



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



ESCUELA TÉCNICA
SUPERIOR INGENIEROS
INDUSTRIALES VALENCIA

TRABAJO FIN DE MASTER EN INGENIERÍA INDUSTRIAL

DISEÑO E IMPLEMENTACIÓN DE UNA PLATAFORMA PARA LA VALIDACIÓN SEGURA DE ALGORITMOS DE CONTROL AVANZADOS EN VEHÍCULOS AÉREOS NO TRIPULADOS TIPO QUADROTORS

AUTOR: VICENTE BALAGUER GARÍN

TUTOR: PEDRO GARCÍA GIL

Curso Académico: 2016-17

Resumen

El objetivo del presente trabajo de fin de máster es el desarrollo de una plataforma para la implementación segura de algoritmos de control en quadrotors. Para ello, se parte de un quadrotor ya construido, un banco de pruebas basado en una rótula y de una instalación de captura de movimiento en vivo (Optitrack).

En primer lugar, se realizará la programación del quadrotor desde cero, para conseguir un sistema totalmente modulable donde sea fácil implementar los diferentes algoritmos. Además, se creará una interfaz hombre máquina multiplataforma que permitirá comunicación bidireccional con el quadrotor y el envío de la posición y orientación del mismo obtenida del Optitrack.

En segundo lugar, se modelará el sistema para el caso de la rótula y el del vuelo libre. Se analizarán los parámetros físicos involucrados y se construirá un simulador. A partir de los modelos obtenidos, se estudiará que condiciones deben darse para que las conclusiones extraídas del control con la rótula sean extrapolables al vuelo libre.

En tercer lugar, se identificará experimentalmente el sistema, considerando la problemática causada por ser un sistema inestable y necesitar ser identificado en bucle cerrado. Con la identificación obtenida, se estimarán los parámetros inciertos del quadrotor.

Finalmente, se diseñarán e implementarán diferentes técnicas de control. Estas se validarán en primera instancia de forma segura en la rótula, y en segunda se probarán en vuelo libre. Entre los controladores diseñados cabe destacar el UDE y ADRC basados en estimación de perturbaciones, los cuales han dado excelentes resultados para el control del quadrotor.

Palabras Clave: Quadrotor, identificación de sistemas inestables, rótula, control, UDE, ADRC, Optitrack

Resum

El objectiu del present treball de fi de màster es el desenvolupament d'una plataforma per a la implementació segura de algorismes de control en quadrotors. Per a aconseguir-ho, es parteix de un quadrotor ja construït, un banc de proves basat en una ròtula i d'una instal·lació de captura de moviment en viu (Optitrack).

En primer lloc, se realitzarà la programació del quadrotor partint de zero, per aconseguir un sistema totalment modulable on fàcilment implementar els diferents algorismes. A més, es crearà un HMI multiplataforma que permetrà comunicació bidireccional amb el quadrotor i l'enviament de la posició i orientació del mateix obtinguda del Optitrack.

En segon lloc, es modelarà el sistema per al cas de la ròtula i el del vol lliure. S'analitzaran els paràmetres físics involucrats y es construirà un simulador. A partir dels models obtinguts, s'estudiarà que condicions deuen donar-se per a que les conclusions obtingudes del control amb la ròtula siguin extrapolables al vol lliure.

En tercer lloc, s'identificarà experimentalment el sistema, considerant la problemàtica causada per ser un sistema inestable i necessitar ser identificat en bucle tancat. Amb la identificació obtinguda, s'estimaran els paràmetres incerts del quadrotor.

Finalment, es dissenyaran i implementaran diferents tècniques de control. Estes es validaran en primera instància de forma segura en la ròtula, i en segona es provaran en vol lliure. Entre els controladors dissenyats es deu destacar el UDE y ADRC basats en estimació de perturbacions, els quals han donat excel·lents resultats per al control del quadrotor.

Paraules clau: Quadrotor, identificació de sistemes inestables, ròtula, control, UDE, ADRC, Optitrack

Abstract

The objective of the present master's final project is the development of a platform for the secure implementation of control algorithms in quadrotors. To do this, it is based on an already built frame, a test bench based on a ball joint and a live motion capture facility (Optitrack).

First of all, it will be done the programming of the quadrotor from scratch, to achieve a totally modularly system where it is easy to implement the different algorithms. In addition, a multi-platform HMI will be created that will allow bidirectional communication with the Quadrotor and sending the position and orientation of the device obtained from Optitrack.

Second, the system will be modeled for the ball joint and free flight case. The physical parameters involved will be analyzed and a simulator will be built. From the obtained models, it will be studied which conditions must be given so that the conclusions drawn from the control with the ball joint are given to the free flight.

Third, the system will be experimentally identified, considering the problems caused by being an unstable system and needing to be identified in a closed loop. With the identification obtained, the uncertain parameters of the quadrotor will be estimated.

Finally, different control techniques will be designed and implemented. These will be validated in the first instance in a safe way in the ball joint, and in the second they will be tested in free flight. Among the controllers designed, it is worth highlighting the UDE and ADRC based on perturbation estimation, which has given excellent results for the control of the quadrotor.

Keywords: Quadrotor, unstable systems identification, ball joint, control, UDE, ADRC, Optitrack

Índice general

Resumen	III
Índice general	IX
Índice de figuras	XIII
Índice de tablas	XVII
Índice de ecuaciones	XIX
I Memoria	1
1. Introducción	3
1.1. Objetivos del documento	3
1.2. Estructura del documento	4
2. Estado del arte	7
2.1. Introducción	7
2.2. Historia	8
2.2.1. Vehículos aéreos no tripulados	8
2.2.2. El quadrotor	9
2.3. Identificación y control	10
2.4. Normativa	10
2.5. Motivación y justificación	11
3. Descripción de la plataforma	15
3.1. Introducción	15
3.2. El quadrotor	16
3.2.1. Hardware	17
3.2.1.1. Controladores	17
3.2.1.2. Sensores	17
3.2.1.3. Actuadores	18
3.2.2. Software	19
3.3. La rótula	20
3.4. Optitrack	21
3.5. La interfaz hombre máquina	22
4. Modelado del sistema	25

4.1. Introducción	25
4.2. Modelo general del quadrotor	25
4.3. Desacoplamiento	27
4.3.1. Control en “cruz” y “equis”	27
4.4. Modelo del quadrotor en la rótula	28
4.5. Obtención de los parámetros físicos	29
4.6. Simulador	30
4.7. Linealización y representaciones	31
4.8. Comparativa rótula/vuelo libre	32
5. Identificación	35
5.1. Introducción	35
5.2. Situaciones posibles	36
5.3. Identificación en bucle cerrado	37
5.4. Mínimos cuadrados recursivos	38
5.4.1. X-CLOE	38
5.5. Diseño del experimento	39
5.5.1. Pseudo Random Binary Sequence	40
5.5.2. Preprocesado de los datos	41
5.6. Resultados	41
5.7. Obtención de los parámetros inciertos	43
6. Control	47
6.1. Introducción	47
6.2. Compromiso especificaciones/robustez	48
6.3. Control con dos grados de libertad	49
6.4. Control basado en funciones de transferencia	51
6.4.1. Control mediante tablas	51
6.4.2. Control PID	53
6.4.3. Reguladores algebraicos	54
6.4.4. IMC	56
6.5. Control basado en espacio de estados	58
6.5.1. Asignación de polos	58
6.5.2. Control LQR	59
6.6. Control basado en estimación de las perturbaciones	60
6.6.1. ADRC	61
6.6.2. UDE	63
6.7. Control de posición	65
7. Validación	67
7.1. Introducción	67
7.2. Comportamiento en la rótula	67
7.2.1. Controladores descartados	68
7.2.2. Control PID	68
7.2.3. Asignación de polos	71
7.2.4. Control LQR	73

7.2.5. ADRC	75
7.2.6. UDE	77
7.2.7. Análisis	79
7.3. Comportamiento en vuelo libre	80
7.3.1. Control LQR	80
7.3.2. ADRC	82
7.3.3. UDE	85
7.3.4. Análisis	88
8. Conclusiones y trabajos futuros	89
8.1. Conclusiones	89
8.2. Trabajos futuros	90
II Presupuesto	91
9. Presupuesto	93
9.1. Introducción	93
9.2. Cuadro de precios	94
9.3. Cuadro de precios unitarios	95
9.4. Cuadro de precios descompuestos	96
9.5. Presupuesto parcial	98
9.6. Presupuesto de ejecución, por contrata y base de licitación	99
Bibliografía	101

Índice de figuras

2.1. Prototipo de misil guiado usado en la primera guerra mundial [6]	8
2.2. El Gyroplane No. 1 de Bréguet-Richet, 1907 [8]	9
2.3. Prototipo de taxi autónomo para Dubai [12]	10
2.4. Quadrotors desarrollados en el laboratorio	11
2.5. Quadrotor controlado en la rótula	12
2.6. Quadrotor en vuelo libre	13
3.1. Esquema global de la plataforma	16
3.2. Quadrotor utilizado	16
3.3. Detalle de las placas del quadrotor	17
3.4. Detalle de los motores y los drivers	18
3.5. Rótulas (bancos de pruebas) del laboratorio	20
3.6. Programa Motive de Optitrack	21
3.7. Instalación del sistema Optitrack	22
3.8. Detalle de una cámara de la instalación Optitrack	22
3.9. HMI desarrollado para la plataforma	23
4.1. Sistema de referencia del quadrotor	26
4.2. Configuraciones en “cruz” y “equis”	28
4.3. Sistema de referencia de la rotula	28
4.4. Simulador desarrollado en Simulink. Quadrotor	30
4.5. Simulador desarrollado en Simulink. Entorno de control	31
5.1. Situación 1	36
5.2. Situación 2	36
5.3. Situación 3	37
5.4. Situación 4	37
5.5. X-CLOE	39
5.6. Pseudo Random Binary Sequence	40
5.7. Señal PRBS generada	41
5.8. Identificación del yaw	42
5.9. Identificación del pitch	42
5.10. Identificación del yaw	42
5.11. Error de predicción del roll	42
5.12. Error de predicción del pitch	42
5.13. Error de predicción del yaw	43
5.14. Ajuste del roll	44

5.15. Ajuste del pitch	44
5.16. Ajuste del yaw	44
6.1. Bode de los dos modelos considerados	49
6.2. Bucle cerrado resultante	49
6.3. Control con dos grados de libertad	50
6.4. Control con dos grados de libertad. Formulación alternativa	50
6.5. Control 2dof. Salida	50
6.6. Control 2dof. Acción de control	51
6.7. Control mediante tablas. Salida	52
6.8. Control mediante tablas. Acción de control	52
6.9. Control PD. Salida	54
6.10. Control PD. Acción de control	54
6.11. Regulador algebraico. Salida	55
6.12. Regulador algebraico. Acción de control	55
6.13. Internal model control	56
6.14. IMC modificado	56
6.15. IMC. Salida	57
6.16. IMC. Acción de control	57
6.17. Realimentación del estado	58
6.18. Asignación de polos en espacio de estados. Salida	59
6.19. Asignación de polos en espacio de estados. Acción de control	59
6.20. LQR. Salida	60
6.21. LQR. Acción de control	60
6.22. Estimación de perturbaciones	61
6.23. ADRC. Salida	62
6.24. ADRC. Acción de control	62
6.25. ADRC. Perturbación estimada	63
6.26. UDE. Salida	64
6.27. UDE. Acción de control	64
6.28. UDE. Perturbación estimada	65
6.29. Control de posición. Estrategia en cascada	65
7.1. Referencias del experimento realizado en la rótula.	68
7.2. PD-roll. Experimento rótula	69
7.3. PD-pitch. Experimento rótula	69
7.4. PD-yaw. Experimento rótula	69
7.5. PD-ur. Experimento rótula	69
7.6. PD-up. Experimento rótula	69
7.7. PD-uy. Experimento rótula	70
7.8. PID-roll. Experimento rótula	70
7.9. PID-pitch. Experimento rótula	70
7.10. PID-yaw. Experimento rótula	70
7.11. PID-ur. Experimento rótula	71
7.12. PID-up. Experimento rótula	71
7.13. PID-uy. Experimento rótula	71

7.14. Asignación polos-roll. Experimento rótula	72
7.15. Asignación polos-pitch. Experimento rótula	72
7.16. Asignación polos-yaw. Experimento rótula	72
7.17. Asignación polos-ur. Experimento rótula	72
7.18. Asignación polos-up. Experimento rótula	72
7.19. Asignación polos-uy. Experimento rótula	73
7.20. LQR-roll. Experimento rótula	73
7.21. LQR-pitch. Experimento rótula	73
7.22. LQR-yaw. Experimento rótula	74
7.23. LQR-ur. Experimento rótula	74
7.24. LQR-up. Experimento rótula	74
7.25. LQR-uy. Experimento rótula	74
7.26. ADRC-roll. Experimento rótula	75
7.27. ADRC-pitch. Experimento rótula	75
7.28. ADRC-yaw. Experimento rótula	75
7.29. ADRC-ur. Experimento rótula	76
7.30. ADRC-up. Experimento rótula	76
7.31. ADRC-uy. Experimento rótula	76
7.32. ADRC-Estimación perturbación ur. Experimento rótula	76
7.33. ADRC-Estimación perturbación up. Experimento rótula	76
7.34. ADRC-Estimación perturbación uy. Experimento rótula	77
7.35. UDE-roll. Experimento rótula	77
7.36. UDE-pitch. Experimento rótula	77
7.37. UDE-yaw. Experimento rótula	78
7.38. UDE-ur. Experimento rótula	78
7.39. UDE-up. Experimento rótula	78
7.40. UDE-uy. Experimento rótula	78
7.41. UDE-Estimación perturbación ur. Experimento rótula	79
7.42. UDE-Estimación perturbación up. Experimento rótula	79
7.43. UDE-Estimación perturbación uy. Experimento rótula	79
7.44. LQR-roll. Vuelo libre	80
7.45. LQR-pitch. Vuelo libre	80
7.46. LQR-yaw. Vuelo libre	81
7.47. LQR-ur. Vuelo libre	81
7.48. LQR-up. Vuelo libre	81
7.49. LQR-uy. Vuelo libre	81
7.50. LQR X. Vuelo libre	82
7.51. LQR Y. Vuelo libre	82
7.52. LQR Z. Vuelo libre	82
7.53. ADRC-roll. Vuelo libre	82
7.54. ADRC-pitch. Vuelo libre	82
7.55. ADRC-yaw. Vuelo libre	83
7.56. ADRC-ur. Vuelo libre	83
7.57. ADRC-up. Vuelo libre	83
7.58. ADRC-uy. Vuelo libre	83

7.59.ADRC-Estimación perturbación ur. Vuelo libre	84
7.60.ADRC-Estimación perturbación up. Vuelo libre	84
7.61.ADRC-Estimación perturbación uy. Vuelo libre	84
7.62.ADRC X. Vuelo libre	84
7.63.ADRC Y. Vuelo libre	84
7.64.ADRC Z. Vuelo libre	85
7.65.UDE-roll. Vuelo libre	85
7.66.UDE-pitch. Vuelo libre	85
7.67.UDE-yaw. Vuelo libre	85
7.68.UDE-ur. Vuelo libre	86
7.69.UDE-up. Vuelo libre	86
7.70.UDE-uy. Vuelo libre	86
7.71.UDE-Estimación perturbación ur. Vuelo libre	86
7.72.UDE-Estimación perturbación up. Vuelo libre	86
7.73.UDE-Estimación perturbación uy. Vuelo libre	87
7.74.UDE X. Vuelo libre	87
7.75.UDE Y. Vuelo libre	87
7.76.UDE Z. Vuelo libre	87

Índice de tablas

4.1. Parámetros estimados	30
5.1. Coeficientes de los modelos	43
5.2. Parámetros físicos identificados	45
6.1. Parámetros de ajuste mediante el método de las tablas	53
6.2. Parámetros de ajuste de los PID	53
6.3. Coeficientes de los reguladores algebraicos	56
6.4. Coeficientes del IMC	57
6.5. Realimentación del estado por asignación de polos	58
6.6. Realimentación del estado por LQR	60
6.7. Observador de la perturbación. ADRC	62
6.8. Observador de la perturbación. UDE	64
9.1. Cuadro de precios de la mano de obra	94
9.2. Cuadro de precios de los materiales	94
9.3. Cuadro de precios de la maquinaria	95
9.4. Capítulo 1	95
9.5. Capítulo 2	95
9.6. Capítulo 3	95
9.7. C1.UO1. Quadrotor	96
9.8. C1.UO2. Rótula	96
9.9. C1.UO3. Instalación Optitrack	96
9.10. C2.UO1. Software del quadrotor	97
9.11. C2.UO2. Interfaz hombre máquina	97
9.12. C3.UO1. Diseño del simulador	97
9.13. C3.UO2. Identificación del sistema	98
9.14. C3.UO3. Diseño y validación de controladores	98
9.15. Capítulo 1	98
9.16. Capítulo 2	98
9.17. Capítulo 3	99
9.18. Presupuesto por capítulos	99
9.19. Presupuesto de ejecución, por contrata y base de licitación	99

Índice de ecuaciones

4.1. Modelo del quadrotor en vuelo libre	26
4.2. Linealizado de la fuerza de los motores	26
4.3. Descomposición de la fuerza de un motores	27
4.4. Descomposición de las fuerzas de los motores	27
4.5. Cambio de variable para desacoplar el control	27
4.6. Modelo desacoplado del quadrotor	27
4.7. Modelo unidimensional de la rótula	29
4.8. Modelo general de la rotula	29
4.9. Modelo linealizado de la rotula	31
4.10. Función de transferencia unidimensional de la rotula	31
4.11. Modelo en espacio de estados	32
4.12. Función de transferencia discreta de la rotula	32
4.13. Función de transferencia discreta de la rotula, parámetro a	32
4.14. Modelo discreto en espacio de estados	32
4.15. Ecuación característica del bucle cerrado en espacio de estados	32
4.16. Resultado del determinante	32
4.17. Coeficientes de la ecuación característica	33
4.18. Coeficientes de la ecuación característica	33
5.1. Función de transferencia discreta, ejemplo de mínimos cuadrados	36
5.2. Ejemplo de mínimos cuadrados	36
5.3. Mínimos cuadrados recursivos	38
5.4. X-CLOE	39
5.5. Función de transferencia de la identificación	43
6.1. Compromiso especificaciones/robustez. Planta real	48
6.2. Compromiso especificaciones/robustez. Planta identificada	48
6.3. Controlador PID tablas roll/pitch	51
6.4. Modelo roll/pitch según las tablas	51
6.5. Parámetros tabla roll/pitch	51
6.6. Controlador PID tablas yaw	52
6.7. Modelo yaw según las tablas	52
6.8. Parámetros tabla yaw	53
6.9. PID discreto utilizado en los quadrotors	53
6.10. Ecuación del bucle cerrado	54
6.11. Discretización de las especificaciones	55
6.12. Controlador algebraico obtenido	55
6.13. Función prototipo del IMC	57

6.14. Valores propios en bucle cerrado	58
6.15. Función a minimizar en el LQR	59
6.16. Modelo discreto ampliado	61
6.17. Observador del estado reducido (I)	61
6.18. Valores propios en bucle cerrado	61
6.19. Observador del estado reducido (II)	62
6.20. Modelo del sistema. UDE	63
6.21. Obtención de la expresión del UDE (I)	63
6.22. Obtención de la expresión del UDE (II)	63
6.23. Estimación de la perturbación. UDE	63

Parte I

Memoria

Capítulo 1

Introducción

1.1 Objetivos del documento

El presente documento se expone la labor realizada para el trabajo de fin de máster “Diseño e implementación de una plataforma para validación segura de algoritmos de control avanzados en vehículos aéreos no tripulados tipo quadrotors” de la titulación de Máster Universitario en Ingeniería Industrial cursada en la Escuela Técnica Superior de Ingenieros Industriales de la Universidad Politécnica de Valencia durante los años 2015-2017.

El proyecto consiste en la realización de una plataforma para validación segura de algoritmos de control en quadrotors. Mediante un banco de pruebas, se permitirá validar los algoritmos sin riesgo para el usuario, para posteriormente probarlos en vuelo libre. Junto a ello, se creará una sistemática de identificación para obtener el modelo experimental del vehículo y se estudiarán diferentes algoritmos de control.

El trabajo surge a partir de los numerosos trabajos realizados en el grupo de investigación dirigido por el Dr. Pedro García Gil sobre quadrotors y la necesidad de crear un marco común que permita validar diferentes estrategias de control. Además, recientemente el grupo ha obtenido una instalación de captura de movimiento en vivo, lo cual permite obtener medidas de la orientación y posición del quadrotor.

Los objetivos que se pretender conseguir son los siguientes:

- Desarrollo del software del quadrotor. Dicho software deberá ser en tiempo real permitir cumplir con los tiempos de control y será fácilmente modificable para modificar cualquier parte del mismo sin necesidad de modificar las restantes y permitirá el uso de cualquier estructura de vehículo multirotor.
- Desarrollo de un banco de pruebas o rótula que permita realizar el control de orientación del vehículo de forma segura tanto para el quadrotor como el propio usuario.

- Desarrollo de una interfaz hombre máquina multiplataforma que permita comunicarse con el quadrotor y con el programa de captura de movimiento.
- Estudio del modelo teórico del quadrotor en vuelo libre y en la rótula, analizando sus diferencias y semejanzas.
- Obtención de una sistemática de identificación experimental de los parámetros del quadrotor para su posterior uso en control.
- Diseño de diferentes técnicas de control a partir de los modelos identificados e implementación en el vehículo para controlarlo en la rótula y en vuelo libre. Se estudiarán desde técnicas básicas hasta técnicas basadas en estimación de perturbaciones.

1.2 Estructura del documento

A continuación se presentan los diferentes capítulos de los que se compone la memoria del trabajo de fin de máster.

- Capítulo 1. Introducción.
Presenta una breve introducción al trabajo de fin de máster y describe los diferentes capítulos que lo componen.
- Capítulo 2. Estado del arte
Expone el estado del arte de los vehículos aéreos no tripulados, los quadrotors y la identificación y control de este tipo de vehículos. Se presenta la motivación y justificación del trabajo.
- Capítulo 3. Descripción de la plataforma.
Describe la plataforma desarrollada con todos los elementos: el quadrotor (hardware y software), la rótula, el sistema de captura de movimiento (Optitrack) y la interfaz hombre máquina.
- Capítulo 4. Modelado del sistema.
Presenta los modelos teóricos del quadrotor para el vuelo libre y la rótula. Se presentan sus versiones lineales y se estiman los valores de los parámetros físicos. Finalmente se hace un estudio comparativo para determinar si se pueden extraer conclusiones en la rótula que sirvan para el vuelo libre.
- Capítulo 5. Identificación.
Determina la problemática que supone la identificación del quadrotor, diseñando una sistemática que lo permita identificar. Una vez identificado, se estiman los parámetros físicos corregidos del modelo teórico.
- Capítulo 6. Control.
A partir del modelo identificado, se diseñan diferentes controladores de diferentes índoles. Estos controladores son validados en primera instancia en un simulador desarrollado en Simulink, permitiendo ejecutar el mismo control que en la plataforma real.
- Capítulo 7. Validación.
Los controladores diseñados en el capítulo anterior son validados en la realidad, en primer lugar en la rótula y en segundo en vuelo libre.

