) + u^T(t)Ru(t) \right) = 0 \quad (\text{A.10})$$

La ecuación anterior es conocida como ecuación de Hamilton-Jacobi-Bellman para sistemas lineales y la función que satisface dicha ecuación es llamada función de Bellman. Cuando la función de Bellman $V(x(t))$ es conocida o propuesta, la solución de la ecuación (A.10) con respecto a $u(t)$, proporciona el control óptimo $u^*(t)$. Dicha solución es mostrada a continuación.

La función $V(x(t))$ puede ser considerada como una función de Lyapunov, aplicando $u^*(t)$ al sistema en lazo cerrado, debido a que el sistema considerado es lineal, dicha función puede proponerse como:

$$V(x(t)) = x^T(t)Px(t) \quad (\text{A.11})$$

donde $P \in \mathbb{R}^{n \times n}$ es definida positiva. Ahora se procede a calcular la derivada de la función propuesta a lo largo de las trayectorias del sistema (A.5)

$$\left. \frac{dV(x(t))}{dt} \right|_{(\text{A.5})} = 2x^T(t)P(Ax(t) + Bu(t)) \quad (\text{A.12})$$

sustituyendo (A.12) en (A.10), se obtiene:

$$\min_{u \in \mathbb{U}} (2x^T(t)PAx(t) + 2x^T(t)PBu(t) + (x^T(t)Qx(t) + u^T(t)Ru(t))) = 0 \quad (\text{A.13})$$

Teniendo en mente esto, es posible hallar de forma explícita la ley de control óptima $u^*(t)$ observe que la ecuación (A.13) es una función cuadrática y fuertemente convexa [14] respecto a $u(t)$, por lo anterior la existencia de un mínimo para la ecuación (A.13) está garantizada globalmente. De acuerdo con el enfoque de cálculo de variaciones la primera variación de la función a optimizar con respecto a $u(t)$ debe ser cero.

$$\frac{\partial}{\partial u(t)} (2x^T(t)PAx(t) + 2x^T(t)PBu(t) + x^T(t)Qx(t) + u^T(t)Ru(t)) = 0$$

De esta última expresión se obtiene:

$$2B^T Px(t) + 2Ru(t) = 0$$

Por lo tanto la ley de control óptima para el sistema (A.5) es:

$$u^*(t) = -R^{-1}B^T Px(t) \quad (\text{A.14})$$

La ecuación anterior es óptima debido a que:

$$\frac{\partial^2}{\partial u^2(t)} (2x^T(t)PAx(t) + 2x^T(t)PBu(t) + x^T(t)Qx(t) + u^T(t)Ru(t)) = 2R \quad (\text{A.15})$$

Debido a que la matriz R es estrictamente definida positiva, la condición suficiente para la optimalidad del cálculo de variaciones, implica que $u^*(t)$ de la ecuación (A.14) es óptimo.

La matriz P de la función $V(x(t))$, es desconocida, sin embargo, puede hallarse a partir de matrices conocidas como se describe a continuación. Primero se calcula la derivada temporal de la ecuación (A.11) y se evalúa a lo largo de las trayectorias del sistema (A.5) en lazo cerrado con la ley de control óptima (A.14), para obtener la siguiente expresión:

$$\left. \frac{dV(x(t))}{dt} \right|_{(A.5)} = (x^T(t)(A^T P + PA)x(t) + u^{*T}(t)B^T Px(t) + x^T(t)PBu^*(t)) \quad (\text{A.16})$$

Sustituyendo la expresión dada en (A.16) y la ley de control óptima dada por (A.15) en la ecuación (A.9), se tiene que para todo valor del estado $x(t)$:

$$x^T(t)(A^T P + PA + Q - PBR^{-1}B^T P)x(t) = 0 \quad (\text{A.17})$$

La ecuación anterior es conocida como ecuación algebraica del tipo de Riccati, la cual puede ser resuelta numéricamente para hallar el control óptimo $u^*(t)$ [21].

A.2. Control PID

Sea $e(t)$ la señal de error, definida como la diferencia entre la referencia $r(t)$ y el valor medido $y(t)$, en un sistema de control en lazo cerrado. Un controlador PID (Proporcional Integral Derivativo) consiste en la sumatoria de tres acciones aplicadas a dicha señal de error, éstas son la acción proporcional, la acción integral y la acción derivativa, las cuales son descritas a continuación.

Acción proporcional: La acción proporcional actúa de forma gradual en función de la magnitud de la señal de error, como lo indica la siguiente expresión:

$$u(t) = K_p e(t) = K_p [r(t) - y(t)]$$

donde K_p es la ganancia proporcional, la cual puede definirse como la función de transferencia de un controlador meramente proporcional, es decir:

$$C(s) = U(s)/E(s) = K_p$$

Acción integral: La acción integral genera una señal de salida proporcional a la integral de la señal de error, ésta está dada por la siguiente ecuación:

$$u(t) = K_i \int_0^t e(\tau) d\tau$$

siendo K_i la ganancia integral. En cierta forma esta acción, está relacionada con los valores anteriores del error, ya que podría decirse que es la acumulación de errores anteriores y está en función de éstos. La función de transferencia para la acción integral es:

$$C(s) = U(s)/E(s) = \frac{K_i}{s}$$

Acción Derivativa: Mientras la acción proporcional está basada en los valores actuales de la señal de error y acción integral está basada en los valores pasados de dicha señal, la acción derivativa actúa sobre valores futuros. Una ley de control derivativa ideal puede ser expresada como:

$$u(t) = K_d \frac{de(t)}{dt}$$

K_d es la ganancia derivativa. La correspondiente función de transferencia es:

$$C(s) = U(s)/E(s) = K_d s$$

Defina:

$$K_i = \frac{K_p}{T_i}$$

$$K_d = K_p T_d$$

donde T_i es el tiempo de integración y T_d es el tiempo de derivación, ambos constantes. La función de transferencia de un controlador PID puede escribirse como:

$$C(s) = K_p + \frac{K_i}{s} + K_d s = K_p \left(1 + \frac{1}{T_i s} + T_d s \right)$$

Implementación digital: Para aplicar un control PID de forma digital, se requiere discretizar las leyes de control antes mencionadas. Considere la expresión en tiempo continuo de un controlador PID en su forma ideal:

$$u(t) = K_P \left(e(t) + \frac{1}{T_i} \int_0^t e(\tau) d\tau + T_d \frac{de(t)}{dt} \right) \quad (\text{A.18})$$

Defina ΔT como el tiempo de muestreo. La acción integral en (A.18) puede ser escrita como:

$$\int_0^{tk} e(\tau) d\tau = \sum_{i=1}^k e(t_i) \Delta t$$

donde $e(t_i)$ es el error del sistema en tiempo continuo, en el i -ésimo instante de muestreo. La acción derivativa discretizada vía una aproximación de Euler resulta:

$$\frac{de(t_k)}{dt} = \frac{e(t_k) - e(t_{k-1})}{\Delta t}$$

De lo anterior, la ley de control discretizada es:

$$u(t_k) = K_p \left(e(t_k) + \frac{\Delta t}{T_i} \sum_{i=1}^k e(t_i) + \frac{T_d}{\Delta t} (e(t_k) - e(t_{k-1})) \right) \quad (\text{A.19})$$

De esta manera, el valor de la variable de control es determinada directamente. Alternativamente, la variable de control en el tiempo t_k puede ser calculada empleando su valor anterior $u(t_{k-1})$. Substrayendo $u(t_{k-1})$ de $u(t_k)$, se obtiene:

$$u(t_k) = u(t_{k-1}) + K_P \left[\left(1 + \frac{\Delta t}{T_i} + \frac{T_d}{\Delta t} \right) e(t_k) + \left(-1 - \frac{2T_d}{\Delta t} \right) e(t_{k-1}) + \frac{T_d}{\Delta t} e(t_{k-2}) \right] \quad (\text{A.20})$$

La ecuación (A.20) puede ser reescrita como:

$$u(t_k) - u(t_{k-1}) = K_1 e(t_k) + K_2 e(t_{k-1}) + K_3 e(t_{k-2})$$

siendo:

$$K_1 = K_p \left(1 + \frac{\Delta t}{T_i} + \frac{T_d}{\Delta t} \right)$$

$$K_2 = -K_p \left(1 + \frac{2T_d}{\Delta t} \right)$$

$$K_3 = K_p \frac{T_d}{\Delta t}$$

Definiendo el operador q^{-1} como el operador de recorrido hacia atrás en tiempo discreto, es decir:

$$q^{-1}u(t_k) = u(t_{k-1})$$

El controlador PID discretizado puede ser escrito como:

$$C(q^{-1}) = \frac{K_1 + K_2 q^{-1} + K_3 q^{-2}}{1 - q^{-1}}$$

donde K_1 , K_2 y K_3 pueden ser vistos como los parámetros de sintonización [23].