- Capítulo 8. Conclusiones y trabajos futuros.

Con todos los resultados obtenidos en los diferentes capítulos, se analizan los resultados obtenidos y se presentan posibles futuros trabajos que permitan usar la plataforma diseñada.

Capítulo 2

Estado del arte

2.1 Introducción

Durante los últimos años, el uso de los vehículos aéreos no tripulados (UAVS), remotamente tripulados (RPAS) [1] o más comúnmente conocidos como drones se han visto consolidados en numerosas aplicaciones de uso civil. Como consecuencia, la Unión Europea los ha definido como un sector estratégico a desarrollar dentro del programa HORIZONTE 2020 [2], destacando el elevado impacto que pueden tener dichos vehículos a mediano y largo plazo en diferentes sectores de la actividad socioeconómica.

Entre los principales nichos de mercado, destacar el elevado potencial en aplicaciones de cartografía, agricultura, protección contra incendios, y en general, todas aquellas actividades donde el uso de estos dispositivos pueda suponer una mejora del rendimiento y seguridad del proceso [3]. En este aspecto, se espera que el uso de estas aeronaves se vea incrementado de forma exponencial en las próximas décadas. No obstante, aunque se ha avanzado mucho en los últimos años, todavía queda pendiente dotar a estos vehículos del suficiente grado de autonomía y reactividad que permita un uso seguro en situaciones donde pueda interactuar con otras aeronaves y/o personas [4].

Dentro de los vehículos aéreos remotamente tripulados se encuentran los quadrotors. Los quadrotors son un tipo de UAVS multirotor formado por cuatro rotores fijos que mediante la modificación relativa de sus velocidades producen su movimiento. Esta subclase de UAVS es la más utilizada en el ámbito civil, caracterizándose por emplear cuatro motores para su control y su variedad de tamaños (desde gramos hasta decenas de kilogramos de fuerza de empuje). No obstante, el estudio en particular de los quadrotors frente a otros tipos aeronaves no resta generalidad a las estrategias diseñadas [5], permitiendo un estudio pormenorizado de su comportamiento fácilmente extrapolable a todo tipo de UAVS.

2.2 Historia

Los primeros quadrotors datan de principios del siglo XX. Aunque debido a su complejidad de pilotaje inicialmente fueron un fracaso, sin embargo a lo largo del siglo surgieron nuevas disciplinas (UAVS) y técnicas de control, las cuales permitieron al quadrotor llegar a ser el vehículo de hoy en día.

2.2.1 Vehículos aéreos no tripulados

Los UAVS surgieron principalmente como arma bélica. En la primera guerra mundial se crearon los primeros dispositivos (figura 2.1) que se consideran UAVS [6]. Estas primeras maquinas permitían preestablecer una determinada trayectoria que era seguida gracias a unos primitivos giroscopios. En los siguientes años se siguió investigando en el campo, destacando los misiles alemanes V1 y V2 desarrollados durante la segunda guerra mundial por Von Braun. Estos misiles, mejoraban considerablemente la precisión y alcance de los primeros, siendo el precursor de los cohetes espaciales que llevarían el hombre a la luna [7]. Sin embargo, estas armas tenían carácter experimental, no llegándose a producir en masa.

En la guerra de Vietnam, se desarrollaron drones que permitían tomar imágenes de diferentes zonas para estudiar diferentes alternativas en las guerras. Este tipo de drones ya se fabricaron industrialmente, permitiendo tomar ventaja en batalla. En los siguientes años gracias al desarrollo del transistor y más tarde del chip, la complejidad de estos dispositivos fue en aumento llegando a los UAVS actuales [6].



Figura 2.1: Prototipo de misil guiado usado en la primera guerra mundial [6]

En cuanto a la evolución, se pueden clasificar en tres etapas. En una etapa inicial, los UAVS eran dispositivos diseñados para hacer un determinado blanco. En una segunda, se usaban principalmente como sensor, permitiendo obtener datos de una zona determinada. Finalmente, en una última etapa se considera su uso como herramienta, permitiendo la interacción con el entorno de múltiples formas.

En un inicio las aeronaves tipo avión/torpedo dominaron la aplicación. No obstante, conforme se desarrollo la tecnología se vio que otros tipo de aeronaves, como por ejemplo los quadrotors,

suponían una ventaja en determinadas situaciones [3] como su relativo tamaño o permanecer inmóvil en vuelo.

2.2.2 *El quadrotor*

Como la mayoría de aeronaves, los primeros quadrotors surgieron la inicio del siglo XX como propuesta de medio de transporte aéreo. El primer quadrotor (figura 2.2) del que se tiene constancia de su vuelo data de 1907 diseñado por los ingenieros franceses Charles Richet y Louis Bréguet [8]. Este vehículo se pilotaba de forma totalmente manual, necesitándose un piloto con gran habilidad para realizar vuelos. Por tanto, debido su complejidad de control y la invención de aeronaves más sencillas de pilotar como el helicóptero o el avión, los quadrotors quedaron parcialmente en el olvido.

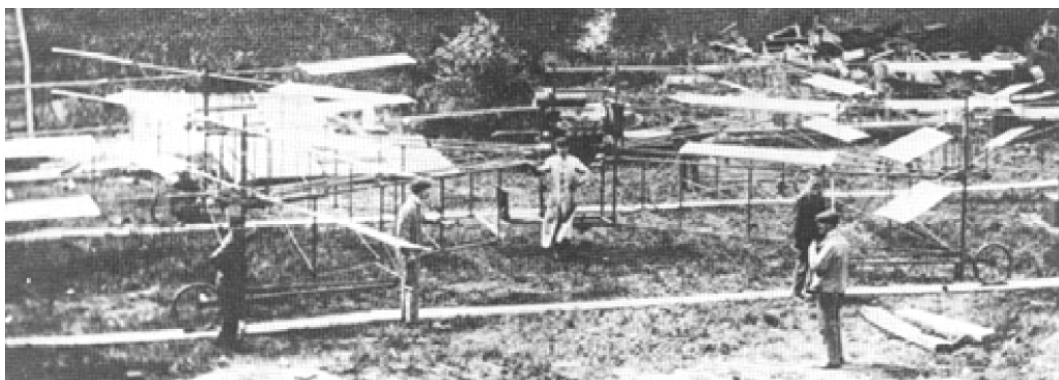


Figura 2.2: El Gyroplane No. 1 de Bréguet-Richet, 1907 [8]

Durante las siguientes décadas surgieron diferentes proyectos para construir comercialmente quadrotors, destacando el proyecto de Convertawings [9] que creo un quadrotor comercial que permitía sustentar a una carga de 4,9t, con un peso total de 19t y una velocidad máxima de 278Km/h. No obstante, tubo escaso éxito comercial y no se llegaron a comercializar.

No sería hasta 1996, cuando el DraganFlyer radio control apareciera. Debido a los avances tecnológicos, se pudo construir un quadrotor de pequeño tamaño que disponía de un control rudimentario, aunque se necesitaban buenas habilidades para su manejo [10]. En los siguientes años continuarían saliendo nuevos modelos, tecnologías, estrategias hasta llegar a la situación actual donde ya están integrados dentro de los UAVS.

Cabe señalar la evolución que han tenido estos vehículos respecto su orientación y tamaño, siendo los primeros maquinas de gran tonelaje y existiendo quadrotors que caben en la palma de la mano [11]. En contraste con la evolución, una vez solucionados los problemas iniciales, se esta volviendo a investigar su uso como medio de transporte, debido a su facilidad de construcción y despegue vertical. Por ejemplo, en la figura 2.3 se puede observar uno de los taxis autónomos que se estan diseñando para Dubai [12].



Figura 2.3: Prototipo de taxi autónomo para Dubai [12]

2.3 Identificación y control

Debido a los problemas de control del DraganFlyer y derivados, el control de este aparato centro la atención de numerosos grupos de investigación. A del éxito de estos aparatos, obtuvieron el centro de atención de parte de la comunidad investigadora, dedicándose a mejorar el control básico del mismo [13], su modelado [10], o su construcción física [14]. Como consecuencia de su éxito, surgieron multitud de proyectos de quadrotors e investigaciones.

Una vez resuelto el problema inicial, surgieron numerosos problemas de control considerando diferentes situaciones como la presencia de grandes perturbaciones [15], maniobras agresivas [16], seguimiento de trayectorias [17] o tareas colaborativas [11].

En lo referente al modelo usado para el control, en la mayoría de casos se utiliza el modelo simplificado del quadrotor con los parámetros obtenidos del fabricante [18] o identificación de las partes de manera independiente [19], [20]. Sin embargo, en la literatura se pueden encontrar diferentes estudios de identificación del quadrotor mediante técnicas de caja negra [21].

2.4 Normativa

Debido al incremento de popularidad de los vehículos aéreos remotamente tripulados (RPAS) se vio que era necesario empezar a legislar su uso. En Julio de 2014 se realizó la primera normativa que legislaba el vuelo de los RPAS en España [22] y que es de actual aplicación. No obstante, se está trabajando en una actualización de la normativa vigente para 2017, que está en fase de borrador, que ampliará las situaciones consideradas en la primera [23]. También a nivel Europeo se está trabajando en una directiva que permitirá establecer unas pautas en la legislación a aplicar en los países de la Unión.

En España los RPAS se clasifican según su capacidad máxima en despegue: mayores de 150kg, entre 150kg y 25kg, entre 25kg y 2kg y menores de 2kg. Los RPAS mayores de 150kg se rigen por la normativa tradicional de vehículos aéreos. Los segundos, necesitan de una licencia oficial de vuelo concedida por la agencia Estatal de Seguridad aérea. Los menores de 25kg y mayores de

2kg solo necesitan un carnet siempre que se cumplan unos requisitos mínimos. Los menores a 2kg se pueden volar en zonas habilitadas teniendo el carnet correspondiente. La matriculación y obtención del certificado de aeronavegabilidad es necesaria para los RPAS mayores a 2kg.

Las zonas aptas para el vuelo vienen delimitadas por la categoría del aparato y capacidad del piloto del mismo. No está permitido el vuelo en zonas donde no se tenga visión directa del aparato, excepto en los casos considerados. Se establecen restricciones para permitir operaciones sobre aglomeraciones de edificios en ciudades, pueblos o lugares habitados o reuniones de personas al aire libre. En el borrador propuesto se consideran situaciones donde se permite el vuelo nocturno, así como el vuelo en espacio aéreo controlado siempre que se notifique a las autoridades y se disponga de un radiofonista acreditado.

2.5 Motivación y justificación

El origen del presente trabajo de fin de máster surge a partir de los numerosos trabajos de fin de titulación realizados en los últimos años en el grupo de investigación.

En [24] se construye el predecesor del quadrotor actual, el cual permitía conectarse por Wifi con la interfaz hombre maquina (HMI) del PC/estación base. Este quadrotor se diferenciaba del actual por disponer de un microcomputador IGEPv2 como unidad de procesamiento central y un Arduino Due como interfaz entre el hardware del dispositivo. Debido a diferentes problemas con su estructura, en [25] se investigó el uso de la Raspberry Pi en lugar de la IGEPv2. El quadrotor resultante permitió reducir los componentes programables, siendo la Raspberry Pi encargada de la comunicación con la estación base y el hardware. En [26] se actualizó el quadrotor mediante el empleo de la Raspberry Pi 2. En la figura 2.4 se pueden observar algunos de los quadrotors desarrollados en el laboratorio.



Figura 2.4: Quadrotors desarrollados en el laboratorio

El primer banco de pruebas (rótula, ver sección 3.3) surgió en [25] como necesidad de efectuar diferentes ensayos de forma segura. Esta primera plataforma solo disponía de un grado de libertad, no obstante [26]-[28] diseñaron diferentes dispositivos que experimentar de forma segura en

los tres grados de libertad. A partir de [29] también se diseñó un banco de pruebas basado en un giroscopio, el cual permite maniobras de más de 360° .

La primera interfaz hombre máquina desarrollada para [24] era unidireccional, permitiendo el envío de datos hacia el quadrotor. Estos datos eran básicamente las constantes del PID y los modos del quadrotor. En [30] se actualizó el programa para permitir comunicación bidireccional, seguimientos de trayectorias o implementación del protocolo MavLink entre otras mejoras. Sin embargo, el programa era muy complejo de utilizar. Por eso, en [31] se realizó una versión mejorada del primer HMI incluyendo la comunicación bidireccional. Todos estos HMI fueron programados en C++ usando QT y la compilación realizada sobre Ubuntu.

El principal objetivo de este trabajo es la realización de una plataforma que unifique los temas tratados en los diferentes trabajos bajo una plataforma integral que permita la implementación segura y sistemática de diferentes algoritmos de control en quadrotors. Junto los elementos descritos, cabe destacar que recientemente el grupo de investigación ha obtenido una instalación que permite la captura de movimiento online (Optitrack, ver sección 3.4). En primera instancia los algoritmos se evaluarán de forma segura en la rótula para una vez validados emplearlos en vuelo libre junto el Optitrack.

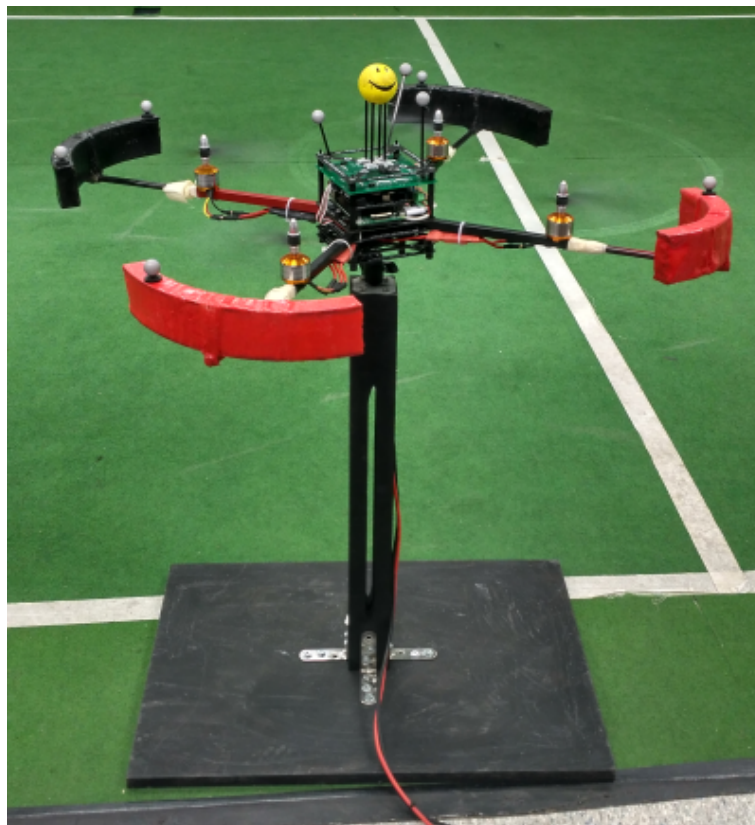


Figura 2.5: Quadrotor controlado en la rótula

Para ello se partirá de la última versión de quadrotor realizada y una de las rotulas (banco de pruebas) del laboratorio. En primer lugar todo el software será rehecho, incluyendo el HMI donde se programará un software multiplataforma que permitirá la comunicación con el Optitrack. El

software resultante será completamente modulable, por tanto las diferentes partes serán fácilmente modificables sin afectar las restantes. Una vez obtenido la plataforma base, se diseñará un procedimiento sistemático para permitir identificar los diferentes parámetros del quadrotor, permitiendo el diseño y su uso como plataforma segura de validación de estrategias de control. En la figura 2.5 se muestra el quadrotor y la rotula empleados en el trabajo y en la figura 2.6 se muestra el quadrotor en vuelo libre usando las medidas del Optitrack.

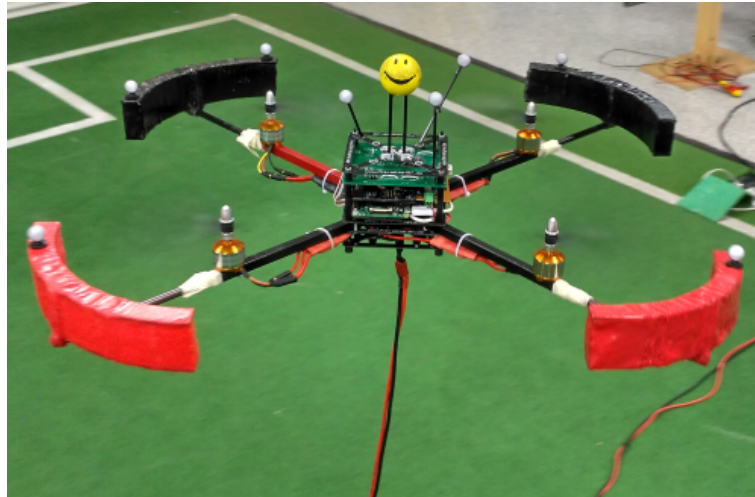


Figura 2.6: Quadrotor en vuelo libre

Descripción de la plataforma

3.1 Introducción

La plataforma desarrollada consiste del conjunto de elementos de hardware y software que permiten efectuar experimentos con el quadrotor. Dicha plataforma posee las siguientes características:

- Facil manipulación del software. El software se encuentra modulado de forma que para modificar una parte no sea necesario modificar las restantes.
- Hardware modulable. El software permite múltiples configuraciones de multirrotores y el cambio de sensores modificando unicamente su driver.
- Plataforma segura de pruebas. La rotula permite evaluar reguladores de forma segura previo a vuelo libre, al estar limitado el alcance del vehículo.
- Sistema operativo en tiempo real. El sistema operativo del quadrotor es de tiempo real, por tanto las tareas que se ejecuten se efectuarán dentro de los tiempos preestablecidos.
- Formulación de una identificación sistemática. Se ha diseñado una formulación sistemática de identificación del sistema, obteniendo los parámetros del vehículo empleados para el diseño de controladores.
- Interfaz hombre máquina (HMI) multiplataforma. La interfaz programada en Java se puede ejecutar en cualquier sistema operativo que disponga de la maquina virtual de Java.
- Conexión mediante wifi a la estación tierra. El quadrotor genera una red wifi con la que la estación a ejecutar el HMI se conecta.
- Obtención de medidas con el software Optitrack. Opcionalmente el programa de visión por computador permite obtener las medidas de posición y orientación del vehículo. El HMI permite su envío al quadrotor.

En la figura 3.1 se encuentra la estructura global del sistema. Cabe destacar que exceptuando el programa de visión Optitrack la estructura global es como las de [26]. No obstante, la mayoría

de elementos de software han sido rediseñados, así como algunos cambios en los componentes físicos.

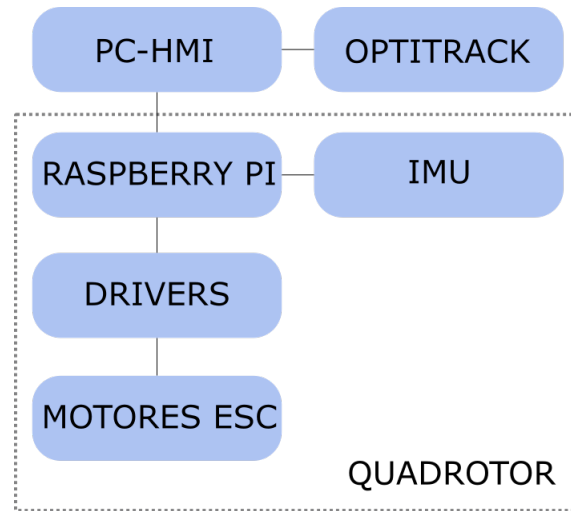


Figura 3.1: Esquema global de la plataforma

3.2 El quadrotor

Entre los objetivos del presente trabajo no se encuentra la construcción de un quadrotor, al ser un tema frecuentemente tratado en los últimos años dentro del grupo de investigación. Por tanto, el quadrotor utilizado (figura 3.2) es una versión modificada del vehículo que se construyó en [27]. Como resultados de dichos trabajos la estructura del quadrotor se encuentra bien definida, debiéndose avanzar con la obtención de una plataforma generalista que permita realizar experimentos de forma sencilla y sistemática.

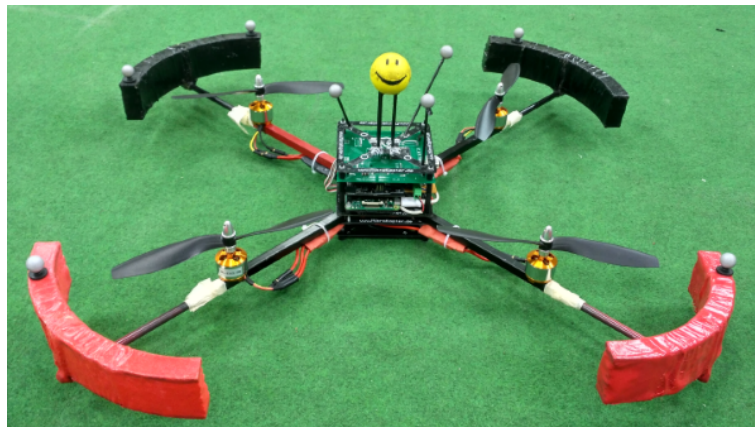


Figura 3.2: Quadrotor utilizado

En este apartado se definirán los componentes de físicos y de software que permitirán el uso de cualquier vehículo multirrotor en la plataforma diseñada. No obstante, gracias a la modularidad del programa, este sería fácilmente modificable para permitir su implementación en cualquier tipología de robot móvil.

3.2.1 Hardware

Los elementos físicos de un sistema de control en bucle cerrado se pueden clasificar en tres grupos: controladores, sensores y actuadores. El quadrotor se alimenta con 12V, el estándar de las baterías usadas con este tipo de vehículos. Sin embargo, por comodidad con los experimentos realizados, se ha usado una fuente de tensión para generar la tensión necesaria. Además, se tiene un convertidor de tensión de entre 12-5V para alimentar la Raspberry y esta a su vez alimenta la IMU. Para una descripción más detallada de los componentes puede consultarse [25].