A.3. Regulación vía control Integral

Las ventajas de la parte integral de un control PID, pueden también extenderse para un control por retroalimentación de estado. Considere el sistema no lineal:

$$\dot{x} = f(x, u)$$

$$y = h(x)$$

donde $x \in \mathbb{R}^n$, $u \in \mathbb{R}^p$, f y h son funciones continuamente diferenciables en un dominio incluido en $\mathbb{R}^n \times \mathbb{R}^p$, $y \in \mathbb{R}^p$ es la salida a regular.

Sea $y_R \in \mathbb{R}^p$ una señal de referencia constante, se desea diseñar un control en realimentación tal que:

$$y(t) \rightarrow y_R \text{ cuando } t \rightarrow \infty$$

aquí se supone que se mide la salida y . Se desea estabilizar el sistema en lazo cerrado tal que $y = y_R$. Es claro que, para que esto sea posible, deben existir valores x_f y u_f tales que:

$$0 = f(x_f, u_f)$$

$$0 = h(x_f) - y_R$$

Se considera que la ecuación anterior tiene una solución única en el dominio de interés. Defina el error de seguimiento como: $e \triangleq y - y_R$. Al integrar el error e se obtiene:

$$\dot{\sigma} = e = y - y_R$$

Considere a σ como otra variable en la representación espacio estado del sistema como sigue:

$$\dot{x} = f(x, u)$$

$$\dot{\sigma} = h(x) - y_R$$

El control se diseña como una realimentación de las variables de estado (x, σ) , tal que el sistema en lazo cerrado tenga un punto de equilibrio en (x, σ) con $\bar{x} = x_f$.

Ahora, se asume que el sistema no lineal es linealizabile alrededor del punto de equilibrio considerado, por lo que $(x, \sigma, u) = (x_f, \bar{\sigma}, u_f)$, de esto se obtiene:

$$\dot{\xi} = \begin{bmatrix} \mathbf{A} & \mathbf{0} \\ \mathbf{C} & \mathbf{0} \end{bmatrix} \xi + \begin{bmatrix} \mathbf{B} \\ \mathbf{0} \end{bmatrix} v \triangleq \mathcal{A}\xi + \mathcal{B}v$$

con

$$\xi = \begin{bmatrix} x - x_f \\ \sigma - \bar{\sigma} \end{bmatrix} \quad (\text{A.21})$$

$$v = u - u_f \quad (\text{A.22})$$

donde:

$$\mathbf{A} = \left. \frac{\partial f}{\partial x} \right|_{(x,u)=(x_f,u_f)}, \quad \mathbf{B} = \left. \frac{\partial f}{\partial u} \right|_{(x,u)=(x_f,u_f)}, \quad \mathbf{C} = \left. \frac{\partial h}{\partial x} \right|_{x=x_f}$$

Si el par (\mathbf{A}, \mathbf{B}) es controlable y la matriz $\begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{0} \end{bmatrix}$, tiene rango $n + p$, entonces $(\mathcal{A}, \mathcal{B})$ es controlable. Suponga que esto se cumple, esto implica que es posible calcular \mathcal{K} tal que $\mathcal{A} + \mathcal{B}\mathcal{K}$ sea Hurwitz. \mathcal{K} se puede dividir en dos submatrices $[\mathcal{K}_1 \ \mathcal{K}_2]$. El control es entonces:

$$u = \mathcal{K}_1(x - x_f) + \mathcal{K}_2(\sigma - \bar{\sigma})$$

por lo que:

$$u = \mathcal{K}_1(x - x_R) + \mathcal{K}_2 \int_0^t (y - y_R) dt$$

Observe que si $y = x$ (el estado del sistema está disponible), $u = \mathcal{K}_1(x - x_R) + \mathcal{K}_2 \int_0^t (x - x_R) dt$. El cálculo de las submatrices \mathcal{K}_1 y \mathcal{K}_2 se hace de tal forma que se minimice el índice de desempeño J , definido como:

$$J = \int_0^{\infty} (\xi^T \mathcal{Q} \xi + u^T \mathcal{R} u) dt$$

$$\Rightarrow u = -\mathcal{R}^{-1} \mathcal{B}^T \mathbb{P} \xi$$

con:

$$\mathcal{Q} \in \mathbb{R}^{2n \times 2n}, \quad R \in \mathbb{R}^{p \times p}, \quad \mathbb{P} \in \mathbb{R}^{2n \times 2n}$$

El esquema del control integral es mostrado en la figura 56 [14].

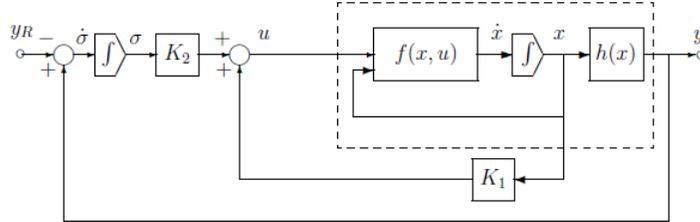


Figura 56: Esquema control integral.

A.4. Filtro Complementario

En ciertas ocasiones existen casos prácticos donde se tienen dos sensores para estimar la misma variables. Las características del ruido de dos medidas simultaneas, pueden ser tales que una de las fuentes genera información más precisa a bajas frecuencias, mientras que la otra fuente presenta una mayor precisión a altas frecuencias. El uso de sensores que trabajan con la tecnología MEMS presenta esta problemática, al emplearlos para estimar los ángulos de orientación en un UAV. Lo anterior debido a que los acelerómetros generan una buena medida de desplazamiento angular a bajas frecuencias, mientras un giroscopio es mejor midiendo la velocidad angular a altas frecuencias.

Para trabajar con este fenómeno debido a la naturaleza de los sensores, existen algunos métodos de filtrado, como por ejemplo, el filtro complementario, el filtro de Kalman, el filtro de Kalman extendido, etc.

El filtro complementario tiene la ventaja de ser aplicado de forma relativamente fácil además de no generar retardos grandes en el sistema al ser empleado. Matemáticamente un filtro complementario es representado de la siguiente forma:

$$\hat{x} = F_1(s)x_b + F_2(s)\frac{x_h}{s} \tag{A.23}$$

donde \hat{x} denota la estimación de la variable x . $F_1(s)$ representa el filtro pasa bajas, $F_2(s)$ se refiere al filtro pasa altas, x_b es la medida a baja frecuencia libre de derivar y x_h es la medida a altas frecuencias. El término $\frac{1}{s}$ indica que la medida a altas frecuencias debe ser integrada primero antes de introducirla al filtro. El filtro es nombrado complementario si $F_1(s) + F_2(s) = 1$. Por ejemplo.

$$F_1(s) = \frac{k}{s+k}$$

$$F_2(s) = \frac{s}{s+k}$$

Siendo k la frecuencia de conmutación. Por tanto, si $k = 0$ la ecuación (A.23) se convierte en un integrador puro, ya que $F_1(s) = 0$ y $F_2(s) = 1$. En este caso, la estimación \hat{x} diverge de la derivada intrínseca del giroscopio. Por otro lado si $k = \infty$ se tiene $F_1 = 1$ y $F_2 = 0$, lo que implica que toda la información generada por el giroscopio es eliminada. En consecuencia, se debe calcular un valor de k para el cual se obtiene una buena estimación de la variable. El esquema elemental que representa un filtro complementario es mostrado en la figura 57. La medida x_b corresponde a la señal generada por el acelerómetro y la señal x_h es aquella producida por el giroscopio [20].

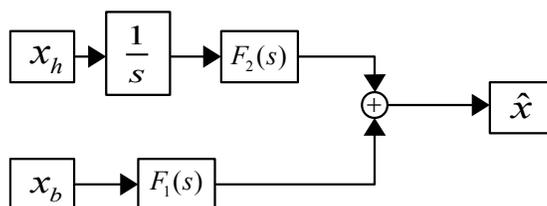


Figura 57: Esquema del filtro complementario.