3.2.1.1 Controladores

El microcomputador que gobierna el quadrotor es una Raspberry Pi 2. El sistema operativo que dispone es de tiempo real, por tanto puede asegurar que las diferentes tareas que se ejecuten en ella serán ejecutadas en un tiempo determinado. A señalar que aunque la Raspberry Pi 3 este en el mercado, a fecha de realización del trabajo no dispone de un sistema operativo en tiempo real contrastado como en el caso de la Raspberry Pi 1/2 y se han tenido problemas con la generación de la red Wifi de la Raspberry, usada para conectarse con la estación de tierra. No obstante, las ultimas pruebas realizadas con la plataforma realizada han dado buenos resultados con el ultimo modelo de Raspberry Pi.

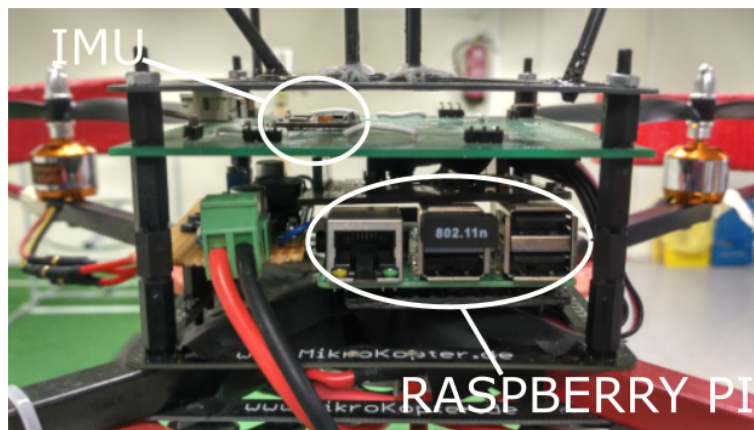


Figura 3.3: Detalle de las placas del quadrotor

3.2.1.2 Sensores

La unidad de medición inercial (IMU) es el principal sensor del quadrotor. Esta formada por los acelerómetros, giroscopios y el magnetómetro. La comunicación se realiza mediante el protocolo i2c a una velocidad de 450k baudios. A continuación se describen sus características:

- Acelerómetro. Permite obtener la aceleración en los tres ejes, obteniendo mediante proyección de la aceleración de la gravedad los ángulos formados con la tierra, por tanto la medida obtenida es absoluta. No permite obtener el yaw, ya que la aceleración de la gravedad es independiente al angulo. Debido a multitud de fuerzas presentes en el vehículo, se trata de un sensor con poca precisión.

- **Giroscopio.** Permite obtener las velocidades angulares de los tres ejes. La precisión de estos dispositivos es muy buena, no obstante es necesaria su integración para obtener los ángulos. Debido a la integración, su medida presenta deriva con el tiempo.
- **Magnetómetro.** Permite obtener la dirección del campo magnético. En ausencia de otros campos, permite obtener la dirección del polo norte y marcar una referencia para el cálculo del yaw.

Mediante el uso de un filtro de Kalman extendido [32], [33], se realiza la fusión de las medidas del acelerómetro y del giroscopio para el caso del roll y pitch y del magnetómetro y giroscopio para el yaw. Las varianzas para el ajuste del filtro se han obtenido con las varianzas experimentales fijando el quadrotor.

Además, el sistema Optitrack (ver sección 3.4) puede ser considerado como otro sensor, donde se obtienen directamente las medidas angulares y de posición del vehículo.

3.2.1.3 Actuadores

Los cuatro motores sin escobillas (ESC, brushless) forman los actuadores presentes del sistema. Estos permiten grandes velocidades de giro, que junto las hélices generan las diferencias de momento para desplazar el quadrotor por el espacio. Sin embargo estos motores necesitan un driver o variador para poder mandarle las referencias de velocidad y que generen las tensiones necesarias para el motor. Estos drivers funcionan mediante el protocolo PWM/PPM propio de los servos, donde una señal PWM de un frecuencia inferior a 450Hz genera un pulso de entre 1ms para mínima velocidad y de 2ms para máxima.

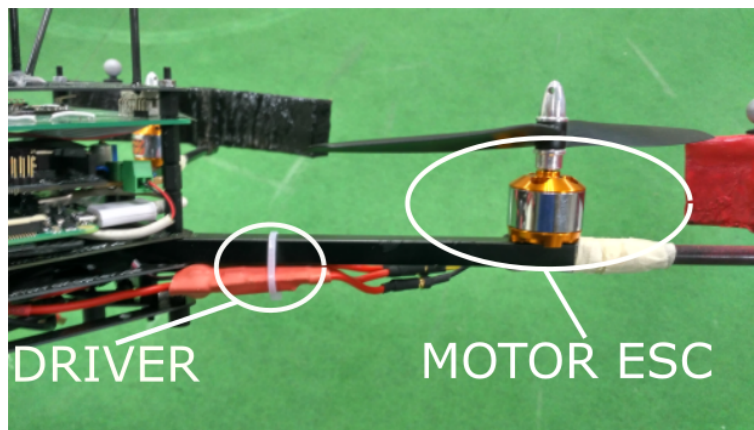


Figura 3.4: Detalle de los motores y los drivers

3.2.2 *Software*

Uno de los principales problemas que había con el programa era el elevado número de personas que habían intervenido en su desarrollo a lo largo de los diferentes trabajos (ver sección 2.5). Como consecuencia, el software resultante era de difícil entendimiento y modificación. Por tanto, el software del quadrotor ha sido totalmente remodelado respecto al usado hasta entonces para subsanar los fallos mencionados.

El programa ha sido desarrollado el paradigma de la programación orientada a objetos (POO) con el lenguaje de programación C++. Dicho lenguaje, al ser de bajo nivel, permite implementar las características de tiempo real necesarias para poder realizar controladores que se ejecuten dentro de los tiempos establecidos.

Todos los elementos principales forman una clase, creando instancias todos los objetos en el archivo principal y pasando la referencia los objetos que sean necesarios para el funcionamiento del mismo. Por ejemplo, al objeto IMU se le pasaría la referencia del puerto i2c. Esto permite cambiar todas las relaciones entre los objetos en un único archivo, siendo la estructura global fácilmente reconocible.

La comunicación con el PC se ha realizado mediante una red wifi punto a punto generada por el quadrotor y por el protocolo UDP. Los datos a enviar se cifran con el protocolo Mavlink [34], el cual permite enviar tramas de datos con cierta estructura y es compatible con la mayoría de quadrotors del mercado. Existen dos puertos dedicados a esta comunicación, uno para enviar ordenes a la Raspberry y otro para recibir medidas de la misma. Por otra parte, paralelamente se utiliza otro puerto sin Mavlink para el envío en exclusiva de los datos del Optitrack, ya que se envían las tramas a 100Hz de frecuencia.

La señal PWM encargada de gobernar los drivers de los motores se ha generado por software al no disponer de puertos PWM reales, con el uso de la librería Pigiopio [35]. Por tanto, aunque permite controlar el quadrotor, la precisión de la señal se podría mejorar usando una placa extensora de puertos PWM. Por contra, el programa permite generar un gran número de puertos PWM y se ha implementado la opción de elegir el número de motores deseado para poder usarse con otro tipo de multirrotor. El rango de actuación de los motores se ha fijado de 0 a 1.

Las tareas necesarias para el funcionamiento, se han creado en hilos periódicos con una prioridad establecida para el Schedule de tiempo real. Las tareas críticas para mantener estable el quadrotor son las de control y sensorización, por tanto poseen la máxima prioridad. Las diferentes tareas cíclicas realizadas son: controladores de los puertos, el generador de las señales PWM para los motores, la lectura de la IMU, la lectura de las medidas del Optitrack, la obtención del estado del quadrotor, la comunicación con el HMI y el control del quadrotor.

En lo referente al control, para realizar pruebas con diferentes controladores solo se necesita modificar la clase correspondiente. Dentro de esta, se disponen de las medidas necesarias y los comandos específicos para mandar las referencias a los motores. Además, mediante el HMI se puede elegir si se desea realizar un control de posición o simplemente de orientación.

3.3 La rótula

La rótula es el nombre del banco de pruebas utilizado que emplea una rótula real como elemento móvil, permitiendo el giro del quadrotor en los tres grados de libertad (roll, pitch y yaw). Es necesario destacar que en el trabajo normalmente se utilizará el nombre de rótula para describir el dispositivo de pruebas y no el elemento rótula.

Desde [25], se han construido diferentes bancos de pruebas. En [26]-[28], se encuentra la estructura básica de la rótula empleada. Sin embargo, debido a problemas de fricción en los componentes, el componente rótula ha debido ser sustituido por otro de menor fricción.

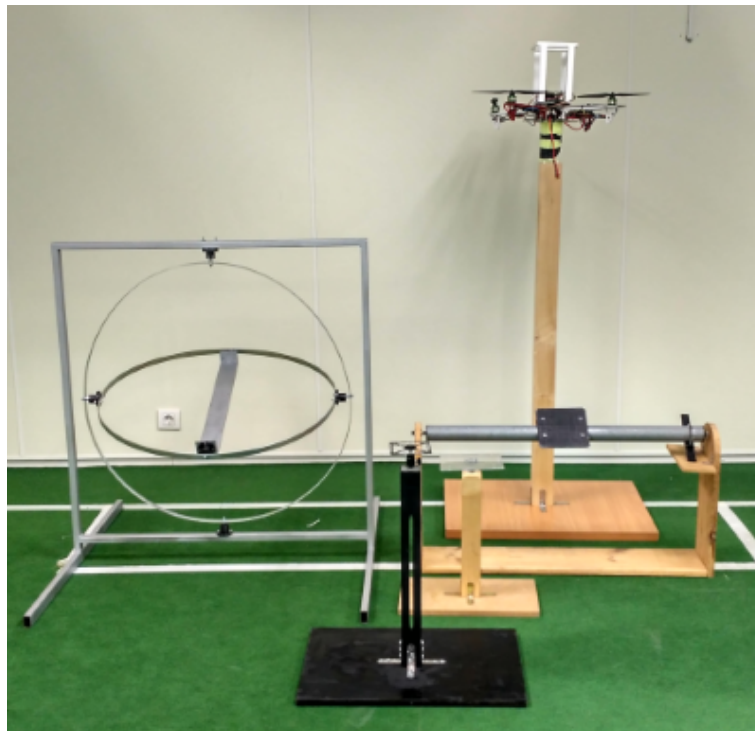


Figura 3.5: Rótulas (bancos de pruebas) del laboratorio

En la figura 3.5 se pueden ver las diferentes rotulas del laboratorio. La rótula empleada es la del centro, pero gracias a la plataforma desarrollada cualquier rótula se puede usar considerando sus características físicas (ver sección 4.4), incluso la plataforma de un grado de libertad o la giroscópica.

El principal uso de la rótula en validación segura de algoritmos de control. Por tanto se pueden probar diferentes estrategias de control sin riesgo al solo permitirse el giro sus ángulos. En el quadrotor utilizado se ha dispuesto un agarre en forma de bola que permite mantener el quadrotor recto, cogerlo en caso de volverse inestable el control y aplicar fácilmente perturbaciones. Una vez obtenido un control estable, considerando la sección 4.8 se puede probar en vuelo libre.

3.4 Optitrack

Optitrack [36] es una de las principales compañías de captura de movimiento del mundo. El equipo de captura de movimiento consiste en una serie de cámaras, el procesador independiente de imágenes y su software específico para PC. Todo ello permite obtener medidas con gran precisión (tolerancias menores a un milímetro) y con latencias de décimas de milisegundos. Entre los usos de este tipo de instalaciones se encuentran la realización de películas, videojuegos, metrología industrial, estudio de distancias límites y robótica.

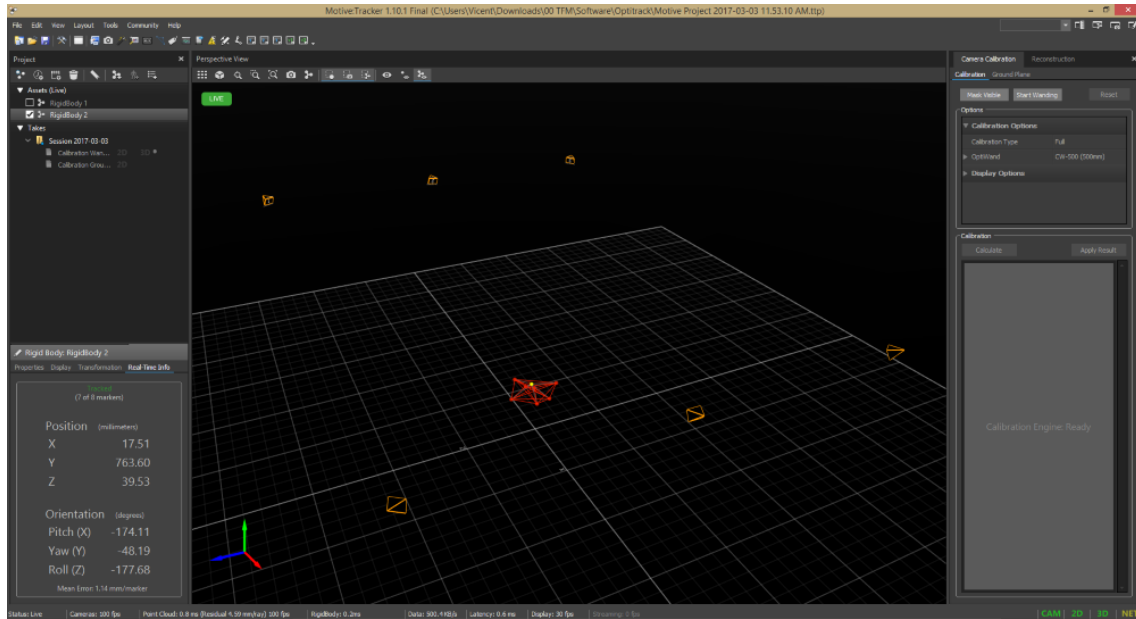


Figura 3.6: Programa Motive de Optitrack

La captura de movimientos se realiza mediante la reflexión de luz infrarroja. Los objetos a detectar se les fija un número suficiente de marcadores que permitan su localización unívoca en el espacio. Estos marcadores son detectados por las cámaras infrarrojas y se calcula la posición más probable del marcador. En la figura 3.2 se pueden observar los marcadores dispuestos en el quadrotor. Con el correcto calibrado del programa puede obtener la posición y orientación del objeto formado por dichos marcadores. El programa da las medidas de posición y la orientación del objeto en cuaterniones.

El programa de procesamiento de la captura permite el envío de datos online mediante el protocolo propio NatNet. El envío de datos se realiza hasta una frecuencia de 100 Hz y puede enviar información sobre diferentes cuerpos capturados. Esto permitirá en un futuro realizar trabajos cooperativos entre múltiples quadrotors. Sin embargo esta trama no puede enviarse directamente al vehículo debiendo pasar por el HMI.

El uso del Optitrack ha sido posible gracias a que recientemente el equipo de investigación a conseguido una licencia y se ha rehabilitado una instalación antigua de seis cámaras. Las cámaras se han orientado de forma que la zona central de la habitación se obtenga la máxima precisión, siendo captada por las seis cámaras. Para obtener una mejor identificación se han colocado ocho marcadores en el quadrotor. Los marcadores deben situarse lo más alejados del centro de giro

para minimizar el efecto del error al calculo de la orientación. Por ello se han situado cinco en los extremos de las protecciones y tres en el medio.



Figura 3.7: Instalación del sistema Optitrack



Figura 3.8: Detalle de una cámara de la instalación Optitrack

3.5 La interfaz hombre máquina

La interfaz hombre máquina o HMI (human machine interface) es el programa ejecutado en el PC base que permite mandarle las diferentes instrucciones al quadrotor. Este elemento ha sido realizado desde cero en comparación con los HMI usados anteriormente [30], [31].

En primer lugar este HMI ha sido programado en Java mediante el IDE open source NetBeans en lugar de con C++ en QT. Por tanto el programa obtenido es perfectamente utilizable sin modificaciones por cualquier sistema operativo que disponga de la maquina virtual de Java, a diferencia de los antiguos que se ejecutaban en Ubuntu. A continuación se muestran las características principales del programa, las cuales amplían considerablemente las que habían en los anteriores HMI.

- Comunicación UDP, mediante el protocolo Mavlink (ver subsección 3.2.2) con la Raspberry Pi.
- Tres modos disponibles: Desarmado, motores en manual y control automático con/sin control de posición.
- Envío de hasta 15 parámetros por controlador.
- Visualización de las principales variables del quadrotor.
- Recepción de las tramas NatNet del Optitrack y envío de las mismas a la Raspberry.
- Envío de las seis referencias más el thrust inicial.
- Modo patrón. Mediante un pequeño “lenguaje” de programación interpretado creado para la ocasión, se puede crear un set de instrucciones para el envío patrones establecidos. Por ejemplo cambiar la referencia del yaw, esperar 3000 milisegundos y volverla ha cambiar (ver sección 7.2).
- Configuración fácilmente modificable mediante la modificación de un archivo de texto, entre ellas el numero de parámetros a enviar o las IP.
- Posibilidad de cambio de las referencias angulares mediante un Joystick.
- Permitir la carga/guardado de los parámetros de los controladores.
- Creación de una copia de seguridad de todos los valores previos a cerrar el programa.

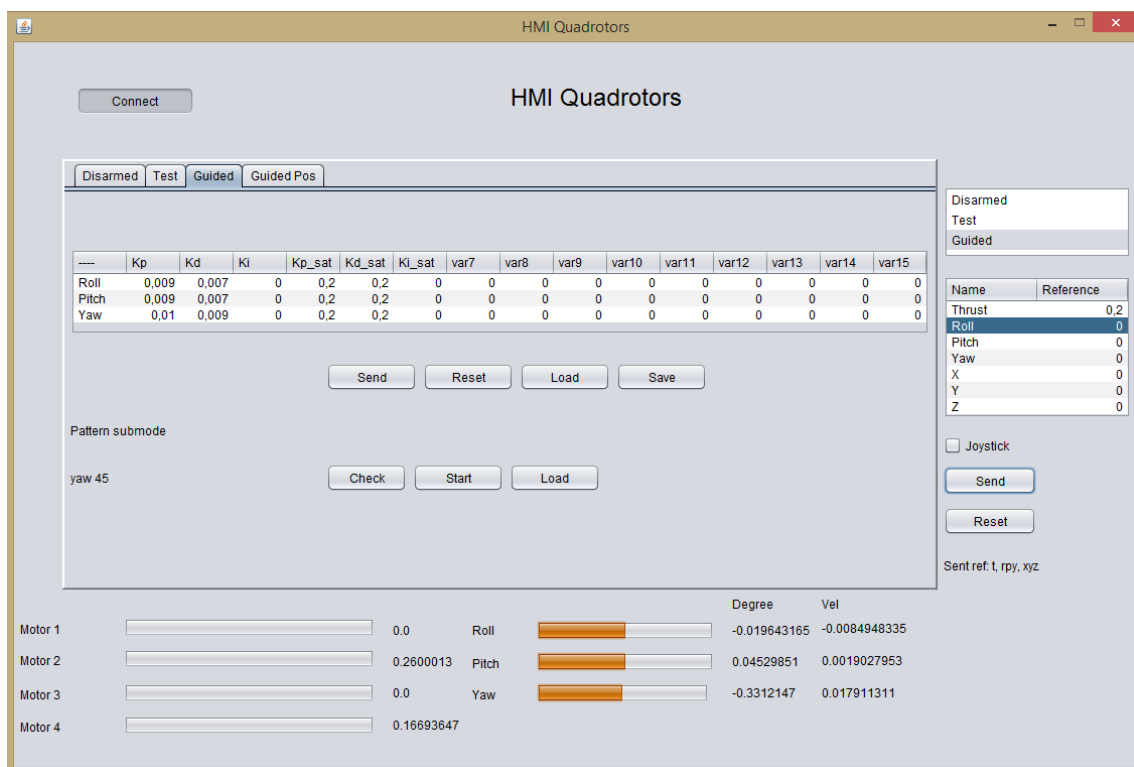


Figura 3.9: HMI desarrollado para la plataforma

Modelado del sistema

4.1 Introducción

El modelado consiste en la obtención de las ecuaciones que caracterizan cierto sistema a partir de los principios científicos que dominan el proceso [37]. El modelo obtenido junto los parámetros estimados del mismo supondrá una primera aproximación al comportamiento real del sistema, permitiendo el diseño de controladores así como la construcción de un simulador.

Es importante señalar que en la práctica no se pueden modelar las interacciones de todos los componentes de un sistema, debiéndose tomar una serie de simplificaciones, por lo que se debe tener en cuenta los posibles errores de modelado debidos a estas simplificaciones [38]. No obstante, aunque el modelo obtenido no sea totalmente preciso permitirá establecer relaciones de causalidad, que en el caso de identificación experimental de sistemas MIMO permitirá reducir las combinaciones de entradas y salidas resultando en un número menor de funciones a identificar.

4.2 Modelo general del quadrotor

El modelo general representa el comportamiento del quadrotor en vuelo libre. En la literatura se encuentran multitud de representaciones del mismo, con mayor o menor número de simplificaciones [19]. El modelo considerado parte de las siguientes premisas:

- El quadrotor es un cuerpo rígido.
- Estructura simétrica.
- La matriz de inercia es invariante.
- Velocidades lineales/angulares pequeñas, para anular términos cruzados/poco significantes.
- Ángulos pequeños.
- Motores iguales con dinámica despreciable respecto la del quadrotor.

4.3 Desacoplamiento

A partir del modelo simplificado mostrado, se puede realizar un cambio de variable para conseguir desacoplar el sistema. Por tanto sera posible considerar el sistema MIMO como un conjunto de sistemas SISO independientes. Si se considera la descomposición (Ecuación 4.3) de la fuerza en un termino inicial igual para todos y un incremento, a partir de la Ecuación 4.1 se pueden expresar las fuerzas como en la Ecuación 4.4.

$$F_x = kl_1(u_0 + \Delta_x) \quad (4.3)$$

$$\begin{cases} F_4 - F_2 = kl_1(\Delta_4 - \Delta_2) \\ F_3 - F_1 = kl_1(\Delta_3 - \Delta_1) \\ F_2 + F_4 - F_1 - F_3 = kl_1(\Delta_2 + \Delta_4 - \Delta_1 - \Delta_3) \end{cases} \quad (4.4)$$

Mediante el cambio de variables de la Ecuación 4.5 y considerando J_x es la inercia del eje de giro de x , se obtiene finalmente en Ecuación 4.6 la expresión desacoplada del control de orientación (angular) . Como puede observarse, es un sistema lineal correspondiente a un doble integrador.

$$\begin{cases} 2u_\phi = \Delta_4 - \Delta_2 \\ 2u_\theta = \Delta_3 - \Delta_1 \\ 4u_\psi = \Delta_2 + \Delta_4 - \Delta_1 - \Delta_3 \end{cases} \quad (4.5)$$

$$\begin{cases} J_\phi \ddot{\phi} = 2kl_1 u_\phi \\ J_\theta \ddot{\theta} = 2kl_1 u_\theta \\ J_\psi \ddot{\psi} = 4kl_1 u_\psi \end{cases} \quad (4.6)$$

4.3.1 Control en “cruz” y “equis”

Existen dos formas de realizar el control de un quadrotor según el sistema de referencia dado, donde coincide con los ejes del quadrotor (control en “cruz”) o se encuentran rotados 45° respecto al eje Z. En la figura 4.2 se muestran ambas configuraciones.