A.5. Filtro de Kalman

Debido a que los modelos matemáticos son sólo una aproximación del comportamiento de los sistemas dinámicos, y a que las señales sensadas son afectadas por ruido, existe incertidumbre

en las medidas y a pesar de usar modelos determinísticos, éstos se ven afectados por procesos estocásticos, el filtro de Kalman permite estimar los estados de un sistema estocástico. Puede definirse como un algoritmo de procesamiento de datos óptimo recursivo. Óptimo en el sentido de que minimiza un criterio determinado y porque incorpora toda la información que se le suministra para determinar el filtrado. Recursivo porque no precisa mantener los datos previos, los que facilita su implementación en sistemas de procesamiento en tiempo real. Este filtro ha sido diseñado para ser implementado en sistemas discretos. El objetivo del filtro de Kalman es estimar los estados de una forma óptima, de manera que se minimiza un índice de error cuadrático medio.

Un sistema lineal discretizado empleando un tiempo de muestreo ΔT puede ser representado como:

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) + v(k) \\ y(k) &= Cx(k) + w(k) \end{aligned}$$

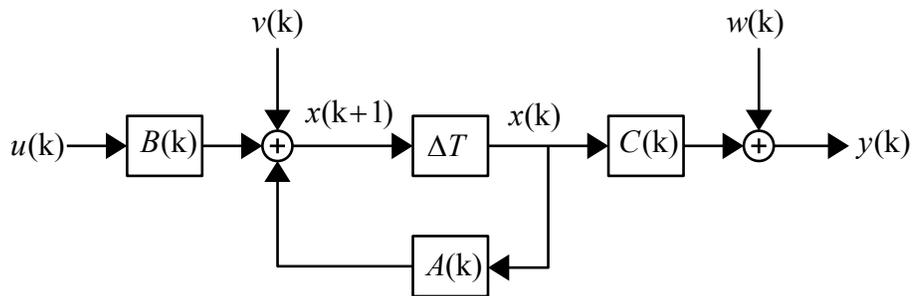


Figura 58: Esquema de sistema discretizado.

Siendo:

$x(k)$ el estado del sistema.

$y(k)$ la salida del sistema.

$w(k)$ proceso estocástico asociado a la medida.

$v(k)$ proceso estocástico asociado al sistema.

A, B, C matrices determinísticas que definen la dinámica del sistema.

Para implementar este filtro, es necesario suponer las siguientes consideraciones:

$$\begin{aligned}
 E[x(0)] &= x_0 \\
 E[w(k)] &= 0 \quad \forall k \\
 E[v(k)] &= 0 \quad \forall k \\
 E[x(0), w(k)] &= 0 \quad \forall k \\
 E[x(0), v(k)] &= 0 \quad \forall k \\
 E[v(k), w(j)] &= 0 \quad \forall k \\
 E[w(k), w(k)] &= R(k) \\
 E[w(k), w(j)] &= 0 \quad \forall k \neq j \\
 E[v(k), v(k)] &= Q(k) \\
 E[v(k), v(j)] &= 0 \quad \forall k \neq j \\
 E[x(0), x(k)] &= P_0 \quad \forall k
 \end{aligned}$$

donde $E[\cdot]$ es la esperanza matemática, Q , R y P son las matrices de covarianza las cuales son diagonales y por tanto simétricas. Sea $\hat{x}(k)$ la estimación del estado $x(k)$ en el instante k . Definiendo el error $e(k)$ como la diferencia entre el valor real del estado y la estimación se tiene:

$$e(k) = x(k) - \hat{x}(k)$$

El objetivo es minimizar:

$$P(n) = E \{e(n)e(n)^T\} \tag{A.24}$$

$P(n)$ es la matriz de covarianza del error. Dependiendo del valor que tome n en la ecuación (A.24), se tienen distintas representaciones del filtro de Kalman. Si $n = k + 1$, el filtrado es de predicción, si $n = k$ el filtrado es de alisado.

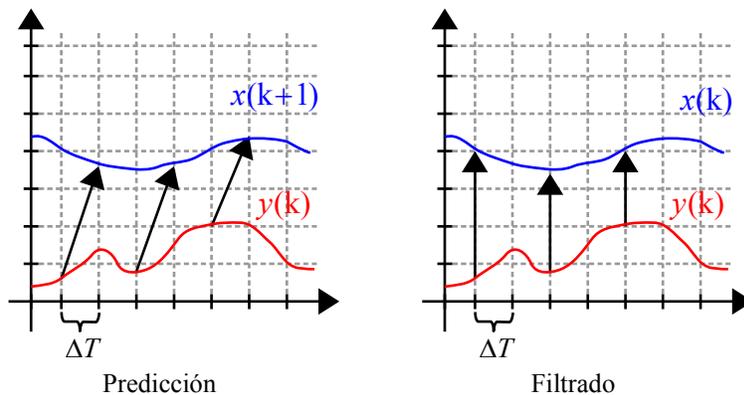


Figura 59: Representaciones del filtro de Kalman.

El objetivo del filtro de Kalman consiste en determinar los valores de $\hat{x}(k)$ a partir de las señales contaminadas $y(0)$, $y(1)$, ..., $y(k)$ para que la matriz de covarianza del error $P(k)$ sea mínima, o dicho de otra forma, determinar los valores de $\hat{x}(k+1)$ a partir de las medidas afectadas

por el ruido de la observación $y(k+1)$ para que la matriz $P(k+1)$ se a mínima.

El filtrado de $\hat{x}(k+1)$, se realiza a partir del estado anterior y contemplando un factor de corrección que está en función del error. El algoritmo se compone de dos pasos los cuales son ejecutados de forma iterativa. Predicción antes de tener la medida $y(k+1)$ y corrección o actualización del estado. En la siguiente figura se puede ver el proceso de cálculo.

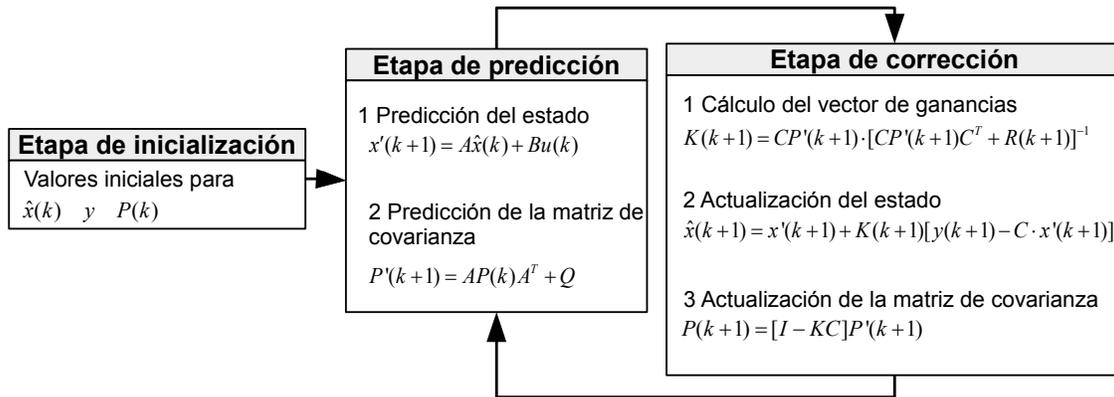


Figura 60: Algoritmo del filtro de Kalman.

Paso 1 Predicción-Extrapolación-Estimación.

Se calcula una predicción del estado $x(k+1)$, sea $x'(k+1)$ dicha predicción, el valor de la predicción es hallado a partir del valor más actualizado del estado.

$$x'(k+1) = A\hat{x}(k) + Bu(k)$$

Se predice el valor de la matriz de covarianza del error previo a la medida $P'(k+1)$. De acuerdo a la definición de error dada en (A.24), el error está dado por:

$$e'(k+1) = x(k+1) - x'(k+1)$$

Lo anterior implica que $e'(k+1)$ en esta etapa de predicción vale:

$$\begin{aligned} e'(k+1) &= Ax(k) + Bu(k) + v(k) - A\hat{x}(k) - Bu(k) \\ &= A[x(k) - \hat{x}(k)] + v(k) \end{aligned}$$

La matriz de covarianza $P'(k+1)$ por tanto es:

$$P'(k+1) = E \left\{ [A(x(k) - \hat{x}(k)) + v(k)][A(x(k) - \hat{x}(k)) + v(k)]^T \right\}$$

implica:

$$P'(k+1) = E \left\{ [A(x(k) - \hat{x}(k))][A(x(k) - \hat{x}(k))]^T \right\} + E \left\{ v(k)v(k)^T \right\}$$

Ya que $P(k) = E[e(k)e(k)^T]$ y que $Q = E[v(k)v(k)^T]$ la ecuación anterior puede reescribirse como:

$$P'(k+1) = AP(k)A^T + Q$$

Paso 2 Innovación-Actualización-Corrección.