En el apartado anterior se ha explicado el desacoplamiento para la configuración en “cruz”. Para el caso de la configuración en “equis”, se puede obtener una formulación similar considerando que al estar los ejes rotados todos los motores afectan a los ángulos de roll y pitch. Cualitativamente, se puede obtener el desacoplamiento del sistema considerando que para el giro de roll (eje x) los motores 1 y 2 deberán aumentar su fuerza (incrementar su Δ), mientras que las 3 y 4 disminuirla (decrementar su Δ). En el ejemplo, en la configuración en “cruz” deberá aumentar el motor 1 y disminuir el 3. En el caso del yaw, al ser el giro alrededor de su eje de giro, no se ve afectado por la configuración tomada.

El control en “equis” permite una mejor respuesta en frecuencias y robustez del sistema frente

asimetrías del vehículo [24]. No obstante, al aumentar la ganancia del sistema a altas frecuencias, el sistema aumenta su sensibilidad respecto ruido de medida. Por tanto, debido a la baja precisión de la señal PWM generada por la Raspberry Pi se ha decidido controlar el quadrotor mediante una configuración en “cruz”.

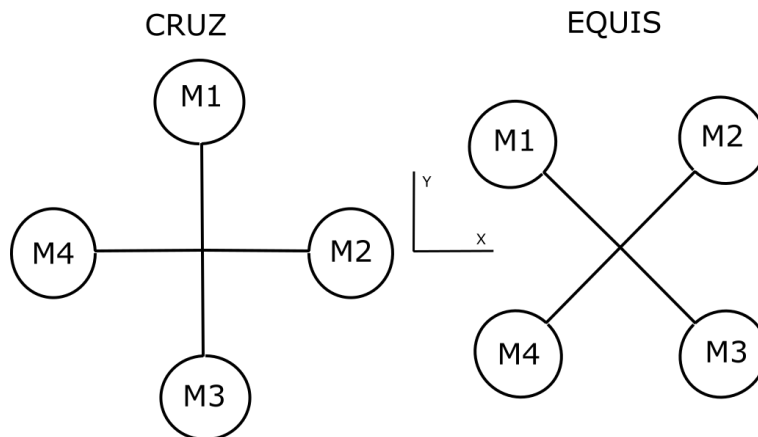


Figura 4.2: Configuraciones en “cruz” y “equis”

4.4 Modelo del quadrotor en la rótula

Una vez analizado el modelo general del quadrotor, se va a obtener el modelo para el caso de la rótula. En diferentes investigaciones donde se han usado rotulas [20], [29] se ha considerado que el modelo no difiere con el vuelo libre, siendo el mismo pero con un aumento de la inercia [29].

Para modelar la rótula se parten de las mismas consideraciones que las simplificación de la sección 4.2. En primer lugar se identificará el roll/pitch y en segundo el yaw. El sistema de referencia correspondiente al roll y pitch es el mostrado en la figura 4.3.

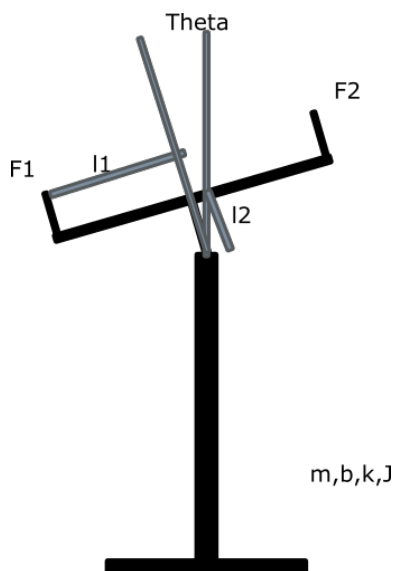


Figura 4.3: Sistema de referencia de la rotula

Considerando las simplificaciones tomadas y nomenclatura en la sección 4.2 y sección 4.3, m es la masa del vehículo, b es el rozamiento de la rótula y l_2 la distancia entre el centro de la rótula y el del quadrotor. A partir de la mecánica clásica de Newton, se obtiene la ecuación diferencial (Ecuación 4.7) que relaciona el ángulo con la acción de control. Cabe destacar que se trata de una ecuación no lineal debido al termino que acompaña la dinámica introducida por la rotula ($l_2mg \sin \theta$).

$$J_\theta \ddot{\theta} = 2kl_1 u_\theta + l_2mg \sin \theta - b\dot{\theta} \quad (4.7)$$

En el caso del yaw, se puede obtener de la expresión anterior considerando que l_2 es nula o que la componente peso es perpendicular al eje y que los cuatro motores actúan sobre ella. Por tanto, considerando la Ecuación 4.7 y en el caso del roll y pitch el rozamiento de la rotula es despreciable (pequeños ángulos, no para el yaw), se obtiene el modelo de la rótula para los tres ángulos de Euler (Ecuación 4.8).

$$\begin{cases} J_\phi \ddot{\phi} = 2kl_1 u_\phi + l_2mg \sin \phi \\ J_\theta \ddot{\theta} = 2kl_1 u_\theta + l_2mg \sin \theta \\ J_\psi \ddot{\psi} = 4kl_1 u_\psi - b\dot{\psi} \end{cases} \quad (4.8)$$

Como puede observarse, si l_2 y b son nulos se obtiene el doble integrador, que como se ha visto en la sección 4.2 corresponde al quadrotor en vuelo libre. Por tanto, el quadrotor en vuelo libre podría considerarse un caso particular del quadrotor en la rótula.

4.5 Obtención de los parámetros físicos

A partir de los modelos estudiados en la sección 4.2 y en la sección 4.4 se deben obtener los parámetros del mismo para poder determinar el comportamiento del sistema. Tal como se remarco en los apartados anteriores, el modelo obtenido es aproximado, por tanto sera suficiente con obtener una medida aproximada de los mismos. Con ello, la mayoría de parámetros son directamente medibles, a excepción de las constantes de los motores y las inercias.

Para obtener las inercias se debe tener en consideración que la masa del quadrotor no se encuentra uniformemente distribuida, siendo los motores los componentes de mayor peso. Por tanto, se considera el quadrotor como un cubo de $\frac{2}{3}l_1$ de lado, siendo simétrico en los tres ejes de interés. Considerando la distancia l_2 se obtienen la inercia respecto su centro de masas (vuelo libre) y la inercia respecto el giro de la rótula (caso rótula).

Las constantes de los motores se ha obtenido como media de diferentes motores disponibles en el mercado. En la tabla 4.1 se muestran los parámetros estimados.

Tabla 4.1: Parámetros estimados

Parámetro	Medida
Masa	800gr
l_1	0.25m
l_2	0.125m
J_{vuelo}	$0.015kgm^2$
J_{rotula}	$0.0275kgm^2$
$k_{motores}$	1N

4.6 Simulador

Con los parámetros y las ecuaciones que describen el comportamiento de la plataforma, se ha programado un simulador en Simulink para poder analizar el comportamiento del sistema en diferentes situación, previas a su análisis en la realidad. Por tanto se emplearán las ecuaciones de la sección 4.4 y los parámetros de la sección 4.5 en una primera fase y en una segunda los corregidos en la etapa de identificación experimental.

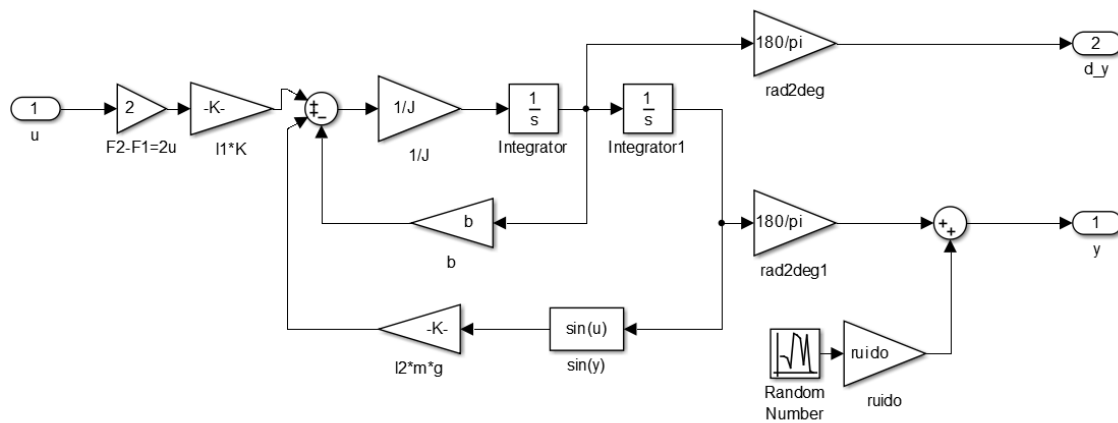


Figura 4.4: Simulador desarrollado en Simulink. Quadrotor

Debido a las ecuaciones obtenidas, el simulador básico permite simular los tres ángulos mediante $b = 0$ para el caso del roll y pitch y $l_2 = 0$ para el yaw. En la figura 4.4 se muestra el detalle del simulador y en la figura 4.5 el entorno de control.

Cabe destacar que las mediadas que se obtienen son el angulo y su derivada, tal y como de obtienen en el quadrotor (ver subsección 3.2.1.2). Como puede observarse, se ha incluido la posibilidad de usar medidas del angulo con ruido (en la velocidad no debido a la precisión del giroscopio, ver subsección 3.2.1.2), para simular el comportamiento real. El valor de la varianza se obtiene mediante la medida de los ángulos con el quadrotor bloqueado. Destacar también la conversión de radianes a grados, ya que todos los diseños se realizarán en grados.

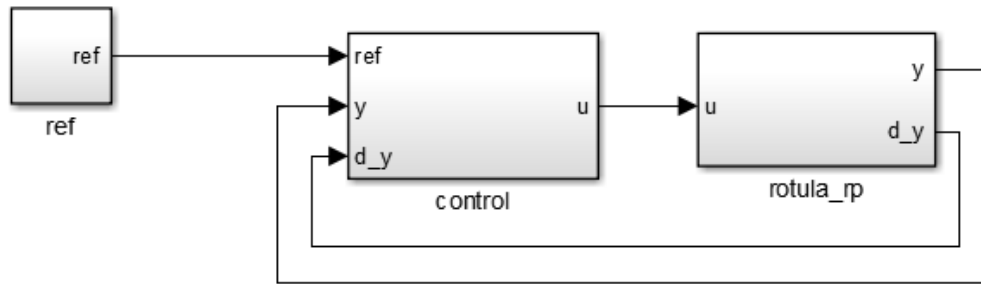


Figura 4.5: Simulador desarrollado en Simulink. Entorno de control

4.7 Linealización y representaciones

La linealización consiste en obtener un sistema lineal equivalente a uno no lineal alrededor de cierto punto de funcionamiento [38]. Esto es necesario debido a que la mayoría de técnicas de control como de identificación consideran que la planta es lineal. Para ello se va a considerar que $\sin(\theta) = \theta$ (Ecuación 4.9).

$$\begin{cases} J_\phi \ddot{\phi} = 2kl_1 u_\phi + l_2 mg \phi \\ J_\theta \ddot{\theta} = 2kl_1 u_\theta + l_2 mg \theta \\ J_\psi \ddot{\psi} = 4kl_1 u_\psi - b \dot{\psi} \end{cases} \quad (4.9)$$

En la Ecuación 4.10 se representa mediante la transformada de Laplace el modelo general unidimensional (con b y l_2) como función de transferencia.

$$G_p(s) = \frac{2kl_1}{Js^2 + bs - l_2mg} \quad (4.10)$$

A partir de la Ecuación 4.10, se observa que para el caso del yaw el sistema presentará un integrador con un polo estable ($s = 0$ y $s = -\frac{b}{J}$). Por tanto el yaw se comportará como el sistema en vuelo libre pero con una dinámica estable adicional. Por otra parte, en el caso del roll y pitch los polos resultantes son reales simétricos respecto el eje complejo, por tanto tiene un polo inestable ($s = \pm\sqrt{\frac{l_2mg}{J}}$). Debido a ello se analizará con más profundidad el modelo del roll y pitch.

En la Ecuación 4.11 se encuentra la representación del modelo del roll y pitch en espacio de estados, la cual será de enorme importancia en el calculo de controladores (ver sección 6.5). El primer estado es el ángulo, mientras que el segundo es su derivada. También cabe considerar las representaciones del sistema en tiempo discreto ($T = 4ms$). En la Ecuación 4.12 se representa mediante la transformada Z (suponiendo la entrada al sistema un retenedor de orden cero) y en la Ecuación 4.14 representación discreta en espacio de estados (aproximando la derivada por el método de Euler).

$$\begin{cases} \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ \frac{l_2 mg}{J} & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{2kl_1}{J} \end{bmatrix} u \\ y = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \end{cases} \quad (4.11)$$

$$G_p(z) = \frac{kl_1(e^{-aT} + e^{aT} - 2)(z + 1)}{l_2 mg(z - e^{-aT})(z - e^{aT})} \quad (4.12)$$

$$a = +\sqrt{\frac{l_2 mg}{J}} \quad (4.13)$$

$$\begin{cases} \begin{bmatrix} x_{1,k+1} \\ x_{2,k+1} \end{bmatrix} = \begin{bmatrix} 1 & T \\ \frac{Tl_2 mg}{J} & 1 \end{bmatrix} \begin{bmatrix} x_{1,k} \\ x_{2,k} \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{2Tkl_1}{J} \end{bmatrix} u \\ y = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \end{cases} \quad (4.14)$$

4.8 Comparativa rótula/vuelo libre

Como se ha visto en los apartados anteriores, el comportamiento de la rótula se diferencia del vuelo libre principalmente a una mayor inercia y un polo inestable. Por ello, hay que considerar si es posible estudiar el comportamiento del quadrotor en la rótula y poder obtener conclusiones en base al comportamiento esperado en vuelo libre. Por ello, se va a analizar la posición de los polos en bucle cerrado en ambos casos con un control equivalente.

La demostración va a realizarse para una realimentación del estado (ver sección 6.5), donde la Ecuación 4.15 es la característica del sistema. Todos los parámetros se consideraran positivos. Para el sistema en general, se obtiene la expresión de la Ecuación 4.16.

$$\det(sI - (A - BK)) \quad (4.15)$$

$$s^2 + s\frac{2kl_1}{J}K_2 + \frac{2kl_1}{J}K_1 - \frac{l_2 mg}{J} \quad (4.16)$$

Considerando los sistemas rótula y vuelo libre por separado, los polos del sistema se obtienen a partir de la formula para las ecuaciones de segundo grado. Los coeficientes para ambos casos se pueden ver en la Ecuación 4.17.

$$\left\{ \begin{array}{l} s = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} \\ a = 1 \\ b_{vuelo} = \frac{2kl_1}{J_{vuelo}} K_{2,vuelo} \\ b_{rotula} = \frac{2kl_1}{J_{rotula}} K_{2,rotula} \\ c_{vuelo} = \frac{2kl_1}{J_{vuelo}} K_{1,vuelo} \\ c_{rotula} = \frac{2kl_1}{J_{rotula}} K_{1,rotula} - \frac{l_2 mg}{J_{rotula}} \end{array} \right. \quad (4.17)$$

Si en el caso de la rotula se tiene un controlador que estabilice el sistema, se puede asegurar que si se cumple $b_{vuelo} = b_{rotula}$ y $c_{vuelo} \geq c_{rotula}$ el regulador en el caso de vuelo libre también será estable, ya que los polos se encontrarán más cerca del eje de simetría de los polos ($-\frac{b}{2a}$). Para ello es suficiente que la relación entre la realimentación del estado sea la misma que las inercias en vuelo libre y en la rótula (Ecuación 4.18).

$$\left\{ \begin{array}{l} b_{vuelo} = b_{rotula} \\ \frac{2kl_1}{J_{vuelo}} K_{2,vuelo} = \frac{2kl_1}{J_{rotula}} K_{2,rotula} \\ \frac{K_{2,vuelo}}{K_{2,rotula}} = \frac{J_{vuelo}}{J_{rotula}} \\ c_{vuelo} \geq c_{rotula} \\ \frac{2kl_1}{J_{vuelo}} K_{1,vuelo} \geq \frac{2kl_1}{J_{rotula}} K_{1,rotula} - \frac{l_2 mg}{J_{rotula}} \\ Si : \frac{2kl_1}{J_{vuelo}} K_{1,vuelo} = \frac{2kl_1}{J_{rotula}} K_{1,rotula} \\ \frac{K_{1,vuelo}}{K_{1,rotula}} = \frac{J_{vuelo}}{J_{rotula}} \\ 0 \geq -\frac{l_2 mg}{J_{rotula}} \end{array} \right. \quad (4.18)$$

Identificación

5.1 Introducción

La identificación de sistemas consiste en la obtención de modelos que relacionen una variables (entradas) con otras (salidas) [39]. Aunque como se ha visto en el Capítulo 4 es posible obtener un modelo a partir de los principios físicos dominantes, debido a las numerosas simplificaciones realizadas es posible que el comportamiento real diste enormemente del teórico, siendo necesaria una etapa de identificación que permita obtener una mejor aproximación a la realidad.

Respecto al diseño de controladores, la respuesta del sistema a frecuencias medias marca la dinámica del controlador [37]. Esto es debido a que la mayoría de sistemas físicos son paso-bajos (baja ganancia a altas frecuencias), necesitándose una acción de control elevada lo cual podría hacer saturar al sistema así como excitar no-linealidades. Debido a la baja ganancia de los sistemas a altas frecuencias, la relación señal ruido empeora, dificultando la correcta identificación incluso sin considerar los problemas anteriores. Por contra, a bajas frecuencias, la acción integral o la perturbación estimada correspondiente permite corregir dicho error.

Por otra parte, las técnicas clásicas de identificación se fundamentan en excitar la entrada con una onda dada y observar la respuesta de la salida [39]. Por ejemplo se encuentran la identificación mediante una entrada escalón y la aproximación de la respuesta a una función de un orden dado o la excitación mediante entradas sinusoidales y la obtención de la ganancia a dicha frecuencia.

No obstante, las técnicas mencionadas se centran en la respuesta del sistema en un ancho de bando muy limitado. Por ello, si se desea realizar una mejor identificación se debe excitar al sistema mediante una señal con componentes de todas las frecuencias, como es un ruido normal [39]. En este caso, se suele identificar la función de transferencia discreta (transformada Z, ecuación en diferencias) empleando mínimos cuadrados. Por ejemplo, si se tiene el prototipo de modelo de la Ecuación 5.1 y un conjunto de n datos temporales, para identificar el proceso se debe resolverse el problema de mínimos cuadrados de la Ecuación 5.2.

$$G_{ej}(z) = \frac{b_1 z^{-1}}{1 + a_1 z^{-1}} \quad (5.1)$$

$$\begin{bmatrix} y_2 \\ y_3 \\ y_4 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} -y_1 & u_1 \\ -y_2 & u_2 \\ -y_3 & u_3 \\ \vdots & \vdots \\ -y_{n-1} & u_{n-1} \end{bmatrix} \begin{bmatrix} a_1 \\ b_1 \end{bmatrix} \quad (5.2)$$

5.2 Situaciones posibles

A continuación se describen las cuatro situaciones modelo más comunes [39], dependiendo de la naturaleza de las perturbaciones presentes. La estructura seleccionada limita las técnicas de identificación disponibles, existiendo diferentes métodos para cada caso particular. Cabe destacar que no existe ningún método universal, por tanto la elección del mismo viene condicionada a la estructura supuesta. En el caso de no ser la situación real, se producirá un error sistemático en los parámetros identificados.

- Situación 1: $A(z^{-1})y(t) = q^{-d}B(z^{-1})u(t) + e(t)$

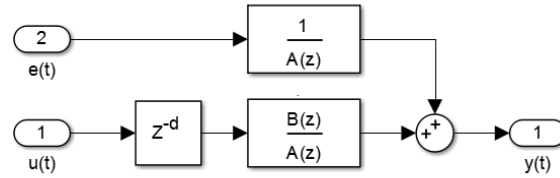


Figura 5.1: Situación 1

- Situación 2: $A(z^{-1})y(t) = q^{-d}B(z^{-1})u(t) + A(z^{-1})e(t)$

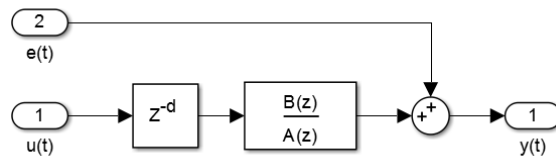


Figura 5.2: Situación 2

- Situación 3: $A(z^{-1})y(t) = q^{-d}B(z^{-1})u(t) + C(z^{-1})e(t)$

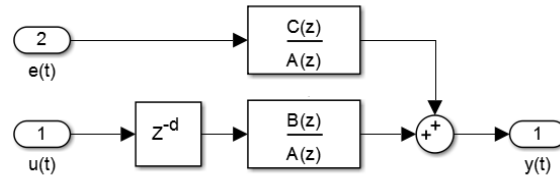


Figura 5.3: Situación 3

- Situación 4: $A(z^{-1})y(t) = q^{-d}B(z^{-1})u(t) + \frac{1}{C}(z^{-1})e(t)$

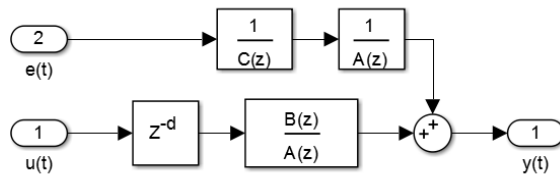


Figura 5.4: Situación 4

Según [39] la situación 3, que corresponde a un modelo ARMAX, se encuentra aproximadamente en 2/3 de las situaciones, la situación 2 en casi 1/3, siendo las situaciones 1 y 4 a casos pocos comunes. Por tanto, para el caso del quadrotor se asumirá que se trata de la situación 3.

5.3 Identificación en bucle cerrado

La identificación en bucle cerrado es necesaria para identificar sistemas inestables, debido a que no se puede excitar la entrada manteniendo la salida en zona segura. Por ello, es necesario disponer de un regulador que estabilice el sistema. Existen dos enfoques [40] de orientar la identificación en bucle cerrado.

- Identificación directa: Se identifica el bucle cerrado con los métodos de identificación en bucle abierto, considerando la entrada la referencia al bucle. La planta se obtiene despejando de la ecuación del bucle cerrado.
- Identificación indirecta: Se identifica directamente el proceso, mediante técnicas que corrigen la correlación entre el error de medida y la excitación resultante.

La identificación indirecta es la más simple de realizar, limitándose a la identificación del conjunto estable y despejando el sistema. Sin embargo, el controlador debe ser conocido y lineal para poder despejar el modelo además de poder obtenerse identificaciones sin sentido físico e incluso no causales. Aunque la identificación indirecta requiere de técnicas desarrolladas en exclusiva, en general se obtienen mejores resultados [40], [41]. Por tanto se utilizará la identificación indirecta para la obtención del modelo experimental del quadrotor.

Por otra parte, si el sistema es inestable, la correlación entre el ruido y la entrada pueden

manifestarse en ceros de fase no mínima en la identificación. Existen técnicas avanzadas que permiten evitar el problema [42], pero en el caso estudiado se solucionara el problema mediante la aproximación en frecuencia de la función de transferencia obtenida en el modelado y la identificada. El controlador utilizado tampoco debe ser muy agresivo, ya que sino parte de la dinámica del regulador se superpondría a la de la planta [37]. Por contra, un regulador poco agresivo no sería capaz de estabilizarla.

Cabe destacar que la inestabilidad del proceso no es el único uso de la identificación en bucle cerrado, siendo también empleado para identificar procesos ya operativos o para la obtención de controladores de altas prestaciones [39]. El primer caso ocurre cuando la parada del proceso para realizar la identificación no es factible por ejemplo debido a su alto coste. El segundo caso sirve para obtener una mejor identificación a altas frecuencias permitiendo diseñar controladores más especializados.

5.4 Mínimos cuadrados recursivos

A partir de los mínimos cuadrados tradicionales (ver sección 5.1), se puede obtener su expresión recursiva. En el caso que el sistema real tenga la estructura supuesta, se obtiene en cada iteración una predicción no desviada con menor incertidumbre. Este algoritmo es la base de la mayoría de técnicas recursivas de identificación incluidas las específicas de bucle cerrado [39].

Considerando el ejemplo de la Ecuación 5.1, sin pérdida de generalidad, en la Ecuación 5.3 se obtiene la representación de los mínimos cuadrados recursivos. Los parámetros $\hat{\theta}$ indican que son estimados. La matriz F es la matriz de covarianza de los parámetros estimados, la cual debe ser inicializada con un valor grande para que el algoritmo converja.

$$\left\{ \begin{array}{l} \hat{\theta}(t)^T = \begin{bmatrix} \hat{a}_1 & \hat{b}_1 \end{bmatrix} \\ \hat{\phi}(t)^T = \begin{bmatrix} -y(t) & u(t) \end{bmatrix} \\ \hat{\theta}(t+1) = \hat{\theta}(t) + F(t)\phi(t)\phi(t)^T \epsilon^0(t+1) \\ F(t+1) = F(t) - \frac{F(t)\phi(t)\phi(t)^T F(t)}{1+\phi(t)^T F(t)\phi(t)} \\ \epsilon^0(t+1) = y(t+1) - \hat{\theta}(t)^T \phi(t) \end{array} \right. \quad (5.3)$$

5.4.1 X-CLOE

En el caso de un modelo ARMAX, el “Extended Closed Loop Output Error” o X-CLOE permite resolver la identificación en bucle cerrado [39]. El sistema considerado es el mostrado en la figura 5.5, donde $R(z)$, $T(z)$ y $S(z)$ son la descomposición del regulador utilizado.

La expresión del método se encuentra en la Ecuación 5.4, considerando las ecuaciones de los mínimos cuadrados recursivos y el mismo ejemplo. Cabe destacar el factor $H(z^{-1})$, el cual permite considerar la correlación entre el error y la entrada de control.

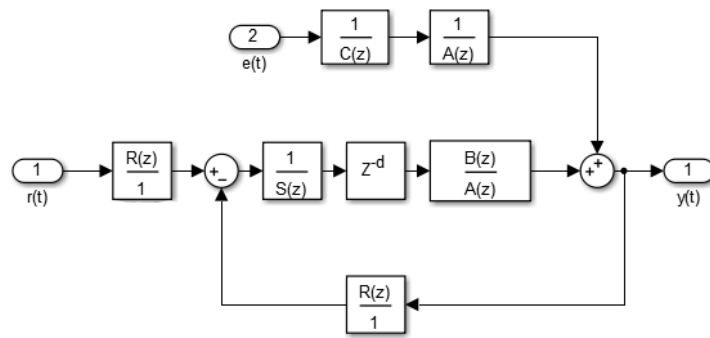


Figura 5.5: X-CLOE

$$\left\{ \begin{array}{l}
 \hat{y}^0(t+1) = -\hat{A}^*(t, z^{-1}) + \hat{B}^*(t, z^{-1})\hat{u}(t-d) + \hat{H}^*(t, z^{-1})\frac{\epsilon_{CL}(t)}{S(z^{-1})} = \hat{\theta}_e(t)^T \phi(t)_e \\
 \hat{u}(t) = -\frac{R(z^{-1})}{S(z^{-1})}\hat{y}(t) + \frac{T(z^{-1})}{S(z^{-1})}r(t) \\
 \hat{\theta}_e(t)^T = \begin{bmatrix} \hat{\theta}(t) & \hat{c}_1 \end{bmatrix} \\
 \phi_e(t)^T = \begin{bmatrix} \phi(t)^T & \epsilon_{CLf}(t) \end{bmatrix} \\
 \epsilon_{CLf} = \frac{\epsilon_{CL}}{S(z^{-1})} \\
 \epsilon_{CL}(t+1) = y(t+1) - \hat{\theta}_e(t+1)^T \phi_e(t) \\
 H(z^{-1}) = 1 + z^{-1}\hat{H}^*(z^{-1}) = 1 + C(z^{-1})S(z^{-1}) - P(z^{-1}) \\
 \hat{P}(z^{-1}) = \hat{A}^*(z^{-1})S(z^{-1}) + z^{-d}\hat{B}^*(z^{-1})R(z^{-1})
 \end{array} \right. \quad (5.4)$$

5.5 Diseño del experimento

Una vez seleccionado el método, se debe diseñar el experimento de forma correcta para obtener la máxima cantidad de información.

Se identificarán tres funciones de transferencia (ver sección 4.7) que relacionarán las variables de manipuladas auxiliares con los ángulos de roll, pitch y yaw. Por tanto, las simplificaciones realizadas en la sección 4.4 deben ser tomadas en cuenta. Por otro lado, se necesitan tomar datos dos conjuntos de datos para identificar y validar posteriormente dando a un total de seis conjuntos de datos. A continuación se muestran las principales consideraciones tomadas.

- Mantener los ángulos alrededor del punto de funcionamiento (origen).
- Modificar solo un ángulo para evitar los efectos de los términos cruzados.
- Entrada aleatoria.
- Dos conjuntos de datos o experimentos por modelo a identificar.

El controlador utilizado se ha calculado a partir de la primera aproximación obtenida en la sección 4.5 y con la metodología explicada en la subsección 6.5.2.

5.5.1 Pseudo Random Binary Sequence

Para conseguir una adecuada identificación frecuencial, se necesita introducir al sistema una señal con componentes en todas las frecuencias, es decir, un ruido blanco (ver sección 5.1). No obstante, un sistema discreto no puede representar una señal con componentes en todas las frecuencias. Por tanto una señal discreta puede representar componentes en frecuencia hasta la mitad de su frecuencia de muestreo (teorema de Shanon-Nyquist [43]).

La Pseudo Random Binary Sequence (PRBS) es una señal pseudoaleatoria que permite representar un ruido blanco discreto. Existen numerosas formas de generarlas, siendo la mostrada en la figura 5.6 una de las mas rápidas de implementar en un sistema empotrado.

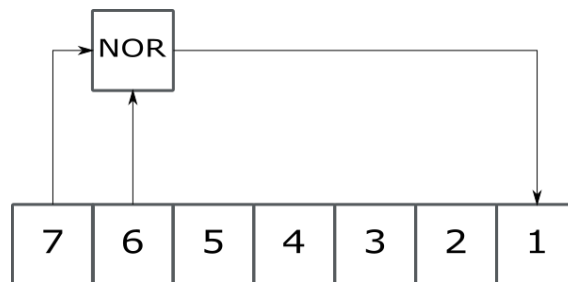


Figura 5.6: Pseudo Random Binary Sequence

El algoritmo consiste en una serie de bits, una operación lógica determinada, un par de posiciones de la serie y el operador de desplazamiento de bits. En el caso de la la figura 5.6 se tienen las posiciones 6 y 7, y la operación lógica NOR. Una vez realizada, se desplazan los bits y el resultado de la operación binaria se sitúa en la parte baja de los datos. Este ultimo bit es el encargado de generar el patrón pseudoaleatorio.

Según los bits empleados, las posiciones y las operaciones se obtiene el tamaño de la cadena pseudoaleatoria, el cual no siempre tiene solución optima. En [44] se encuentran una lista de las posibles combinaciones y sus propiedades.

En el caso del quadrotor la señal aleatoria (figura 5.6) tiene un tamaño de 127 periodos y va de $+3^\circ$ a -3° . El periodo de generación utilizado es un múltiplo del de muestreo para enfatizar la identificación a medias frecuencias. Este se ha calculado considerando que la PRBS llegue a identificar el régimen permanente, obteniendo el periodo base la división del tiempo de establecimiento y el numero de bits de la cadena. En la figura 5.7 se puede observar el patrón pseudoaleatorio empleado en la identificación.

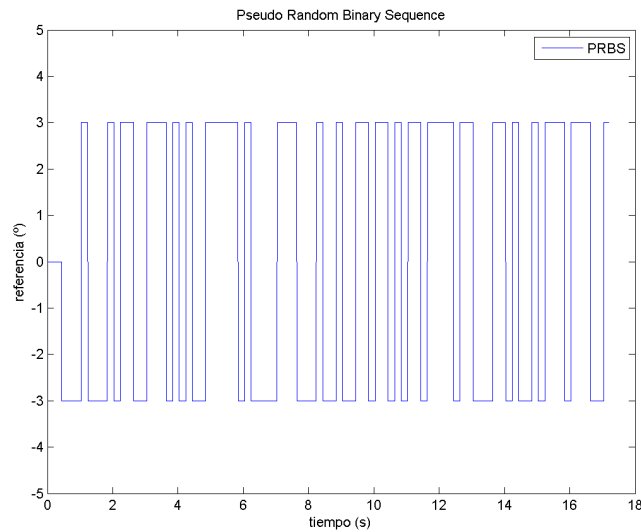


Figura 5.7: Señal PRBS generada

5.5.2 Preprocesado de los datos

Una vez realizado el experimento, los datos obtenidos deben tratarse para conseguir una mejor identificación. El procedimiento a seguir es el siguiente:

1. Seleccionar la información válida del experimento.
2. Eliminar saturaciones y perturbaciones no estacionarias.
3. Hacer que los datos tengan media cero, restando su media o la tendencia que presenten.
4. Filtrar los componentes de altas frecuencias, ya que no son de interés (sección 5.1).

5.6 Resultados

Se han identificado tres modelos de segundo orden (ver sección 4.7) mediante la técnica X-CLOE. En la figura 5.8, figura 5.9 y figura 5.10 se muestran los experimentos de identificación del roll, pitch y yaw respectivamente, junto con la predicción obtenida con la identificación. Se puede observar como el modelo no representa con exactitud al sistema real, pero la mayoría de respuestas entre uno y otro son parecidas. Por ejemplo, en el yaw se puede observar que hay un desfase entre la real y simulada, pero los transitorios tiene la misma forma.

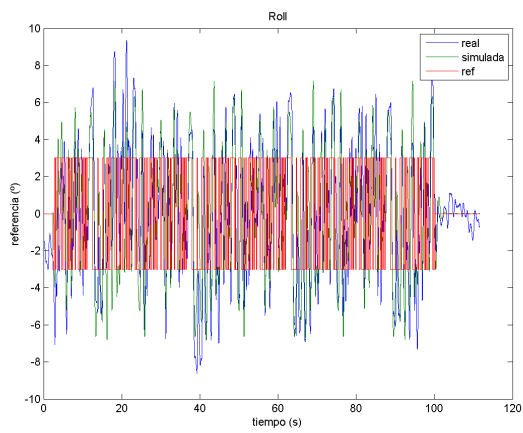


Figura 5.8: Identificación del yaw

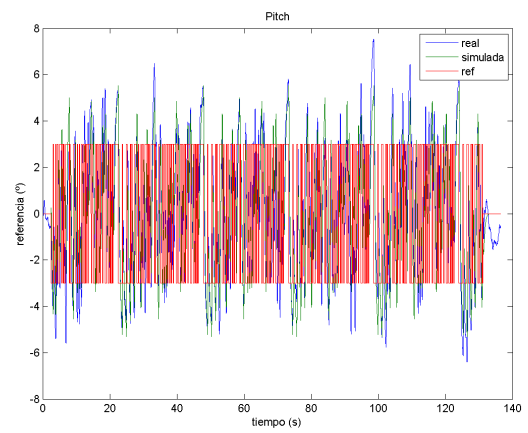


Figura 5.9: Identificación del pitch

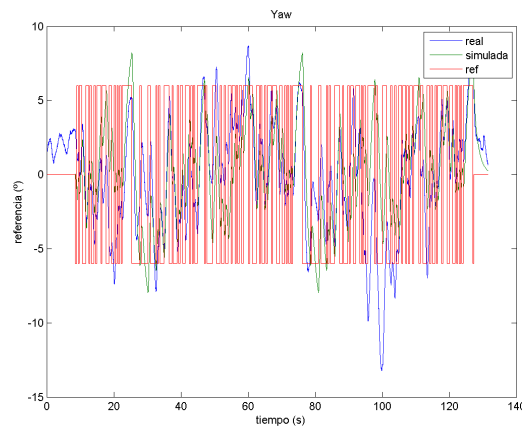


Figura 5.10: Identificación del yaw

Para validar el modelo obtenido, se han representado los errores de predicción [39] con los datos de validación, obteniéndose un ruido casi-normal de pequeña varianza.

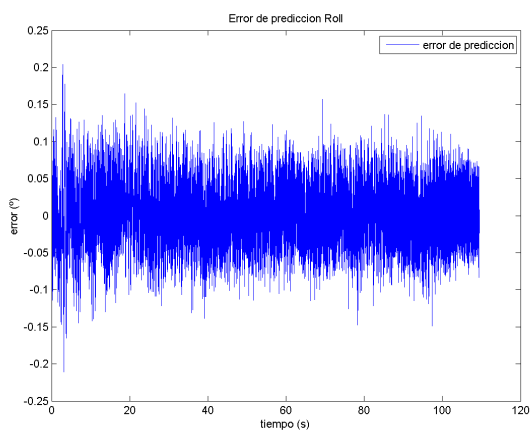


Figura 5.11: Error de predicción del roll

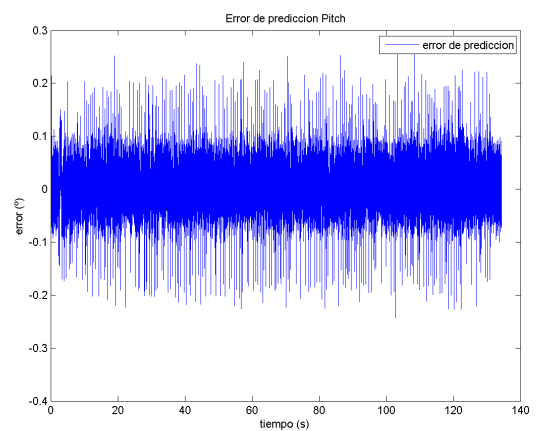


Figura 5.12: Error de predicción del pitch

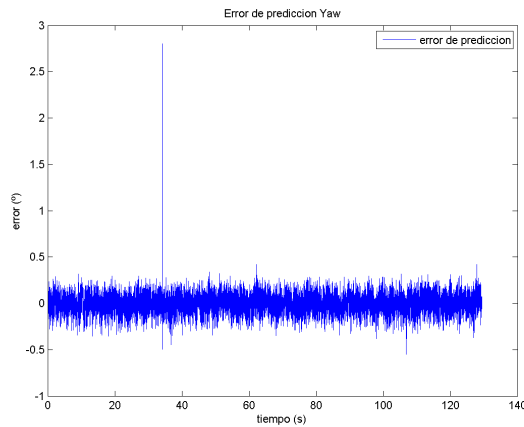


Figura 5.13: Error de predicción del yaw

Con el modelo base de segundo orden de la Ecuación 5.5, en la tabla 5.1 se pueden observar las identificaciones obtenidas para los tres ángulos. A destacar que las funciones identificadas también tienen un polo inestable, tal y como se había modelado en la sección 4.7. Aunque en un primer momento se pensará que una identificación de mayor orden sería más exacta, en la práctica no es así porque la incertidumbre de los parámetros crece llegando incluso a no converger el sistema.

$$G_p(z) = \frac{b_1 z^{-1} + b_2 z^{-2}}{1 + a_1 z^{-1} + a_2 z^{-2}} \quad (5.5)$$

Tabla 5.1: Coeficientes de los modelos

Ángulo	a_1	a_2	b_1	b_2
Roll	-2.006	1.006	-0.5685	0.6195
Pitch	-1.99	0.9894	-1.082	1.138
Yaw	-1.992	0.9919	-0.1187	0.131

Finalmente, con las matrices de covarianzas obtenidas se podría obtener una cota del error en frecuencias. Se deja para un futuro posible trabajo el análisis de la cota de error, el cual no es trivial al estar los diferentes parámetros correlacionados. Esto permitiría el uso de técnicas diferentes a las tratadas en este trabajo.

5.7 Obtención de los parámetros inciertos

Una vez identificados los modelos, se desea obtener la forma equivalente a teórica (ver sección 4.7) y obtener los parámetros inciertos (ver sección 4.5). Para ello, se obtendrá una función de transferencia de la estructura teórica que minimice el error en el dominio frecuencial de interés. Existen diferentes técnicas disponibles para aproximar un modelo a otro deseado. En este caso se ha decidido a usar un algoritmo genético que identifique los parámetros inciertos a partir de una

cota conservativa de los mismos según su valor teórico. Con ello, se obtienen las siguientes aproximaciones (figura 5.14, figura 5.15, figura 5.16). En el caso del yaw, al tratarse de un integrador con frecuencia continua infinita [38], solo se han identificado las frecuencias medias.

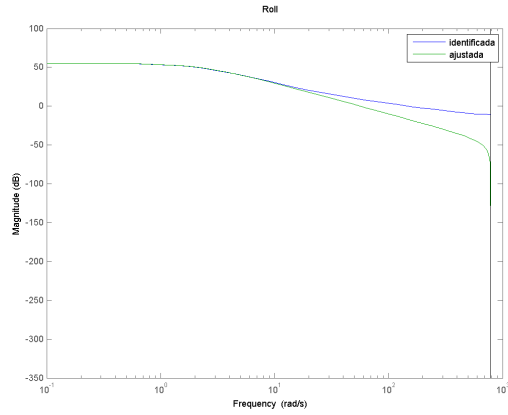


Figura 5.14: Ajuste del roll

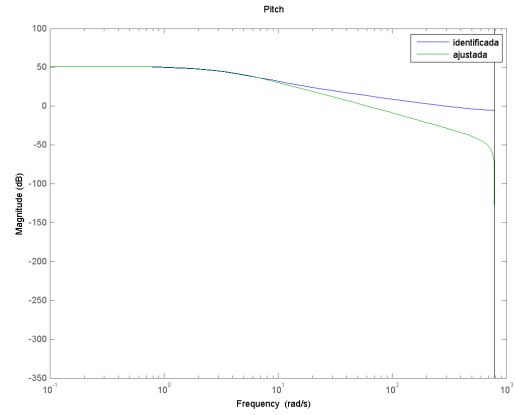


Figura 5.15: Ajuste del pitch

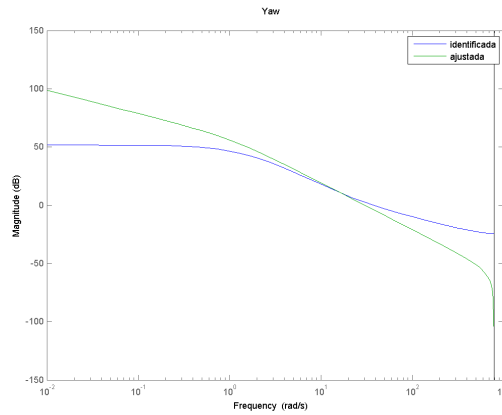


Figura 5.16: Ajuste del yaw

En la tabla 5.2 se encuentran los valores inciertos reidentificados. Se observa un importante aumento de la inercia, posiblemente a causa de dinámica no modelada como la de los motores retardos intrínsecos del sistema. Como se ha visto en la sección 4.8 y las inercias obtenidas, se ha establecido una relación de 0,5 entre las inercias en la rotula y las de vuelo libre.

Tabla 5.2: Parámetros físicos identificados

Parámetro	Medida
J_ϕ	$0.1741kgm^2$
J_θ	$0.0958kgm^2$
J_ψ	$0.0745kgm^2$
k_ϕ	1.9834N
k_θ	1.2646N
k_ψ	0.342N
b_ψ	0.0530

Capítulo 6

Control

6.1 Introducción

El control de sistemas dinámicos consiste en la obtención de acciones de control que permitan llevar y mantener unas salidas a una región delimitada mediante una estrategia de control [38]. Hay dos enfoques principales de orientar el control: control en bucle abierto y en bucle cerrado. El primero se basa en la información del modelo para efectuar las acciones de control que permitan llevarlo a la zona deseada. El segundo utiliza la lectura de ciertas variables para controlar el sistema. Al tratarse de un proceso inestable, es necesario controlar en bucle cerrado y disponer de un controlador que lo estabilice para poder trabajar con él (ver sección 5.3). Por ello, el control de sistemas inestables es un área de enorme importancia y continuo estudio [45].

Los controladores a sintetizar se basarán en el modelo de vuelo libre pero con la inercia de la rótula, excepto en el caso de la subsección 6.4.1 donde la forma de cálculo viene condicionada por la estructura del modelo. Por tanto, los reguladores resultantes deberán ser lo suficientemente robustos para estabilizar la rótula y con el escalado adecuado controlar en vuelo libre.

El diseño de controladores corresponde tanto a un caso práctico de uso de la plataforma desarrollada como es el diseño de controladores previos a la identificación del quadrotor o una comparativa entre diferentes tipos de controladores. Los reguladores a diseñar se pueden clasificar en tres tipos:

- Basados en funciones de transferencia.
- Basados en realimentación del estado.
- Basados en estimación de las perturbaciones.