El valor del estado es calculado a partir del estado anterior y de un factor de corrección que está en función del error. Esto es:

$$\hat{x}(k+1) = x'(k+1) + K(k+1)[y(k+1) - Cx'(k+1)]$$

Donde el factor de corrección es:

$$K(k+1)[y(k+1) - Cx'(k+1)]$$

y $y(k+1)$ es el último valor observado, $x'(k+1)$ es el valor más actualizado disponible del estado, el cual, fue calculado en la fase de predicción. Se busca el valor $K(k+1)$ para hallar un valor óptimo de $\hat{x}(k+1)$ tal que la matriz de covarianza del error sea mínima. Por definición se tiene que el error es:

$$e(k+1) = x(k+1) - \hat{x}(k+1) \tag{A.25}$$

Sustituyendo el valor de $\hat{x}(k+1)$ en la ecuación(A.25) se tiene:

$$e(k+1) = x(k+1) - x'(k+1) - K[y(k+1) - Cx'(k+1)]$$

La observación $y(k+1)$ tiene el valor:

$$y(k+1) = Cx(k+1) + w(k+1)$$

Por lo tanto:

$$e(k+1) = x(k+1) - x'(k+1) - KCx(k+1) - Kw(k+1) - KCx'(k+1)$$

Al agrupar términos se obtiene:

$$e(k+1) = [I - KC]((x(k+1) - x'(k+1)) - Kw(k+1))$$

Ahora aplicando la la definición de la ecuación (A.24).

$$P(k+1) = E \{ \langle [I - KC]((x(k+1) - x'(k+1)) - Kw(k+1)) \rangle \langle [I - KC]((x(k+1) - x'(k+1)) - Kw(k+1))^T \rangle \}$$

$$P(k+1) = E \{ \langle [I - KC]((x(k+1) - x'(k+1))) \rangle \langle [I - KC]((x(k+1) - x'(k+1)))^T \rangle + E \{ Kw(k+1) w(k+1)^T K^T \}$$

$E \{ Kw(k+1) w(k+1)^T K^T \}$ es la matriz de covarianza asociada a la medida $R(k+1)$. Sustituyendo dicha matriz en la expresión anterior y factorizando se tiene:

$$P(k+1) = [I - KC]E \{ \langle (x(k+1) - x'(k+1)) \rangle \langle (x(k+1) - x'(k+1))^T \rangle \} [I - KC]^T + KR(k+1)K^T$$

Anteriormente se definió que:

$$P'(k+1) = E \{ \langle (x(k+1) - x'(k+1)) \rangle \langle (x(k+1) - x'(k+1))^T \rangle \}$$

Con lo que:

$$P(k+1) = [I - KC]P'(k+1)[I - KC]^T + KRK^T \quad (\text{A.26})$$

Haciendo cambio de variable, $P'(k+1) \triangleq P1$ y $P'(k) \triangleq P$, la ecuación (A.26) puede ser reescrita como:

$$P = P1 + KCP + PC^T K^T + KCPC^T K^T + KRK^T$$

$$TrP = TrP1 + TrKCP + TrPC^T K^T + TrKCPC^T K^T + TrKRK^T \quad (\text{Tr traza de la matriz})$$

Diferenciando respecto a K para obtener el mínimo de la expresión e igualamos a cero.

$$\frac{\partial TrP}{\partial K} = -2TrCP + 2TrCPC^T + 2KR = 0$$

$$K = CP[CPC^T + R]^{-1}$$

es decir:

$$K(k+1) = CP'(k+1) \cdot [CP'(k+1)C^T + R(k+1)]^{-1} \quad (\text{A.27})$$

Se puede comprobar que es el mínimo a través del Hessiano. Sustituyendo (A.27) en (A.26):

$$P(k+1) = [I - KC]P'(k+1)$$

Inicialización del algoritmo.

Para iniciar el algoritmo es necesario conocer las siguientes condiciones de contorno: Un valor inicial del estado $x(0)$. El valor de la matriz de covarianza del error, que para $P(0)$. Se puede asumir que $P(0) = Q$. Y los valores de las matrices de covarianza asociadas al sistema y a la medida Q y R [24].

Apéndice B

B. Herramientas

El problema es atacado empleando las siguientes herramientas y programas de computadora.

B.1. Central inercial

Los sensores inerciales son dispositivos electromecánicos, los cuales permiten medir magnitudes vectoriales cinemáticas en un cuerpo, tales como la aceleración, la velocidad y la velocidad angular. Las centrales inerciales ó unidades de medida inercial (IMU Inertial Measurement Unity), se componen por lo general de al menos un acelerómetro de tres ejes y un giroscopio de tres ejes, algunas más sofisticadas contienen además barómetro y/o magnetómetro. Debido a que en una IMU existen múltiples sensores es común que éstas sean manipuladas a través de los protocolos *I²C* y *RS232*.

Una estimación precisa en la orientación, aceleración, altura y velocidades, es esencial para el control autónomo de vehículos aéreos no tripulados. Los criterios a tomar en cuenta para elegir una IMU son por lo regular su exactitud, su sensibilidad al ruido, su precio, además considerando el corto tiempo de vuelo debido a la carga útil limitada que presentan los vehículos aéreos de cuatro rotores, aspectos como el tamaño, la forma y el peso se vuelven criterios muy importantes a la hora de elegir una IMU. En términos de peso, de talla y de costo, los sensores de tecnología MEMS (Micro Electro Mechanical Systems) son una muy buena opción para instrumentar un mini vehículo aéreo no tripulado, las desventajas que presentan este tipo de sensores es su alta sensibilidad al ruido electromagnético.

B.1.1. Acelerómetro

Un acelerómetro permite medir la aceleración de un cuerpo, dicha aceleración está regularmente dada en g que corresponde a la aceleración promedio de la tierra, por tanto $1g$ corresponde a $9.8m/s^2$, la señal entregada por el este sensor es generalmente una tensión eléctrica, la cual es proporcional a la aceleración detectada.

Los acelerómetros construidos con tecnología MEMS, utilizan diferentes fenómenos y efectos para efectuar la medida de la aceleración que sufre un cuerpo al que son fijados. Existen acelerómetros basados en tecnología capacitiva y otros que emplean fenómenos piezoeléctricos, etc [20].

Al emplear un acelerómetro se debe contemplar el hecho de que cualquier cuerpo que se encuentre en la tierra es afectado por la aceleración causada por la fuerza de gravedad de ésta, por ende la lectura de la aceleración está integrada por una componente estática, debida a lo anterior, y por una componente dinámica que es la causada por el movimiento o movimientos que realice dicho cuerpo. Por ejemplo, para el sistema presentado en este trabajo, si se define $\ddot{x}(t)$ como la aceleración presentada en el eje X , puede escribirse que ésta está dada por:

$$\ddot{x}(t) = \ddot{x}_a + \ddot{x}_g \tag{B.1}$$

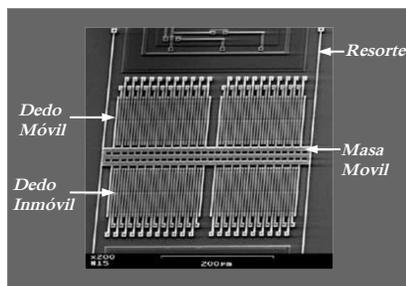


Figura 61: Acelerómetro basado en tecnología MEMS.

donde \ddot{x}_a es la aceleración medida por el acelerómetro, \ddot{x}_g es la aceleración causada por la gravedad y el ángulo de cabeceo θ , en el eje X . El ángulo de cabeceo para este caso es calculado mediante el giroscopio. Si se contempla lo mostrado en la figura 62 puede definirse la aceleración real en eje X como:

$$\ddot{x}(t) = \ddot{x}_a - g \sin(\theta) \tag{B.2}$$

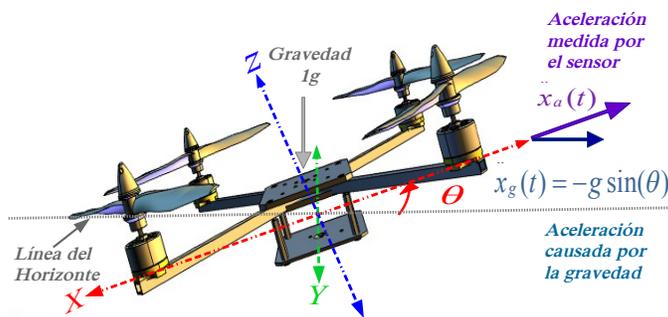


Figura 62: Componentes de aceleración en el eje X .