En las próximas secciones se diseñarán los diferentes controladores. En primer lugar se explicará el tipo de control, y se diseñará detalladamente el control del roll, mostrando su simulación y controlador calculado. Una vez obtenido dicho regulador, al tratarse de un proceso idéntico, se

mostrarán los controladores restantes de dicho tipo. En segundo lugar, en la sección 7.2 se implementarán en la plataforma real y se validarán en la r tula, descart ndose los que no consigan estabilizar o los que no cumplan con los requerimientos. Finalmente los controladores seleccionados se probar n en vuelo libre (secci n 7.3). Las simulaciones realizadas implementar n el control mediante una funci n propia, la cual lo implementar  de la misma forma que luego se programar  en la Raspberry Pi.

Por  ltimo, se alar que se han dejado otros tipos de controladores como robustos (tolerancia del error de modelado), difusos (fuzzy) o predictivos de cara a un posible trabajo, debido a su complejidad de c culo en el caso de sistemas inestables o su alta carga computacional.

6.2 Compromiso especificaciones/robustez

A partir de un modelo conocido, se puede obtener un controlador de orden finito que permita la din mica del sistema cumplir con las especificaciones deseadas. No obstante, estos controladores de altas prestaciones basados en un modelo determinado resultan ser pocos robustos debido a que un ligero cambio del modelo te rico con el real produce grandes cambios en el comportamiento esperado. Como se vio en secci n 5.1, a altas frecuencias la identificaci n es m s inexacta, por tanto en la pr ctica las especificaciones deben actuar en la zona donde la identificaci n tenga una exactitud suficiente.

A continuaci n se puede observar un ejemplo del problema propuesto. Se tiene realmente la planta de la Ecuaci n 6.1 y se ha identificado el modelo de la Ecuaci n 6.2. En la figura 6.1 se representa su comportamiento en frecuencia, donde se observa que a medianas y bajas frecuencias los dos modelos son id nticos. No obstante, si se desean especificaciones agresivas, las diferencias entre los modelos se hacen m s patentes. La figura 6.1 se obtiene realizando el bucle cerrado con una realimentaci n de 50. Mientras te ricamente se obtiene una buena respuesta, en el caso real presenta grandes oscilaciones y un tiempo de establecimiento mucho m s grande, quedando patente la relaci n contraria entre especificaciones y robustez.

$$G_{real}(z) = \frac{5}{(s+5)(s+1)} \quad (6.1)$$

$$G_{real}(z) = \frac{1}{(s+1)} \quad (6.2)$$

En el caso de sistemas estables, para las tiempo de establecimiento se suele usar un tiempo de hasta la mitad de la din mica dominante del sistema (din mica m s lenta). En el caso se sistemas inestables, el tiempo de establecimiento debe ser lo suficientemente r pido para corregir el comportamiento inestable del proceso. A partir de diferentes investigaciones en control de sistemas inestables [46]-[48], para el proceso identificado se ha decidido utilizar unas especificaciones sin oscilaciones de 2 segundos de tiempo de establecimiento.

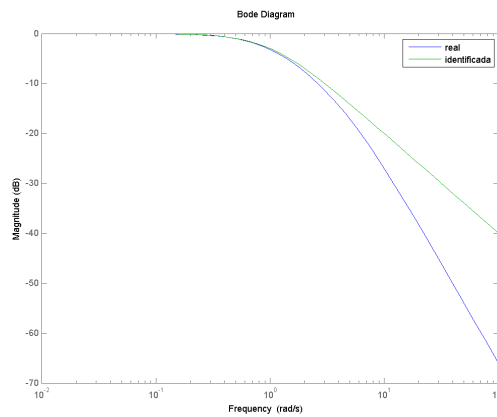


Figura 6.1: Bode de los dos modelos considerados

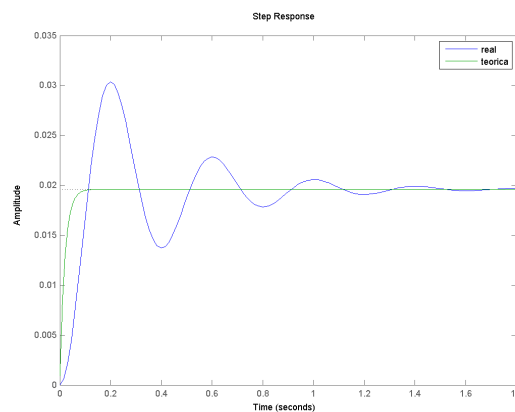


Figura 6.2: Bucle cerrado resultante

Como tanto el roll como el pitch tienen modelos similares a bajas y medias frecuencias, se utilizara un mismo modelo para diseñar los controladores de ambos.

6.3 Control con dos grados de libertad

El control con un grado de libertad (1dof) considera que la entrada al controlador es el error entre la referencia deseada y la medida actual del sistema. Por otra parte, el control con dos grados de libertad (2dof) considera controladores diferentes relacionando por una parte la referencia y por otra la variable controlada. En la figura 6.3 se representa el control clásico con dos grados de libertad. Si $T = R$ se obtiene el control con un grado de libertad [38].

El control con dos grados de libertad permite dividir la dinámica del sistema en rechazo de perturbaciones y seguimiento de referencia (tracking). Como consecuencia, se puede fijar una dinámica mas lenta la cambio de referencia. Esto permite que frente a grandes cambios de referencia, la acción de control resultante sea más pequeña evitando saturaciones y no linealidades del actuador. Cabe destacar que el controlador R es el encargado de estabilizar al sistema, siendo el controlador crítico.

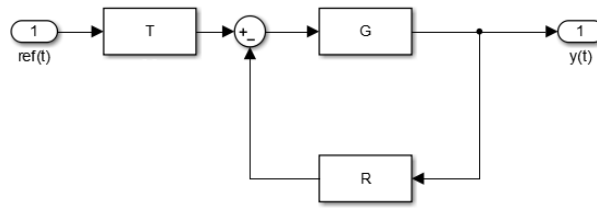


Figura 6.3: Control con dos grados de libertad

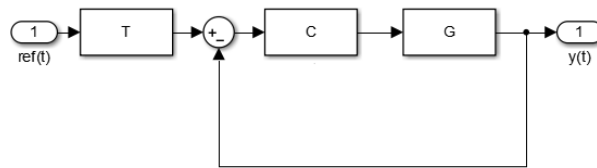


Figura 6.4: Control con dos grados de libertad. Formulación alternativa

En los controladores realizados se ha utilizado la variante mostrada en figura 6.3. La función T se trata de un filtro paso bajo de primer orden. Se ha considerado usar un tiempo de establecimiento de la mitad del tiempo del bucle principal, como en el caso de diseño de observadores de estado [37]. En la figura 6.5 y la figura 6.5 se representan un ejemplo del control con un grado y dos grados usando la estructura comentada. El modelo es el de la Ecuación 6.2 y $C = 25$. Se puede observar la reducción del pico en la acción de control y que el tiempo de establecimiento se ha visto poco afectado.

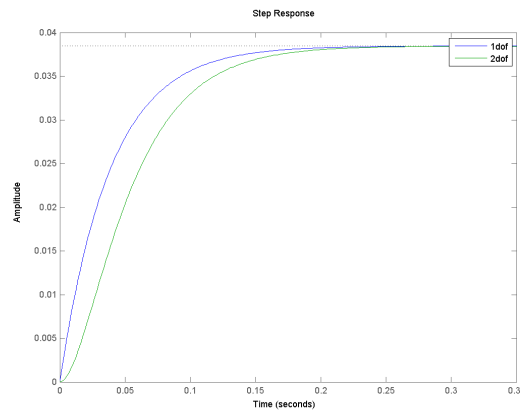


Figura 6.5: Control 2dof. Salida

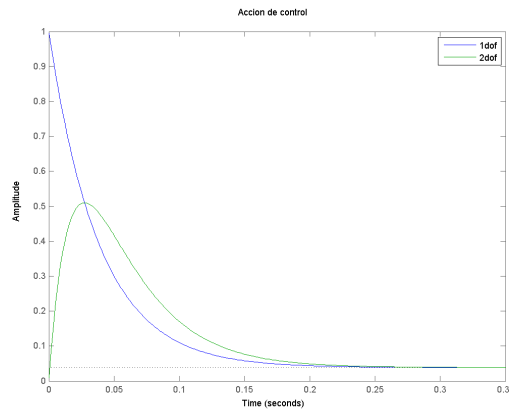


Figura 6.6: Control 2dof. Acción de control

6.4 Control basado en funciones de transferencia

Los controles basados en funciones de transferencia permiten diseñar reguladores a partir de los modelos en representación de función de transferencia, obteniendo controladores en la misma representación.

6.4.1 Control mediante tablas

Los controladores diseñados a partir de tablas son unas de las metodologías más básicas de diseño. A partir de la estructura del modelo, existen tablas desarrolladas empíricamente/minimizando cierta función que según los parámetros del modelo permiten obtener un controlador. Un ejemplo clásico de estas tablas son las de Ziegler-Nichols [49]. Para este trabajo, con el roll se han usado las tablas de [50] que permiten controlar un sistema de segundo orden (SOSPD) con un polo inestable. En el caso del yaw, [51] permite obtener su respuesta considerando su modelo teórico corregido. La mayoría de tablas sintetizan controladores de tipo PID (Ecuación 6.3).

$$C(s) = K_c \left(1 + \frac{1}{T_i s} + T_d s \right) \quad (6.3)$$

$$G_p(s) = \frac{K_m}{(T_{m1}s - 1)(T_{m2}s + 1)} \quad (6.4)$$

$$\begin{cases} \lambda = \frac{1,5}{T_{m2}} \\ T_i = \lambda \left(\frac{\lambda}{T_{m1}} + 2 \right) + T_{m1} \\ K_c = \frac{T_{m1} T_i}{\lambda^2 K_m} \\ T_d = \frac{\lambda \left(\frac{\lambda}{T_{m1}} + 2 \right) T_{m1}}{T_i} \end{cases} \quad (6.5)$$

En la Ecuación 6.4 y la Ecuación 6.5 se muestran la estructura del modelo y las ecuaciones de obtención de los parámetros del controlador para el primer caso. En la figura 6.7 se muestra

la respuesta del sistema a un cambio de referencia escalón y en la figura 6.8 acción de control correspondiente.

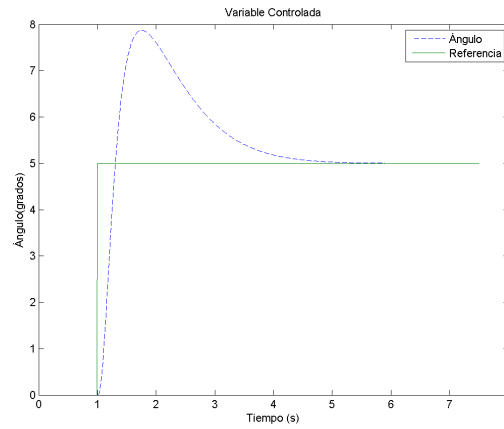


Figura 6.7: Control mediante tablas. Salida

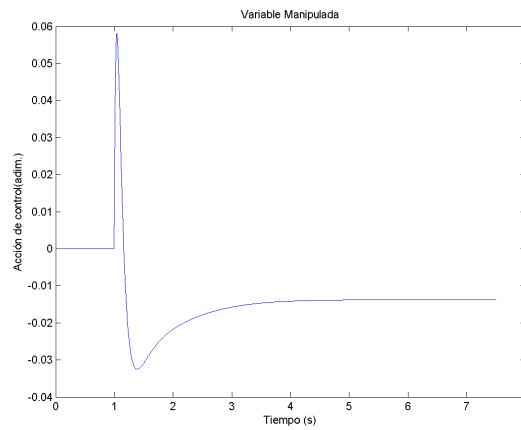


Figura 6.8: Control mediante tablas. Acción de control

Para el diseño del controlador del yaw, se ha usado el controlador PID de la Ecuación 6.6 con la forma del modelo de la Ecuación 6.7. En la Ecuación 6.8 se pueden observar las ecuaciones que permiten realizar el ajuste del PID.

$$C(s) = Kc \left(1 + \frac{s + \frac{1}{T_i}}{s} (T_d s + 1) \right) \quad (6.6)$$

$$G_p(s) = \frac{K_m}{s(T_m s + 1)} \quad (6.7)$$

$$\begin{cases} \lambda = \frac{1,5}{T_{m2}} \\ T_i = 4\lambda \\ K_c = \frac{1}{K_m\lambda} \\ T_d = T_m \end{cases} \quad (6.8)$$

En la tabla 6.1 se encuentran los parámetros obtenidos con las tablas para el diseño de los reguladores. Cabe destacar que para su implementación se han discretizado mediante el método de la bilineal [43].

Tabla 6.1: Parámetros de ajuste mediante el método de las tablas

Ángulo	K_c	T_i	T_d
Roll/pitch	0.0077	1.9533	0.2625
Yaw	0.0049	2.1106	0.3518

6.4.2 Control PID

En los trabajos anteriores (ver sección 2.5) el control que se realizaba era un PID. No obstante, a diferencia de los del apartado anterior, al disponer de la medida de la derivada del angulo se usaba esta medida junto con la derivación de la referencia como derivada del error. El ajuste del PID era online, es decir, se ajustaba mediante prueba y error en plena ejecución. La Ecuación 6.9 representa el PID discreto utilizado.

$$\begin{cases} u_k = K_p(ref - y) \\ u_i = u_i + K_i(ref - y) \\ u_d = K_d(ref - \dot{y}) \\ u = u_k + u_i + u_d \end{cases} \quad (6.9)$$

Mediante la metodología de prueba y error, se han obtenido dos conjuntos de controladores a probar: controlador PD y controlador PID. El controlador PID permite seguimiento de la referencia, pero dificultando su control. La figura 6.9 se muestra la respuesta de la salida y en la figura 6.10 la acción de control. En la tabla 6.2 figuran los parámetros de ajuste tanto de los PD como de los PID.

Tabla 6.2: Parámetros de ajuste de los PID

Ángulo	K_p	T_i	T_d
Roll/pitch PD	0.0075	0	0.0035
Yaw PD	0.02	0	0.01
Roll/pitch PID	0.008	0.00001	0.0045
Yaw PID	0.02	0.0001	0.01

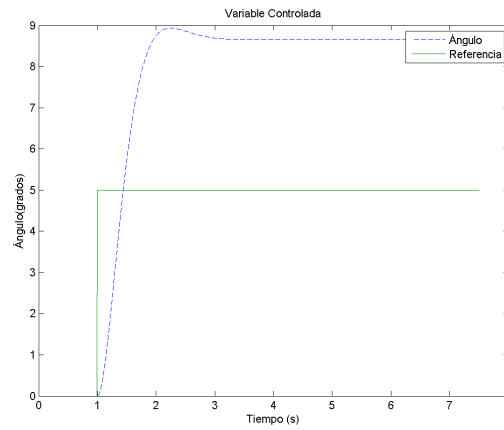


Figura 6.9: Control PD. Salida

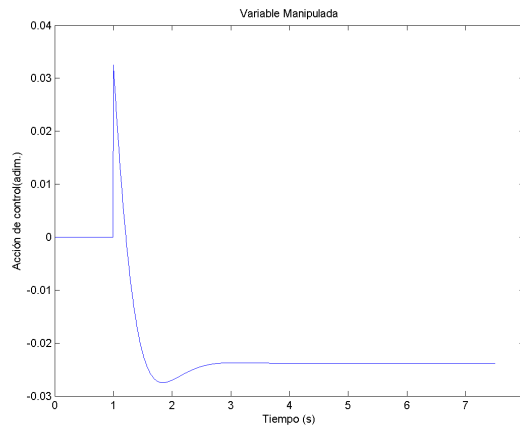


Figura 6.10: Control PD. Acción de control

6.4.3 Reguladores algebraicos

Los reguladores algebraicos [52] permiten situar los polos del bucle cerrado en el lugar deseado. Esto se consigue encontrando un controlador $\frac{Q(z)}{P(z)}$ que permita que la ecuación del bucle cerrado (Ecuación 6.10) tenga los polos donde se deseen, donde el modelo es $\frac{B(z)}{A(z)}$.

$$BC(z) = \frac{B(z)Q(z)}{A(z)P(z) + B(z)Q(z)} \quad (6.10)$$

Las especificaciones se discretizan ($T=4ms$) mediante la Ecuación 6.11 [43]. Para resolver el problema, hay que encontrar el regulador $\frac{Q(z)}{P(z)}$ que sitúa los polos de la ecuación característica ($A(z)P(z) + B(z)Q(z)$) en las especificaciones. En este caso, considerando que el sistema es de segundo orden, se situarán dos polos en las especificaciones, siendo los restantes (λ) necesarios en el origen.

$$z_{especific} = e^{-\frac{4T}{t_e}} \quad (6.11)$$

Para que el problema tenga una una solución, en el caso particular del quadrotor, se deben cumplir las siguientes restricciones:

1. Regulador realizable. $\text{grado}(P) \geq \text{grado}(Q)$
2. Ecuación característica. $\text{grado}(P) + 2 = 2 + \lambda$
3. Numero de incógnitas y numero de ecuaciones. $\text{grado}(P) + 1 + 2 = \text{grado}(P) + \text{grado}(Q) + 1 + 1$

Resolviendo dichas consideraciones, se obtiene que $\text{grado}(P) = 1$, $\text{grado}(Q) = 1$ y $\lambda = 1$. Por tanto, a partir de la ecuación diofántica resultante, se obtienen los coeficientes del regulador. Para el caso del roll y pitch, la figura 6.11 muestra la salida y la figura 6.12 la acción de control que permite controlarlo. En la tabla 6.3 estan los coeficientes de los controladores (Ecuación 6.12).

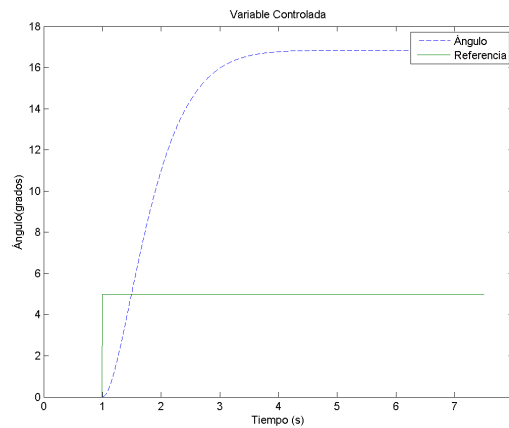


Figura 6.11: Regulador algebraico. Salida

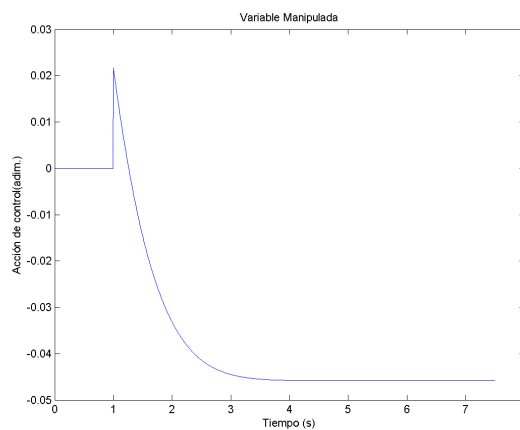


Figura 6.12: Regulador algebraico. Acción de control

$$C(z) = \frac{q_1 z + q_0}{p_1 z + p_0} \quad (6.12)$$

Tabla 6.3: Coeficientes de los reguladores algebraicos

Ángulo	q_1	q_0	p_1	p_0
Roll/pitch	0.2732	-0.26933	1	0.007992
Yaw	0.2603	-0.258	1	0.002299

6.4.4 IMC

El “internal model control” o IMC es una técnica de control basada en establecer unas especificaciones dadas cancelando la dinámica del modelo. En la se muestra el diagrama de bloques del IMC. El controlador C incluye la dinámica del modelo, y G_{modelo} permite corregir las desviaciones del real frente al supuesto. Debido a su enfoque de corrección entre modelo y realidad se podría considerar en la sección 6.6, pero se ha decidido situarlo en esta sección al no ser estimar las perturbaciones una característica explícita del mismo.

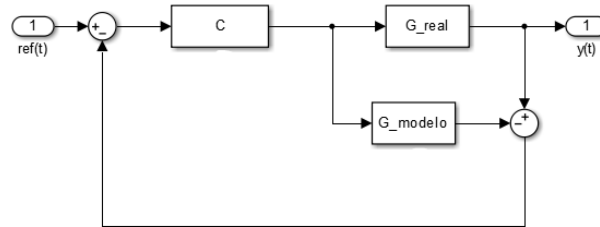


Figura 6.13: Internal model control

No obstante, para el trabajo se ha utilizado una versión modificada discreta de la propuesta en [53] para el control de sistemas inestables, la cual realiza una primera estabilización del bucle estable y una segunda fijando las especificaciones. El controlador usado es el visto en la subsección 6.5.1 y un tiempo de establecimiento del bucle general del doble, es decir, 4 segundos.

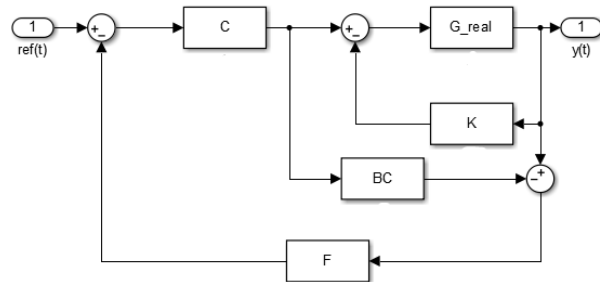


Figura 6.14: IMC modificado

Con ello, se obtienen los controladores y los bucles cerrados (Ecuación 6.13) correspondientes en la tabla 6.4. En la figura 6.15 y la figura 6.15 se observa el comportamiento del IMC para el caso del roll y pitch.

$$C(z) = \frac{q_2 z^2 + q_1 z + q_0}{p_2 z^2 + p_1 z + p_0} \quad (6.13)$$

Tabla 6.4: Coeficientes del IMC

Ángulo	q_2	q_1	q_0	p_2	p_1	p_0
Roll/pitch BC	0	0	0.0593	1	-1.8947	0.8952
Yaw BC	0	0	0.0089	1	-1.9796	0.9797
Roll/pitch C	2.68e-4	-5.087e-4	2.4039e-4	1	-1.9927	0.9920
Yaw C	0.0018	0.0036	0.0018	1	-1.9920	0.992

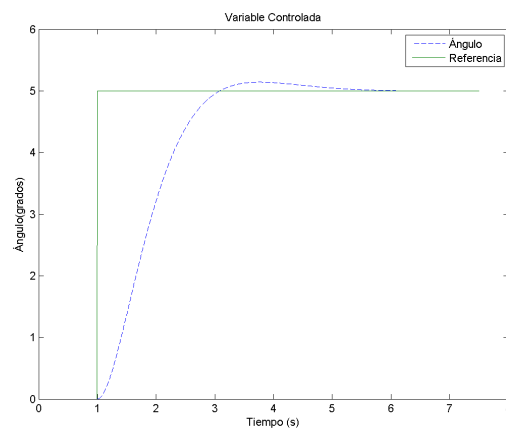


Figura 6.15: IMC. Salida

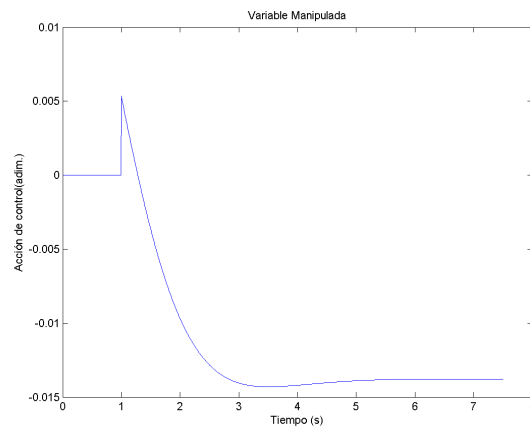


Figura 6.16: IMC. Acción de control

6.5 Control basado en espacio de estados

El control basado en espacio de estado consiste la obtención del estado (x) del procesos (ver sección 4.7) y generar un controlador que realimente dicho estado. En el caso del quadrotor el estado se obtiene directamente de las medidas ($x_1 = y, x_2 = \dot{y}$) y el filtro de Kalman, suponiendo una ventaja frente al control mediante funciones de transferencia. En la figura 6.17 se muestra el esquema de un control por realimentación del estado donde se mide directamente el estado (sin necesidad de un observador).