B.1.2. Giroscopio

Los giroscopios permiten obtener la velocidad angular en torno a los tres ejes de referencia X , Y y Z , mediante una aproximación numérica de una integral es posible calcular los tres ángulos de orientación del mini helicóptero de cuatro rotores (θ, ϕ y ψ). En este trabajo se emplea la aproximación para la integral llamada regla del trapecio.

Existen dos tipos de giroscopios los cuales son frecuentemente empleados en la construcción de sistemas de navegación robótica, los primeros trabajan empleando las propiedades de la luz,

por ejemplo los giroscopios que emplean lasers, los segundos trabajan bajo el principio de la vibración mecánica, éstos últimos son menos precisos que los primeros, sin embargo son significativamente más pequeños y por tanto menos pesados, por esta razón son empleados en los vehículos aéreos no tripulados de pequeña escala.

Se procede a describir brevemente el funcionamiento de los giroscopios que trabajan bajo el principio de una estructura vibrante (VSG), los cuales son dispositivos que generan un voltaje proporcional a la variación del ángulo aplicado a los ejes del giroscopio. Estos giroscopios operan bajo el principio de detección de fuerzas de Coriolis, que agitan un elemento vibrante cuando el cuerpo está en rotación. La aceleración de Coriolis es dirigida ortogonalmente a la velocidad de rotación y a la dirección de la vibración, y ésta tiende a modificar la orientación de la red de vibración proporcionalmente a la rotación alrededor del eje sentido. La fuerza de Coriolis es dada por:

$$F_c = 2m(\omega \times v) \tag{B.3}$$

Donde m es la masa, ω es la velocidad angular y v es la velocidad lineal de la masa [20].

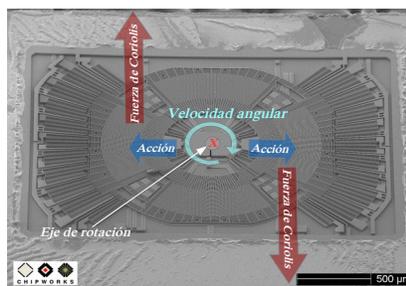


Figura 63: Giroscopio tecnología MEMS referido al eje X.

B.2. Programa MATLAB

MATLAB, acrónimo de MATrix LABoratory, es actualmente una de las principales herramientas software que existen en el mercado para cálculo matemático, análisis de datos, simulación y visualización de resultados. Desde su aparición en los años 70, ha ido introduciéndose con fuerza en el ámbito científico y universitario. En la actualidad es algo más que una herramienta de ayuda al cálculo avanzado, es también un lenguaje de programación con una elevada potencia de cálculo matricial. Una importante ventaja que presenta MATLAB es el entorno gráfico de trabajo, la claridad en la presentación de los resultados y la versatilidad que presenta para la creación de nuevas funciones.

Todas las operaciones que realiza MATLAB se basan en una estructura de datos matricial. Dentro del entorno de trabajo de MATLAB, se pueden definir nuevos comando o funciones,

programadas por el propio usuario, a través de los ficheros con extensión *m*. Este tipo de ficheros se encuentran en las llamadas *toolbox* de MATLAB, que son una colección de funciones ya programadas y disponibles para el usuario. En este sentido, MATLAB funciona con un lenguaje interpretado, es decir, reconoce y ejecuta los comandos que se le pasan a través del editor de comandos (*Command Window*). Todas estas funciones o comandos se agrupan en las *toolboxes* y abarcan diferentes campos, como son el análisis y adquisición de datos, procesamiento de imágenes y de señales, análisis y diseño de sistemas de control, comunicaciones, simulaciones en tiempo real y también herramientas para finanzas, estadística, economía o lógica difusa.

Dentro del campo de control automático, MATLAB ha desarrollado un gran número de funciones para el análisis de los sistemas de regulación. Todas ellas se encuentran dentro de la *Control System Toolbox* y permiten el análisis en el dominio del tiempo y de la frecuencia, tanto de los sistemas continuos como discretos; asimismo permite la definición de los sistemas en distintos formatos y la conversión entre los mismo.

MATLAB incorpora interfases gráficas muy prácticas e intuitivas, como son la *SISO Design Tool* o la *Root Locus Tool* para el análisis y diseño de sistemas de control, o la *System Identification Tool* para la identificación de sistemas. La interfase gráfica por excelencia es *Simulink*, que en realidad es una aplicación integrada dentro de MATLAB, y que permite la realización de simulaciones mediante la construcción gráfica de diagramas de bloques.

Con la incorporación del compilador de MATLAB es posible generar código C, C++, librerías para Windows y Linux, aplicaciones ejecutables (*stand-alone applications*) y otro tipo de componentes de software a partir de los ficheros *.m*; todo esto permite extender la funcionalidad de MATLAB a equipos donde no se tenga instalada esta aplicación [25].

En este trabajo MATLAB es usado para resolver la ecuación algebraica del tipo de Riccati de cada subsistema, mediante el comando DARE(Discret Algebraic Riccati Equation). Además se simula el sistema de control discretizado en Simulink con el fin de validar el comportamiento del UAV y observar el comportamiento de las señales de control $u(t)$, τ_θ , τ_ϕ y τ_ψ .

B.3. Programa Dynamic C

Dynamic C es un sistema de desarrollo integrado para escribir software embebido. Está diseñado para ser usado con controladores Rabbit y otros controladores basados en microprocesadores Rabbit.

Dynamic C integra las siguientes funciones de desarrollo:

- Edición
- Compilación
- Enlace
- Descarga
- Depuración

dentro de un programa. De hecho la compilación, el enlace y la descarga contemplan una etapa. Dynamic C cuenta con un editor versátil y poderoso, fácil de emplear. Los programas creados dentro del entorno de Dynamic C pueden ser compilados y depurados interactivamente a nivel máquina, además se cuenta con rutas cortas mediante el manejo del teclado para mayor comodidad del usuario.

Dynamic C también soporta programación en lenguaje ensamblador. No es necesario dejar de usar C o sus herramientas de desarrollo para escribir códigos basados en lenguaje ensamblador, por tanto C y ensamblador pueden ser mezclados en este entorno.

La depuración bajo Dynamic C incluye la habilidad para usar comandos de impresión de datos, monitoreo de datos y puntos de ruptura. Los datos pueden ser monitorizados en los puntos de ruptura o mientras se ejecuta el programa. Pueden ejecutarse estructuras multitarea gracias a la extensión del lenguaje C empleada en Dynamic C, además se cuenta con demasiadas librerías y funciones, que pueden ser agregadas a los códigos fuente, estas librerías soportan programación en tiempo real, I/O a nivel máquina, y proveen de funciones matemáticas estándar. La compilación se realiza directo a la memoria, en una PC rápida, Dynamic C debería cargar 30,000 byte de código en cinco segundos a una velocidad de transmisión de datos de 115,200 bps [26].

Para este trabajo a través de Dynamic C es programado el microprocesador embarcado en la plataforma experimental en ambiente controlado descrita en el Capítulo 2, gracias a su versatilidad y a la gran ventaja que presenta al permitir programar multitareas, se puede gestionar la comunicación entre la PC y el controlador Rabbit al mismo tiempo que éste se comunica con la IMU del UAV y controlar los motores mediante la gestión y tratamiento de la información generada por las dos fuentes de datos mencionadas.

B.4. Programa Motive de OptiTrack

El programa Motive Tracker de Optitrack permite crear aplicaciones de ingeniería para hacer seguimiento de objetos de seis grados de libertad con una alta precisión. Motive cuenta con soporte para trabajar en tiempo real y mediante librerías para C y C++ puede ser vinculado con programas externos [27]. Esto se logra mediante el uso de marcadores colocados en el objeto o cuerpo a monitorear, un conjunto de cámaras colocadas en torno al objeto y a una altura adecuada, son las encargadas de mandar a la PC varias imágenes, las cuales son procesadas mediante el software Motive para reconstruir la trayectoria, la posición y la orientación actuales del objeto observado. En la figura 64 es mostrada la interfase gráfica del programa Motive, como puede apreciarse el usuario es capaz de monitorear en tiempo real la posición de un objeto.

B.5. Visual C++

Visual Studio es un conjunto de herramientas de desarrollo para la generación de aplicaciones web ASP.NET, Servicios Web XML, aplicaciones de escritorio y aplicaciones móviles. Visual Basic, Visual C# y Visual C++ utilizan el mismo entorno de desarrollo integrado (IDE), que habilita el uso compartido de herramientas y hace más sencilla la creación de soluciones en varios lenguajes. Así mismo, dichos lenguajes utilizan las funciones de .NET Framework, las

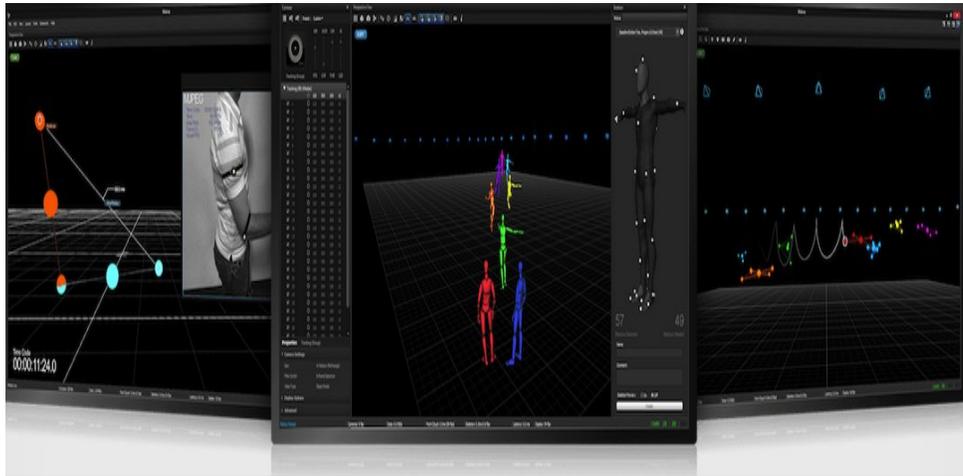


Figura 64: Programa MOTIVE de OptiTrack.

cuales ofrecen acceso a tecnologías clave para simplificar el desarrollo de aplicaciones web ASP y Servicios Web XML[28]. En este trabajo se empleó Visual C++ para generar la ley de control aplicada al UAV en ambiente controlado. Mediante el uso de librerías de OptiTrack se puede tener el estado completo de la planta a través de la interacción del programa Motive y Visual C++.

Apéndice C

C. Diagramas de flujo

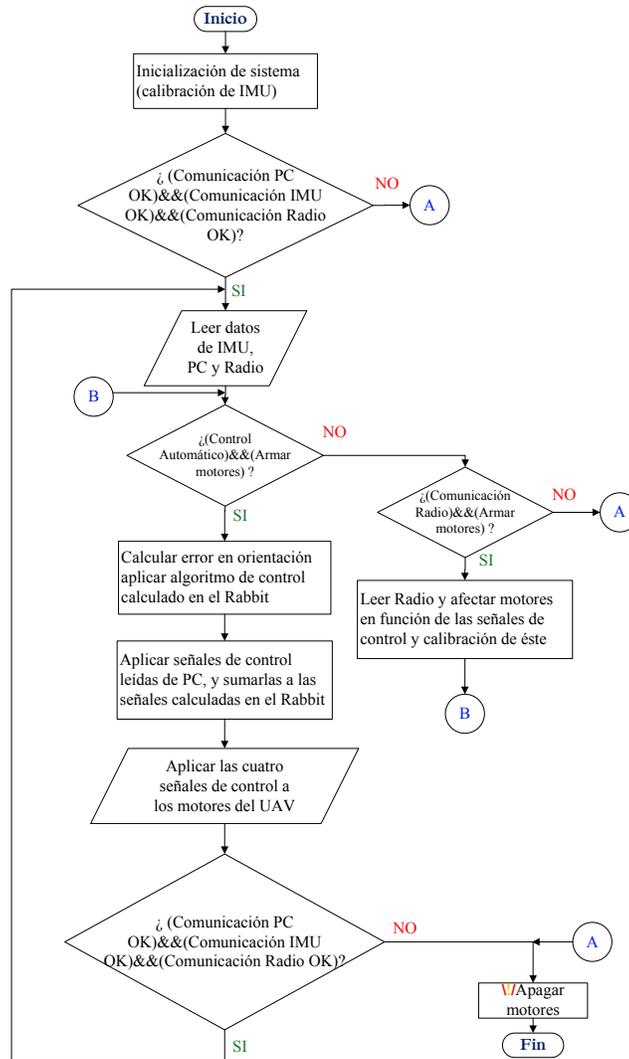


Figura 65: Diagrama de flujo del algoritmo programado en el controlador Rabbit.

C DIAGRAMAS DE FLUJO

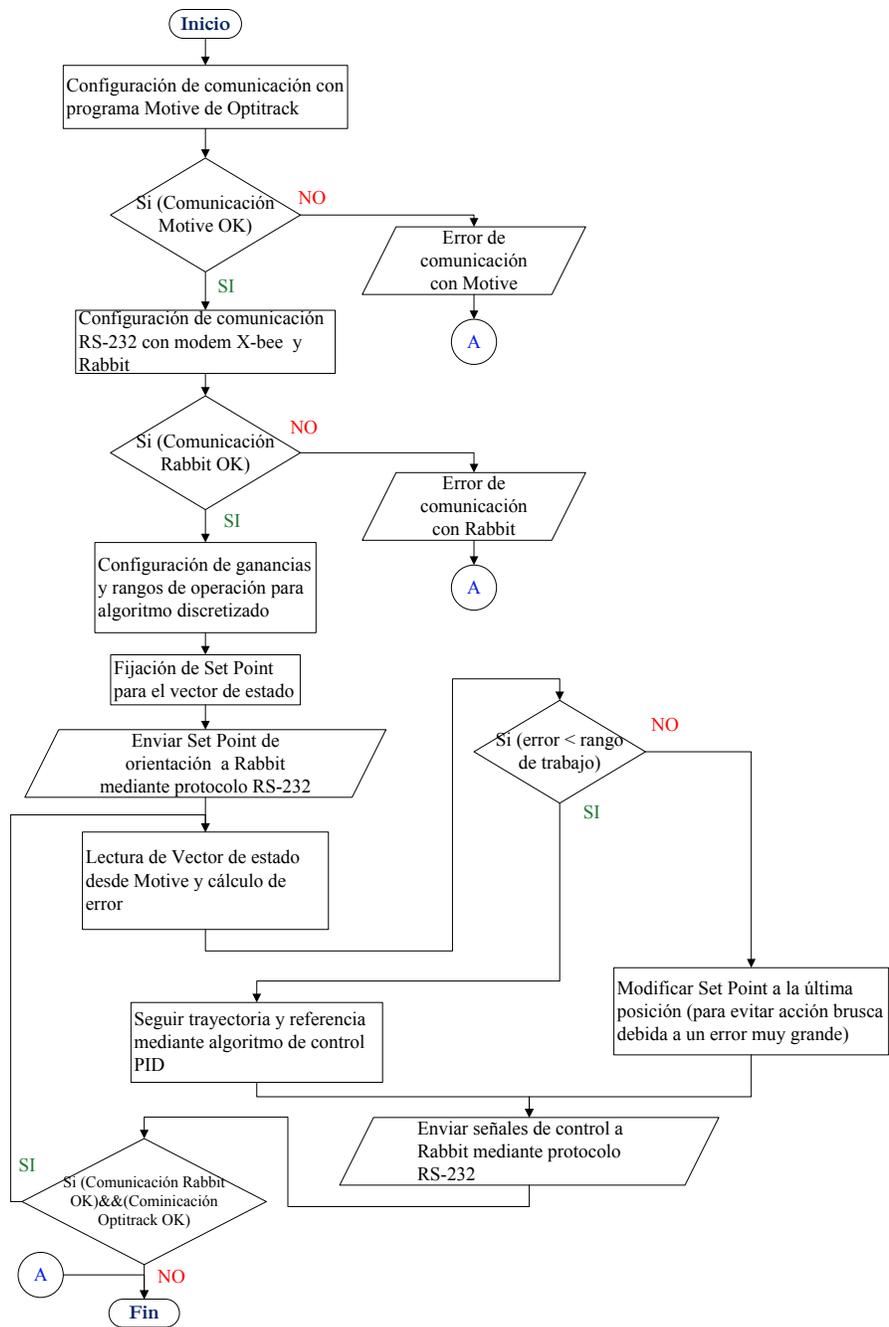


Figura 66: Diagrama de flujo del algoritmo programado en la PC.