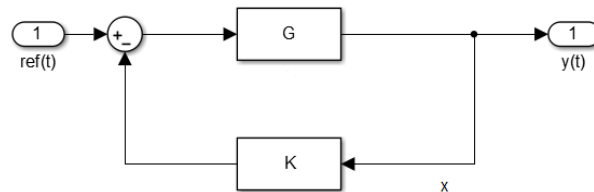


Figura 6.17: Realimentación del estado

6.5.1 Asignación de polos

El control en espacio de estados mediante la asignación de polos es el equivalente al regulador algebraico en espacio de estado. Se diseña una realimentación K que permita situar los polos en bucle cerrado en el lugar deseado. La Ecuación 6.14 representa los valores propios del bucle cerrado, donde se obtiene la K para situar los polos [37].

$$\det(zI - (A - BK)) \quad (6.14)$$

Con ello, mediante las especificaciones deseadas se obtienen los controladores $K = [K_1 K_2]$ de la tabla 6.5. En la figura 6.18 y la figura 6.19 se muestra el comportamiento de la salida y la acción de control respectivamente para el caso del roll y pitch. La referencia a aplicar se ha ponderado por el valor K_1 , ya que en el caso de un integrador resulta en seguimiento de la referencia [38].

Tabla 6.5: Realimentación del estado por asignación de polos

Ángulo	K_1	K_2
Roll/pitch	0.0144	0.0078
Yaw	0.0962	0.0519

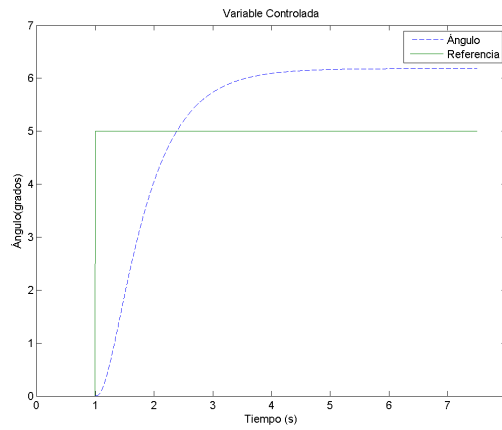


Figura 6.18: Asignación de polos en espacio de estados. Salida

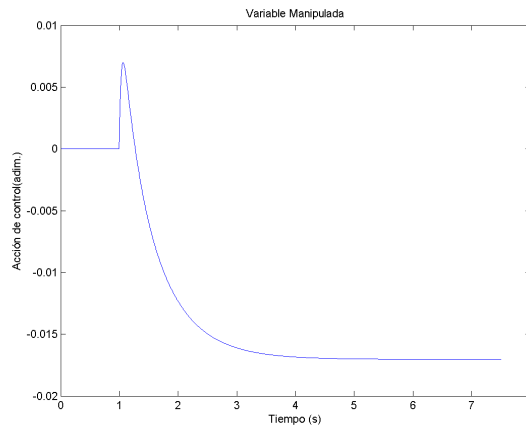


Figura 6.19: Asignación de polos en espacio de estados. Acción de control

6.5.2 Control LQR

El control LQR o “linear quadratic regulator” permite calcular una realimentación óptima del estado a partir de una función a minimizar (en el caso discreto la Ecuación 6.15) [37]. Las ponderaciones se han calculado normalizando los valores de entrada y salida en base a [54], [55]. Las ponderaciones resultantes son $Q = [100; 05]$ y $R = [100000]$.

$$J = \sum_{i=0}^{\infty} (x_k^T Q x_k + u_k^T R u_k) \quad (6.15)$$

En la tabla 6.6 se encuentran los controladores resultantes de la optimización. La respuesta para el roll y pitch se representa en la figura 6.20 y la figura 6.21. Como en el caso anterior, se pondera la referencia con el parámetro K_1 para que en el caso teórico en vuelo libre se siga la referencia.

Tabla 6.6: Realimentación del estado por LQR

Ángulo	K_1	K_2
Roll/pitch	0.0095	0.0071
Yaw	0.0099	0.0092

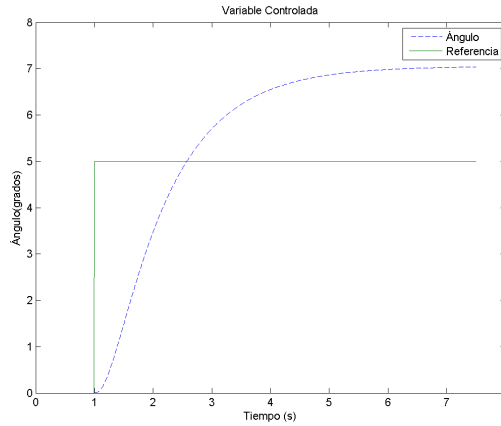


Figura 6.20: LQR. Salida

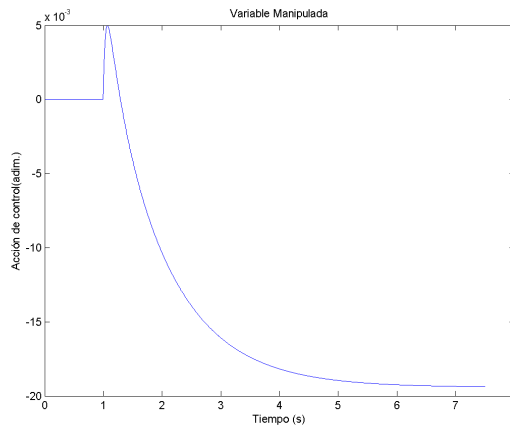


Figura 6.21: LQR. Acción de control

6.6 Control basado en estimación de las perturbaciones

Los controladores basados en estimación de las perturbaciones permiten obtener a partir de un modelo, las entradas al sistema y su respuesta, una estimación de la perturbación equivalente al comportamiento no explicado por el modelo [56]. Como consecuencia, se puede realizar una prealimentación de la perturbación estimada permitiendo al sistema real aproximarse al modelo utilizado para el diseño. Aunque la mayoría de controles de este tipo basan su control principal en espacio de estado ([56], incluido este trabajo), la estructura permite diseñar independientemente el estimador de la perturbación y el control propiamente dicho.

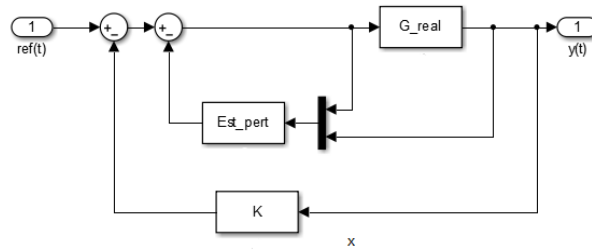


Figura 6.22: Estimación de perturbaciones

6.6.1 ADRC

El “Active Disturbance Rejection Control” o ADRC, es un observador de estado extendido (ESO) para sistemas formados por cadenas de integradores [57]. Si se considera que la perturbación es constante, se obtiene el modelo de la Ecuación 6.16.

$$\begin{cases} \begin{bmatrix} x_{1,k+1} \\ x_{2,k+1} \\ d_{k+1} \end{bmatrix} = \begin{bmatrix} 1 & T & 0 \\ 0 & 1 & Ts \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_{1,k} \\ x_{2,k} \\ d_k \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{2Tkl_1}{J} \\ 0 \end{bmatrix} u \\ y = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ d_k \end{bmatrix} \end{cases} \quad (6.16)$$

A partir del modelo, se puede obtener un observador del estado que permita estimar el valor de la perturbación. Como en el caso del quadrotor se dispone de una medida no sesgada del estado, no hace falta estimarlo, por tanto se construirá un observador del estado reducido que solo estime la perturbación. El sistema extendido puede desglosarse en la Ecuación 6.17 donde los componentes 1 se refieren a los estados conocidos y los 2 a la perturbación. A partir del sistema descompuesto, se le asigna al observador L la dinámica resolviendo Ecuación 6.18, la cual situa el polo en 0.99. Construyendo la Ecuación 6.19 se tiene el observador reducido discreto. Finalmente, se debe multiplicar por $\frac{J}{2kl_1}$ para expresarlo en las mismas unidades que la acción de control y poder realizar la prealimentación.

$$\begin{cases} x_{k+1} = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} x_k + \begin{bmatrix} B_1 \\ B_2 \end{bmatrix} u \end{cases} \quad (6.17)$$

$$\det(zI - (A_{22}^T - A_{12}^T L^T)) \quad (6.18)$$

$$\begin{cases} A = A_{22} - LA_{12} \\ B = \begin{bmatrix} B_1 - LB_2 & (A_{22} - LA_{12})L + A_{21} - LA_{11} \end{bmatrix} \\ C = 1 \\ D = \begin{bmatrix} 0 & L \end{bmatrix} \end{cases} \quad (6.19)$$

En la tabla 6.7 se observa los valores del observador realizado para el roll/pitch y el yaw. En la figura 6.23 y en la figura 6.24 se simula el sistema del roll/pitch con la prealimentación de la perturbación estimada y el control desarrollado en la sección 6.5. En la figura 6.25 se muestra la perturbación equivalente, ya que se ha diseñado para un doble integrador pero se ha simulado con las ecuaciones de la rotula. Como consecuencia, la perturbación estimada explica la acción de control en régimen permanente, ya que un sistema integrador necesita que sea nula para permanecer en reposo. Como puede observarse, la respuesta cumple con el tiempo de las especificaciones y sin oscilaciones.

Tabla 6.7: Observador de la perturbación. ADRC

Ángulo	A	B	D
Roll/pitch	0.99	[-37.09 0 -0.025]	[0 0 2.5]
Yaw	0.99	[-5.547 0 -0.025]	[0 0 2.5]

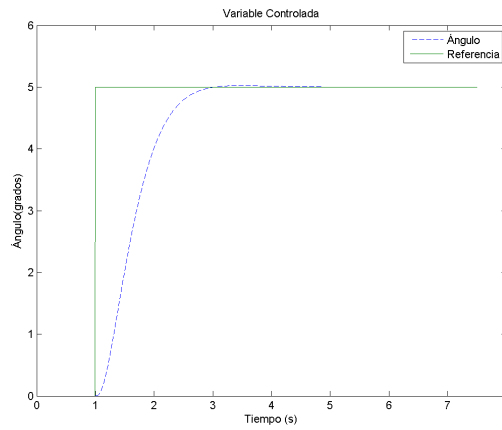


Figura 6.23: ADRC. Salida

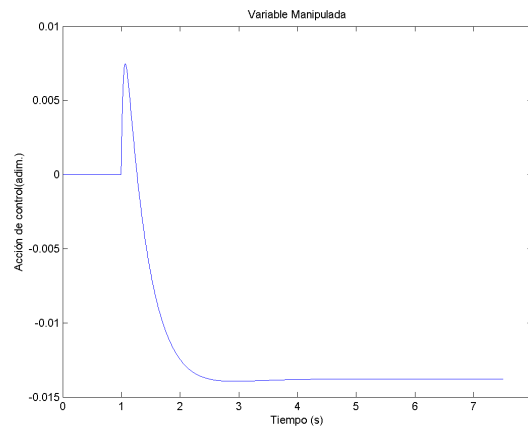


Figura 6.24: ADRC. Acción de control

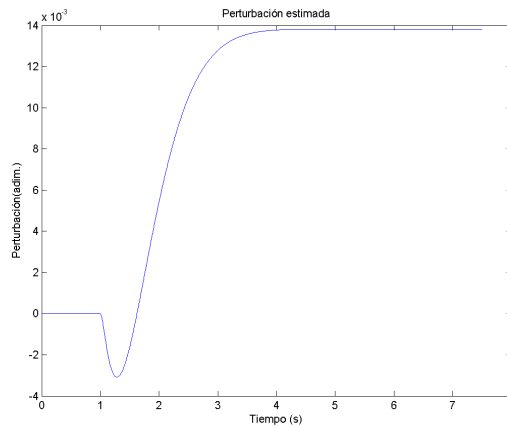


Figura 6.25: ADRC. Perturbación estimada

6.6.2 UDE

El “Uncertainty and Disturbance Estimator” o UDE permite la estimación de la perturbación mediante la predicción del siguiente estado [58]. Toda respuesta que no se pueda explicar mediante el modelo se considerará perturbación.

Se tiene el modelo en espacio de estado considerando la perturbación (Ecuación 6.20). Si se despeja la perturbación (Ecuación 6.21) de la ecuación aplicando la transformada Z, se obtiene un sistema no causal. B^+ es la pseudoinversa $B^+ = (B^T B)^{-1} B^T$. No obstante, el sistema será causal si se incluye un filtro paso bajo en la estimación de la perturbación (Ecuación 6.21), donde $b = (1 - a)$.

$$x_{k+1} = Ax_k + Bu_k + B\hat{d}_k \quad (6.20)$$

$$\hat{d}_k = B^+(zx_k - Ax_k - Bu_k) \quad (6.21)$$

$$\hat{d}_k = B^+ \frac{a}{z - b} (zx_k - Ax_k - Bu_k) \quad (6.22)$$

Para el caso particular del quadrotor como doble integrador, la expresión de la Ecuación 6.23 muestra la expresión final de la estimación de la perturbación.

$$\hat{d}_k = \frac{Ja(z - 1)}{T2kl_1(z - (1 - a))} \dot{y}_k - \frac{a}{z - (1 - a)} u_k \quad (6.23)$$

Como resultado, se obtiene un estimador de perturbaciones con un solo parámetro de ajuste. Como en el caso anterior, se ha establecido el filtro del UDE en 0.99 (tabla 6.8) y se ha utilizado el mismo controlador para las simulaciones. En la figura 6.26 se muestra la salida, en la figura 6.27

la acción de control y en la figura 6.28 la perturbación estimada. Como puede observarse, la perturbación estimada en el UDE posee una dinámica más rápida al considerarse variante y no una constante como en el caso del ADRC.

Tabla 6.8: Observador de la perturbación. UDE

Ángulo	a
Roll/pitch	0.01
Yaw	0.01

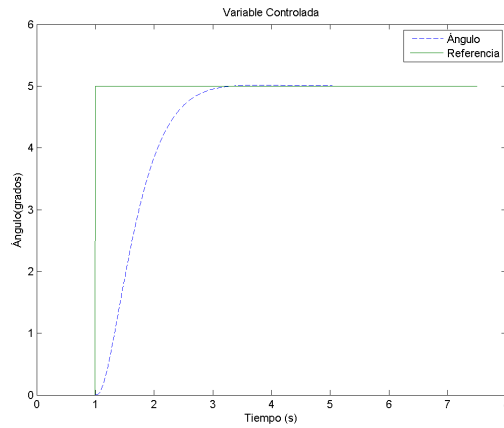


Figura 6.26: UDE. Salida

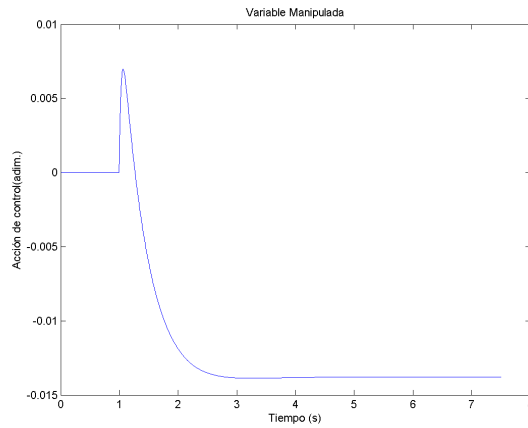


Figura 6.27: UDE. Acción de control

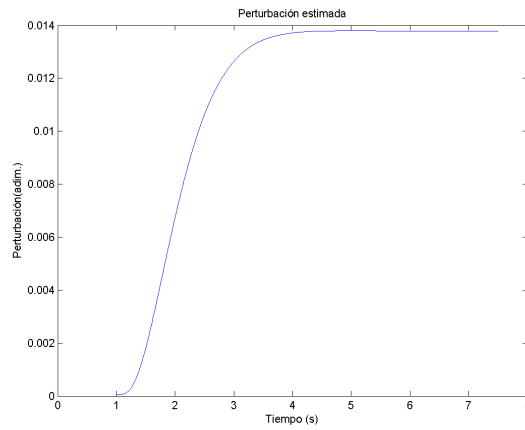


Figura 6.28: UDE. Perturbación estimada

6.7 Control de posición

El control de posición en un quadrotor consiste en situar al vehículo en la posición del espacio deseada. Para ello se suele utilizar una estrategia en cascada con los ángulos para el caso de las componentes x,y (figura 6.29) mientras que en la altura o componente z la acción resultante resulta sumada a todos los motores (ver sección 4.2).

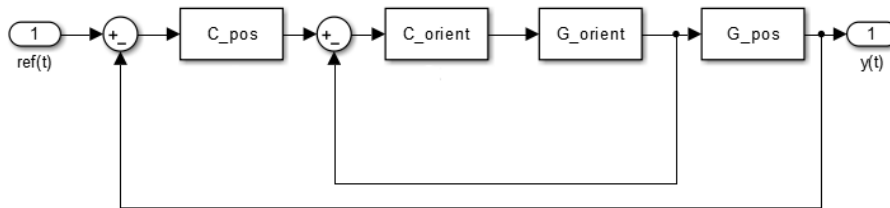


Figura 6.29: Control de posición. Estrategia en cascada

Este el presente trabajo se ha centrado a controlar la orientación del quadrotor, el cual es el control crítico [5]. No obstante, para las pruebas en vuelo libre es necesario implementar un control básico de altura. Por tanto, se ha realizado un control básico por realimentación del estado para controlar la posición en el espacio. Su ajuste se ha realizado online para simplemente conseguir estabilizarla mediante el HMI. En un futuro trabajo se puede profundizar en el control de posición y el seguimiento de trayectorias.

Capítulo 7

Validación

7.1 Introducción

Una vez diseñados los controladores y simulados correctamente, se van a implementar y validar en el sistema real. A parte de las consideraciones del Capítulo 6, se han realizado saturaciones en las diferentes acciones de control para evitar que si el sistema se vuelve inestable puede romper su entorno o herir al usuario.

En un primer lugar, los controladores se realizará una primera selección entre los reguladores que estabilicen el sistema real con la rótula. A continuación, se analizará la bondad de los diferentes reguladores ante unas referencias preestablecidas para realizar una segunda selección. Finalmente, los controladores resultantes se analizará su comportamiento en vuelo libre, teniendo en cuenta las consideraciones de reajuste de las ganancias de la sección 4.8.

7.2 Comportamiento en la rótula

En primer lugar se han probado todos los controladores diseñados en la rótula o banco de pruebas. Se ha diseñado el experimento de la figura 7.1, el cual actúa sobre las referencias una a una y finalmente todas juntas para estudiar el acoplamiento entre las diferentes variables. El tiempo entre cambio se ha fijado en 5 segundos, mayor al tiempo de establecimiento usado para el diseño de los controladores. La amplitud se ha seleccionado acorde a los movimientos normales de los ángulos, siendo más grande para el caso del yaw. El experimento se ha realizado mediante el modo patrón del HMI.

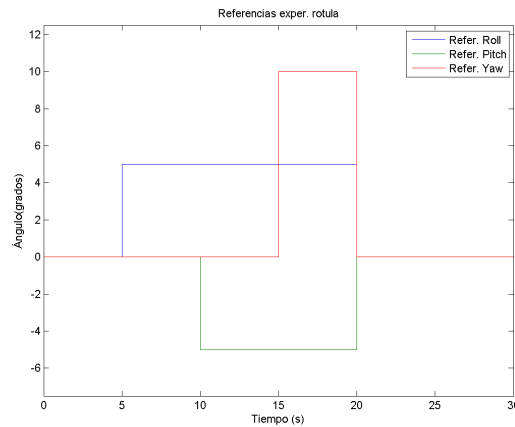


Figura 7.1: Referencias del experimento realizado en la r tula.

7.2.1 Controladores descartados

Los controladores que no han sido capaces de estabilizar el quadrotor son los controladores dise ados a partir de tablas, los reguladores algebraicos y el IMC. Todos ellos tienen en com n que para realizar el control no utilizan la medida de la derivada del  ngulo. El dise o mediante tablas resulto en un controlador demasiado agresivo, llegando a saturar continuamente. Por el contrario, el regulador algebraico resulto ser poco agresivo, siendo incapaz de estabilizar el quadrotor.

Por tanto, una primera conclusi n es que para el caso del quadrotor es necesario realimentar con el valor angular y su derivada, como es el caso de la realimentaci n del estado en el controlador PID modificado de la subsecci n 6.4.2. En el caso del IMC el bucle de estabilizaci n s  lo ten a, pero el bucle del IMC no utilizaba la medida de la velocidad angular para corregir la actuaci n, saturando la correcci n constantemente.

En los siguientes subapartados se van a mostrar las gr ficas de los experimentos realizados. En el caso del UDE y ADRC se van a realizar tres experimentos por cada estimador, correspondientes a el PD, asignaci n de polos en espacio de estados y LQR con la prealimentaci n de la perturbaci n estimada.

7.2.2 Control PID

Control PD dise ado en la subsecci n 6.4.2.