Apéndice D

D. Programa para resolver las DARE para simulación

```
global Kx1 Kx2 Ky1 Ky2 Ku1 Ku2 g m;

g=9.81;
m=1;

%% Subsistema z psi
Az=[0 1 0 0;
    0 0 0 0;
    0 0 0 1;
    0 0 0 0]

Bz =[0 0;
     1 0;
     0 0;
     0 1]

Azps=[0 1 0 0 0 0 0 0;
      0 0 0 0 0 0 0 0;
      0 0 0 1 0 0 0 0;
      0 0 0 0 0 0 0 0;
      1 0 0 0 0 0 0 0;
      0 1 0 0 0 0 0 0;
      0 0 1 0 0 0 0 0;
      0 0 0 1 0 0 0 0]

Bzps=[0 0;
      1 0;
      0 0;
      0 1;
      0 0;
      0 0;
      0 0;
      0 0]

Czps=eye(8);
Dzps=zeros(8,2)

%discretización del sistema aumentado
sys_zpsi=ss(Azps,Bzps,Czps,Dzps)
sysd_zpsi=c2d(sys_zpsi,0.01)

Qzps= [1000 0 0 0 0 0 0 0;
       0 1000 0 0 0 0 0 0;
       0 0 1000 0 0 0 0 0;
       0 0 0 1000 0 0 0 0;
       0 0 0 0 1000 0 0 0;
       0 0 0 0 0 1000 0 0;
       0 0 0 0 0 0 1000 0;
       0 0 0 0 0 0 0 1000]
```

D PROGRAMA PARA RESOLVER LAS DARE PARA SIMULACIÓN

```
Rzps=1*[1 0;
        0 1]
        %cálculo de controlador
        %%[Kzps,Szps,Ezps] = lqr (Azps,Bzps,Qzps,Rzps)
        %%[Kzps,Szps,Ezps] =
lqrd(sysd_zpsi.a,sysd_zpsi.b,Qzps,Rzps,0.01)
[Kzps,Szps,Ezps] = lqrd(Azps,Bzps,Qzps,Rzps,0.01)

Mz= [Az      Bz;
      eye(4) zeros(4,2)]
rMz=rank(Mz)

Ku1=Kzps(:,1:4)
Ku2=Kzps(:,5:8)

%% Subsisstema x the
Ax=[0 1 0 0;
    0 0 -9.81 0;
    0 0 0 1;
    0 0 0 0]

Bx=[0 0 0 1]'

Axth=[Ax      zeros(4,4);
      eye(4) zeros(4,4)]

Bxth=[Bx; zeros(4,1)]

Qxth= [1000 0 0 0 0 0 0 0;
        0 1000 0 0 0 0 0 0;
        0 0 1000 0 0 0 0 0;
        0 0 0 1000 0 0 0 0;
        0 0 0 0 1000 0 0 0;
        0 0 0 0 0 1000 0 0;
        0 0 0 0 0 0 1000 0;
        0 0 0 0 0 0 0 1000]

Rxth=[1];

conx= ctrb (Ax,Bx)
x=rank(conx)

Mx= [Ax      Bx;
      eye(4) zeros(4,1)]

rMx=rank(Mx)
[Kxth,Sxth,Exth] = lqrd(Axth,Bxth,Qxth,Rxth,0.01)
Fxth=-Rxth^(-1)*Bxth'*Sxth;
```

D PROGRAMA PARA RESOLVER LAS DARE PARA SIMULACIÓN

```
%% Subsisstema y phi
Ay=[0 1 0 0;
    0 0 9.81 0;
    0 0 0 1;
    0 0 0 0]

By=[0 0 0 1]'

Ayph=[Ay zeros(4,4);
      eye(4) zeros(4,4)]

Byph=[By; zeros(4,1)]

Qyph= [1000 0 0 0 0 0 0 0;
        0 1000 0 0 0 0 0 0;
        0 0 1000 0 0 0 0 0;
        0 0 0 1000 0 0 0 0;
        0 0 0 0 1000 0 0 0;
        0 0 0 0 0 1000 0 0;
        0 0 0 0 0 0 1000 0;
        0 0 0 0 0 0 0 1000]

Ryph=[1];

cony= ctrb (Ay,By)
y=rank (cony)

My=[Ay By;
    eye(4) zeros(4,1)]

rMy=rank (My)
%
[Kyph, Syph, Eyph] = lqrd (Ayph, Byph, Qyph, Ryph, 0.01)
Fyph=-Ryph^(-1)*Byph'*Syph;

%%Separación de las ganancias para la parte integral y la parte derivativa
% Ku1=Ku(:,1:4)
% Ku2=Ku(:,5:8)
%
% Kpsi1=Kzps(:,1:4)
% Kpsi2=Kzps(:,5:8)

Ku1=Kzps(:,1:4)
Ku2=Kzps(:,5:8)

Kx1=Kxth(:,1:4)
Kx2=Kxth(:,5:8)

Ky1=Kyph(:,1:4)
Ky2=Kyph(:,5:8)
```

Apéndice E

E. Circuito electrónico para medir corriente

En este apéndice es descrito el circuito electrónico usado para medir la corriente consumida por el cuadricóptero en cada uno de los experimentos citados en el Capítulo 4. El principio de funcionamiento se basa en el uso del sensor ACS712, el cual permite medir corriente desde -30A hasta 30A. En función del valor monitoreado se genera una señal analógica de 0V a 5V la cual es recibida por un microcontrolador. Empleando el protocolo RS232 se envía el dato a SIMULINK de MATLAB, con lo anterior es guardada cada lectura con un intervalo de 98ms (como mínimo) en un vector, desde MATLAB a un archivo de texto, además el valor de la corriente es mostrado en una pantalla LCD. El tiempo de monitoreo puede ser ajustado por el usuario mediante tres botones, siendo uno de ellos “INC” que permite incrementar dicho tiempo, “DEC” usado para decrementar el tiempo de muestreo hasta 98ms, y “START” que es el botón con el que se valida la configuración establecida por el usuario.

En las figura (67) es mostrada la parte superior de la tarjeta electrónica armada, la figura (68) muestra la vista inferior de la PCB construida. El esquema del circuito correspondiente a dicha tarjeta es apreciado en la figura (69).



Figura 67: Circuito para adquirir corriente.

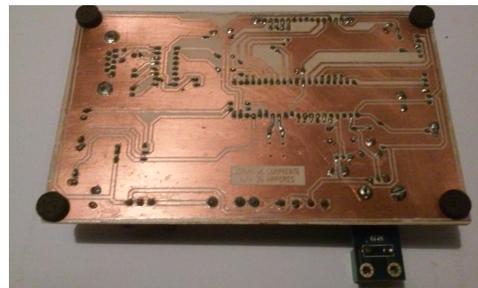
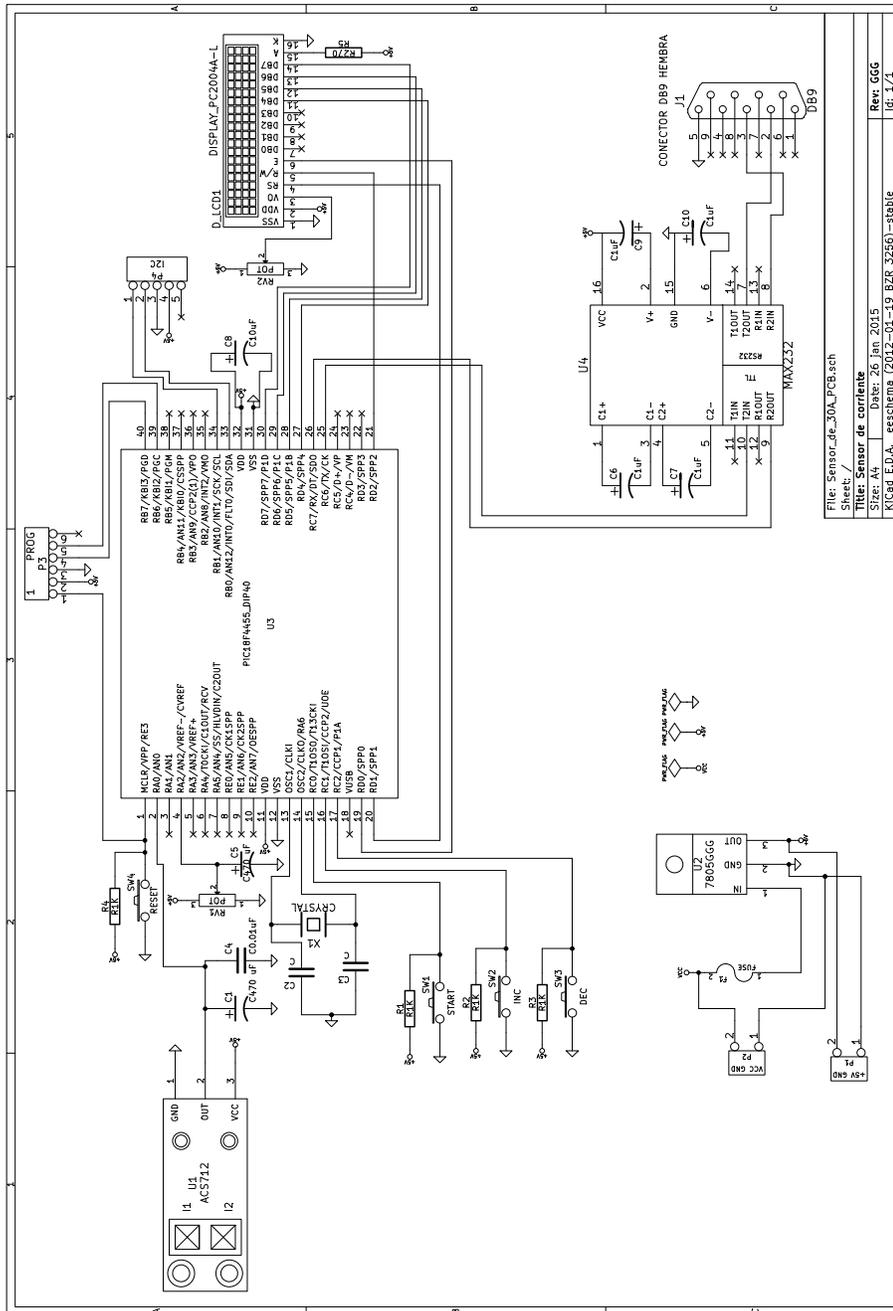


Figura 68: PCB vista inferior.

E CIRCUITO ELECTRÓNICO PARA MEDIR CORRIENTE



File: Sensor_de_30A_PCB.sch
 Sheet: /
 Title: Sensor de corriente
 Size: A4 Date: 26 Jan 2015
 KICad E.D.A. eschema (2012-01-19 BZR 3256) -stable
 Id: 1/1

Figura 69: Esquema de circuito electrónico para medir corriente.

Referencias

- [1] David Melero Cazorla. *Modelado dinámico y diseño de estrategia de control mediante estimadores para vuelo automático de un quadrotor. Caso de estudio: Control de UAV*. Tesis de maestría, Universistas Almerienses, 2012.
- [2] Escamilla Núñez, R. *Diseño, Construcción, Instrumentación y Control de un Vehículo Aéreo No Tripulado. Caso de estudio: Control de UAV*. Tesis de licenciatura, Escuela Superior de ingeniería Macánica y Eléctrica Unidad Profesional Adolfo López Mateos, Instituto Politecnico Nacional, 2010.
- [3] Salazar-Cruz S., Kendoul F., Lozano R. and Fantoni I. Real-time stabilization of a small three-rotor aircraft. *IEE Transactions on Aerospace and Electronic Systems*, 44(2):783–794, 2008.
- [4] Bouabdallah S., Noth A., Siegwart, R. Pid vs lq control tecniques applied to an indoor micro quadrotor. *IEEE Xplore International Conference on Intelligent Robots ans Systems*, pages 2451–2456, 2010.
- [5] Mokhtari A., Benallegue A. Dynamic feedback controller of euler angles and wind parameters estimation for a quadrotor unmanned aerial vehicle. *IEEE International Conference on Robotics and Automation*, pages 2359–2366, 2004.
- [6] Gonzalez I., Salazar S., Lozano R. and Escareno J. Real-time altitude robust controller for a quadrotor aircraft using sliding-mode control technique. *International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 650–659, 2013.
- [7] Endesa empieza a usar drones para revisar el estado de la red eléctrica en toda España. <http://www.legalizaciondrones.es/noticias/9-actualidad/12-endesa-empieza-a-usar-drones.html>, 2015. Consulta: 2015-10-15.
- [8] Drone helps Fukushima clean-up,. <http://www.bbc.com/news/uk-28239851>, 2014. Consulta: 2015-10-15.
- [9] EE.UU. utiliza drones para atacar civiles en tres países,. <http://www.telesurtv.net/news/EE.UU.-utiliza-drones-para-atacar-civiles-en-tres-paises-20151015-0045.html>, 2014. Consulta: 2015-10-15.
- [10] Utilizan un dron para estudiar zonas arqueológicas de Sinaloa,. <http://www.inah.gob.mx/boletines/506-utilizan-un-dron-para-estudiar-zonas-arqueologicas-de-sinaloa>, 2015. Consulta: 2015-10-15.
- [11] Científico mexicano creó drones que vuelan de manera autónoma y aprenden nuevas rutas,. <http://invdes.com.mx/tecnologia-mobil/7369-cientifico-mexicano-creo-drones-que-vuelan-de-manera-autonoma>, 2015. Consulta: 2015-10-15.
- [12] UNAM usa drones para estudios arqueológicos,. <http://archivo.eluniversal.com.mx/computacion-tecno/2015/unam-usa-drones-para-estudios-arqueologicos-99917.html>, 2015. Consulta: 2015-10-15.
- [13] UMI Trabaja con drones submarinos y exoesqueletos,. <http://www.nodaltec.am/2015/08/mexico-estudiantes-exhiben-drones-submarinos-y-exoesqueletos-en-una-estacion-del-metro/>, 2015.
- [14] H. K. Khalil. *Nonlinear Systems*. Prentice Hall Inc, 1996.
- [15] Ortega M. G. Raffo, G. V. and F. R. Rubio. An integral predictive/nonlinear h_∞ control structure for a quadrotor helicopter. *Elsevier*, 48(4):29–39, 2010.
- [16] Tahar M. Meguenni-Zemalache K and Omari A. Control of an under-actuated x4-flyer using integral backstepping controller. *Elsevier*, 2010.

REFERENCIAS

- [17] S. Bouabdallah and R. Siegwart. Control of a quadrotor. *Proceedings of the 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2007.
- [18] Al-Jarrah M. A. Al-Younes, Y. M. and A. Jhemi. Linear vs. nonlinear control techniques for a quadrotor vehicle. *Proceedings of the 7th International Symposium on Mechatronics and its Applications (ISMA10)*, 2010.
- [19] Y. R. Tang and Y. Li. Design of an optimal flight control system with integral augmented compensator for a nonlinear uav helicopter. *Proceedings of the 10th World Congress on Intelligent Control and Automation*, 2012.
- [20] ROMERO TREJO H. *Modélisation et asservissement visuel d'un mini hélicoptère*. Tesis doctoral, Université de Technologie de Compiègne Francia, 2008.
- [21] García Pérez O. *Seguimiento óptimo de trayectorias con un cuadricóptero*. Tesis de maestría, Maestría en Ciencias en Sistemas Autónomos de Navegación Aérea y Submarina, Centro de Investigación y de Estudios Avanzados del Instituto Politécnico Nacional (CINVESTAV), 2015.
- [22] Romero H. Salazar-S. Santos, O. and R. Lozano. Discrete optimal control for a quadrotor uav: Experimental approach. *International Conference on Unmanned Aircraft Systems (ICUAS)*, 2014.
- [23] Visioli A. *Practical PID Control*. Springer, 2006.
- [24] Angus P. Andrews Mohinder S. Grewal. *Kalman Filtering Theory and and Practice Using MATLAB*. WILEY & SONS, 2008.
- [25] Matía Espada F. Pinto Bermúdez E. *Fundamentos de control con MATLAB*. Prentice Hall, 2010.
- [26] Digi International Inc. *Dynamic C User's Manual*.
- [27] Programa Motive,. <http://www.optitrack.com/products/motive/>, 2015. Consulta: 2015-10-23.
- [28] Julian Templeman. *Developer step by step Microsoft Visual C++/CLI*. Octal Publishing, 2013.