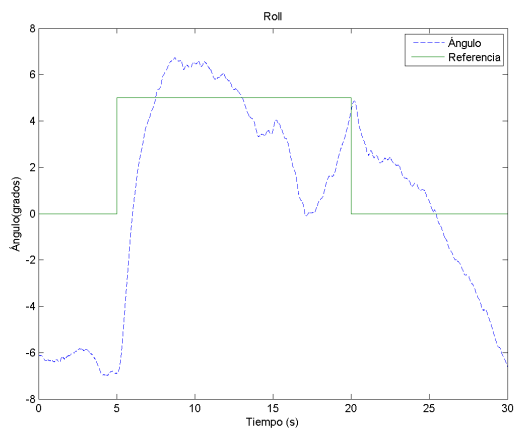


Figura 7.2: PD-roll. Experimento rótula

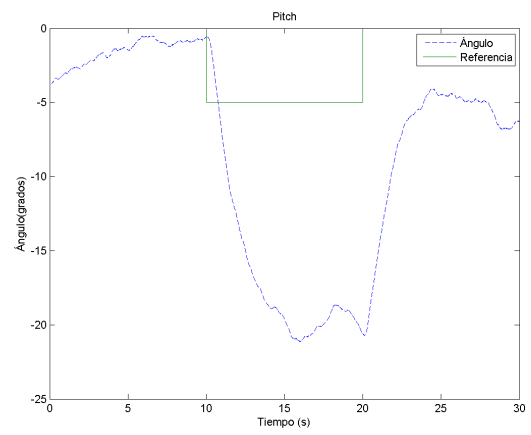


Figura 7.3: PD-pitch. Experimento rótula

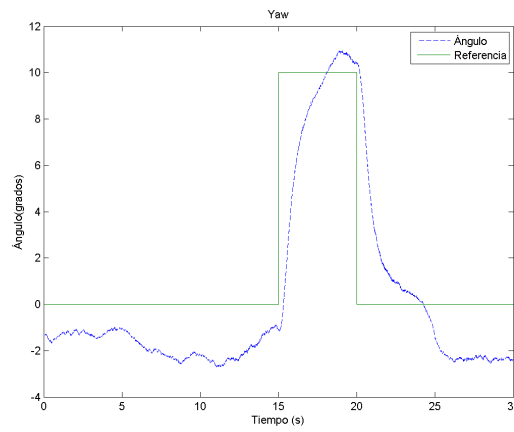


Figura 7.4: PD-yaw. Experimento rótula

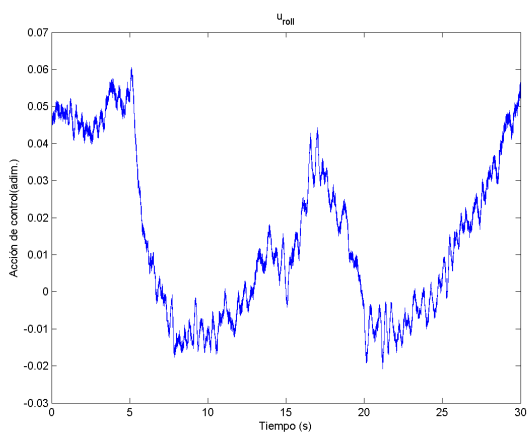


Figura 7.5: PD-ur. Experimento rótula

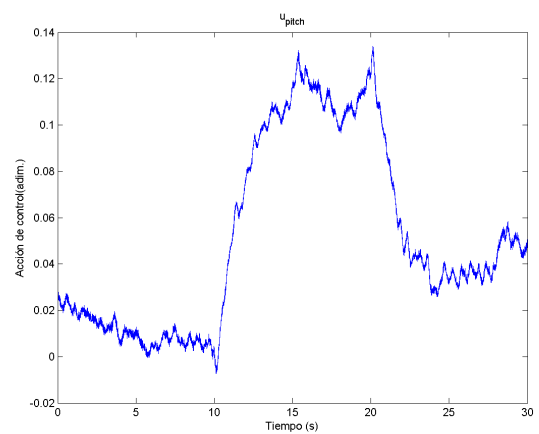


Figura 7.6: PD-up. Experimento rótula

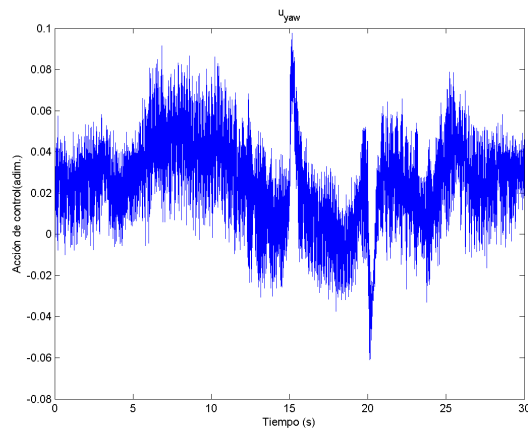


Figura 7.7: PD-uy. Experimento rótula

Control PID diseñado en la subsección 6.4.2.

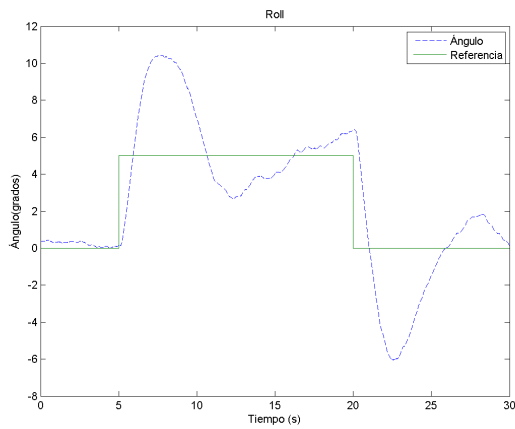


Figura 7.8: PID-roll. Experimento rótula

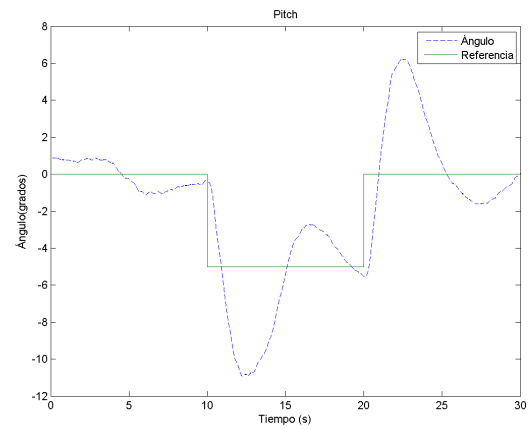


Figura 7.9: PID-pitch. Experimento rótula

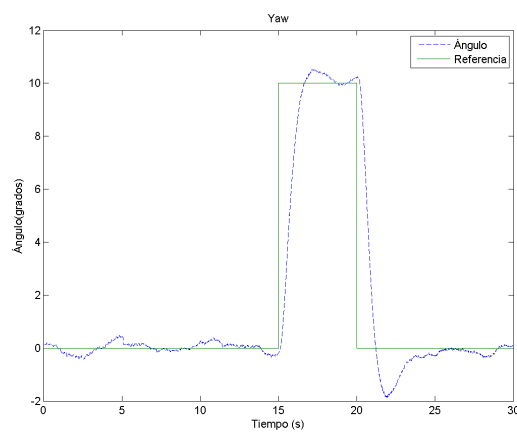


Figura 7.10: PID-yaw. Experimento rótula

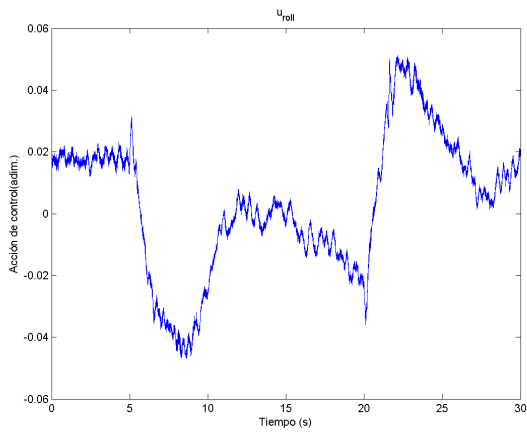


Figura 7.11: PID-ur. Experimento r3tula

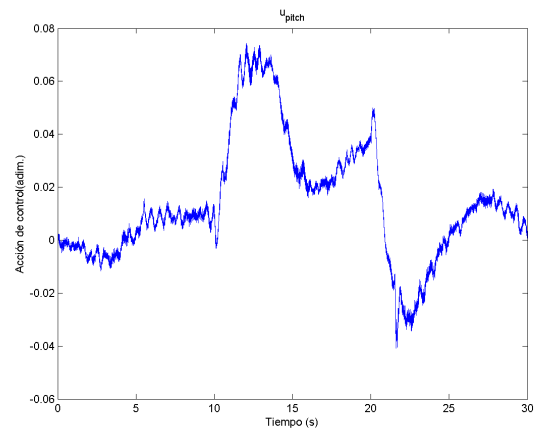


Figura 7.12: PID-up. Experimento r3tula

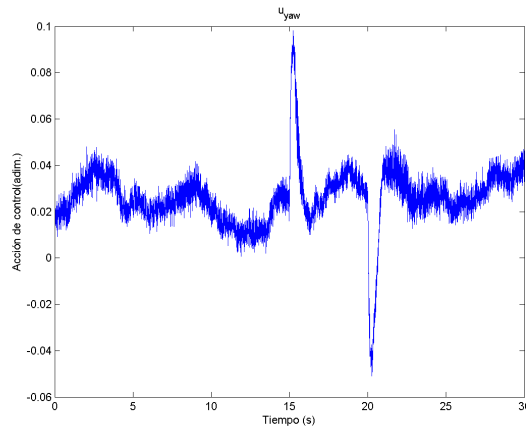


Figura 7.13: PID-uy. Experimento r3tula

7.2.3 Asignación de polos

Control de realimentación del estado diseñado asignando los polos del bucle cerrado (subsección 6.5.1).

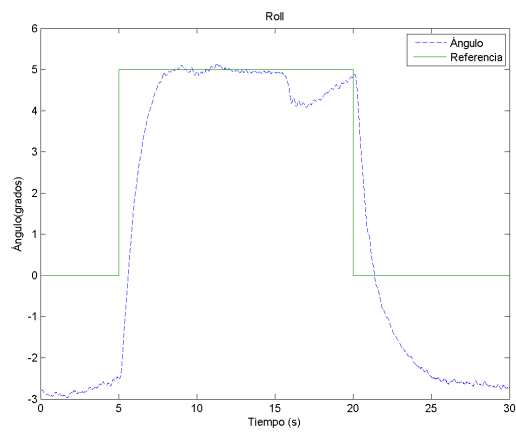


Figura 7.14: Asignación polos-roll. Experimento rótula

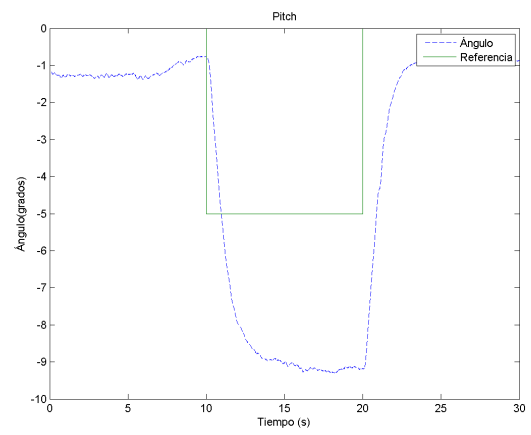


Figura 7.15: Asignación polos-pitch. Experimento rótula

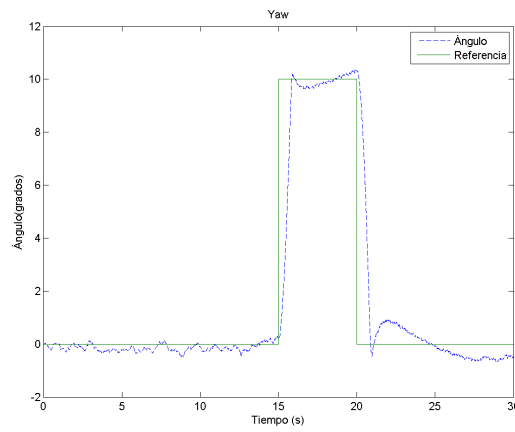


Figura 7.16: Asignación polos-yaw. Experimento rótula

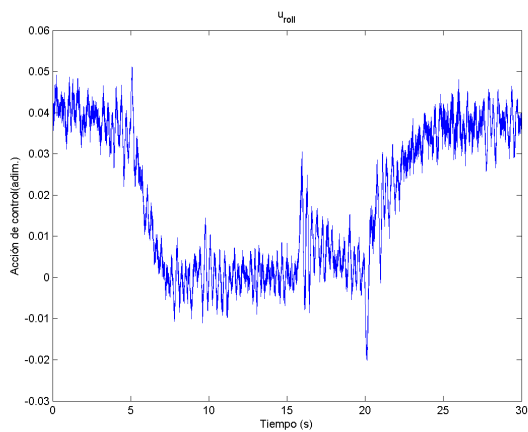


Figura 7.17: Asignación polos-ur. Experimento rótula

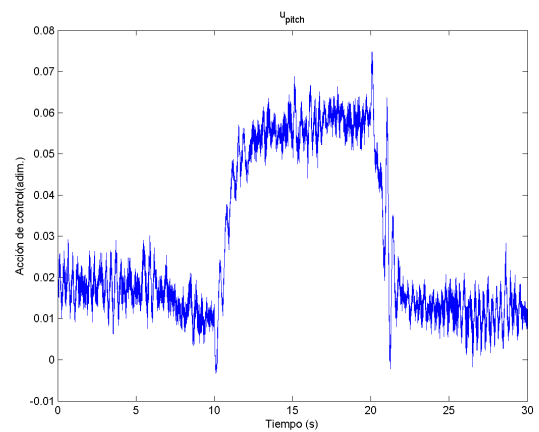


Figura 7.18: Asignación polos-up. Experimento rótula

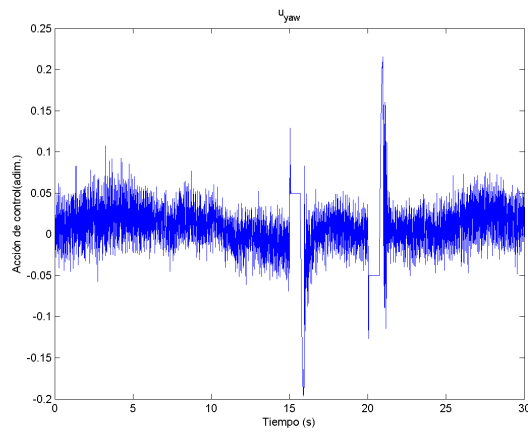


Figura 7.19: Asignación polos-uy. Experimento rótula

7.2.4 Control LQR

Control de realimentación del estado LQR (subsección 6.5.2).

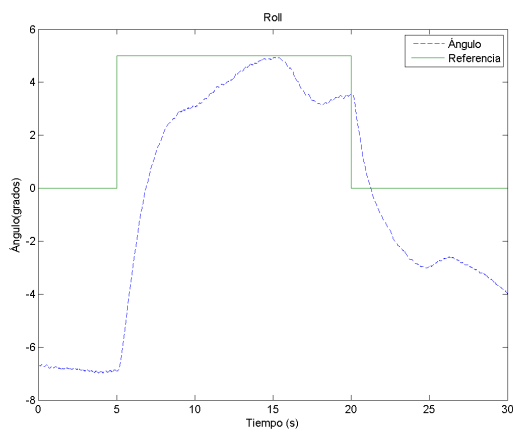


Figura 7.20: LQR-roll. Experimento rótula

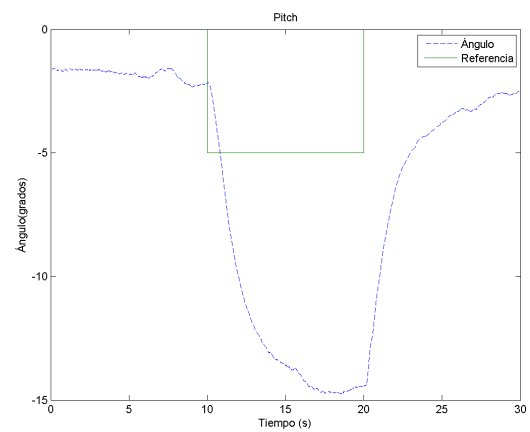


Figura 7.21: LQR-pitch. Experimento rótula

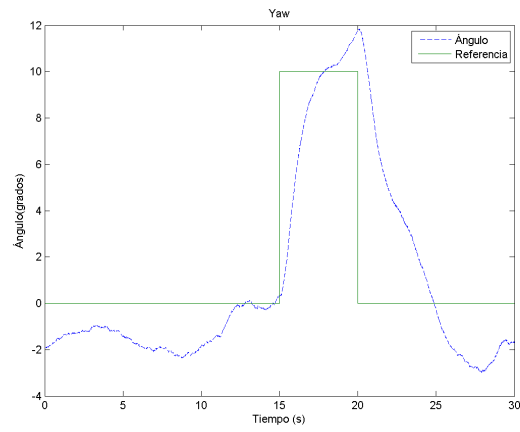


Figura 7.22: LQR-yaw. Experimento rótula

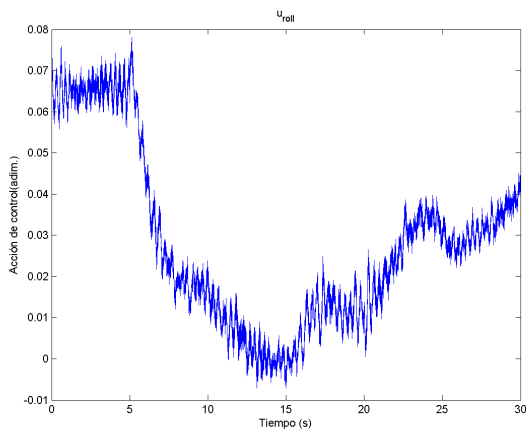


Figura 7.23: LQR-ur. Experimento rótula

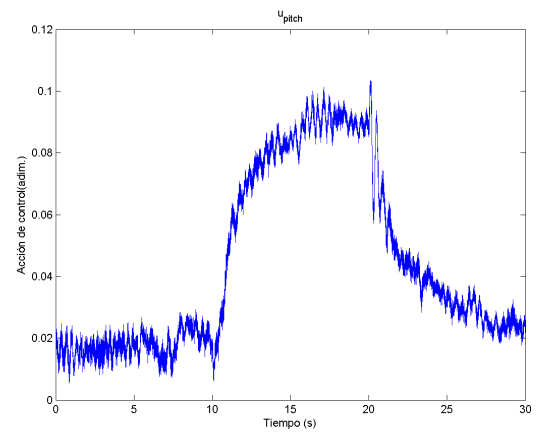


Figura 7.24: LQR-up. Experimento rótula

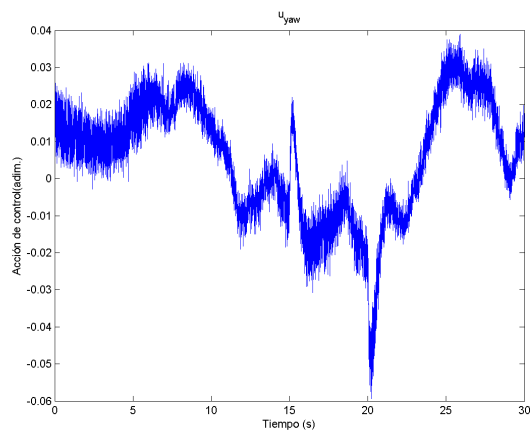


Figura 7.25: LQR-uy. Experimento rótula

7.2.5 ADRC

Control prealimentando la perturbación estimada mediante el ADRC (subsección 6.6.1). Se han realizado tres experimentos, uno el control PD, otro el de asignación de polos y el LQR.

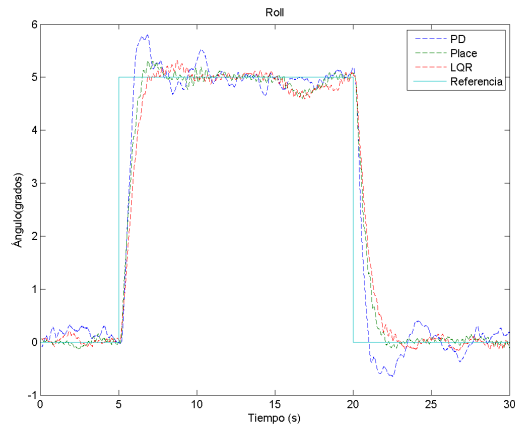


Figura 7.26: ADRC-roll. Experimento r tula

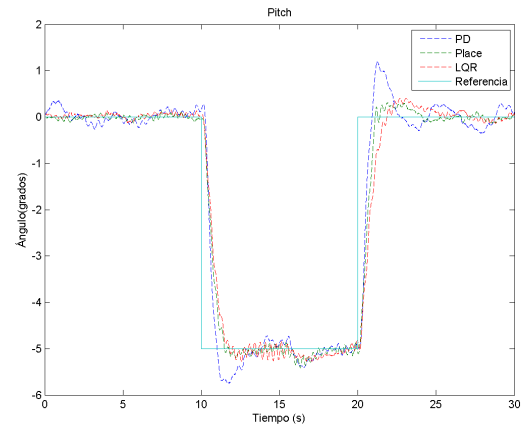


Figura 7.27: ADRC-pitch. Experimento r tula

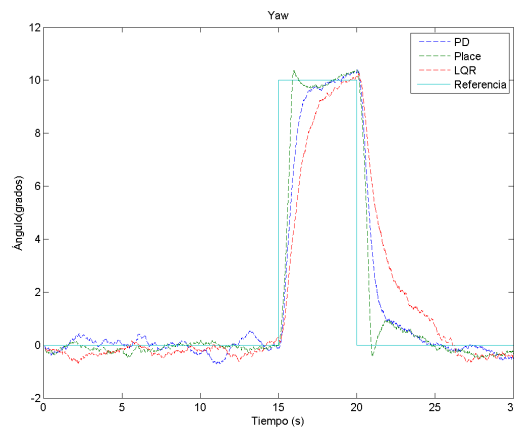


Figura 7.28: ADRC-yaw. Experimento r tula

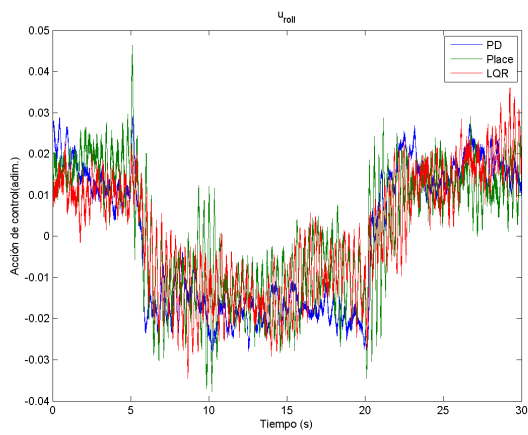


Figura 7.29: ADRC-ur. Experimento rótula

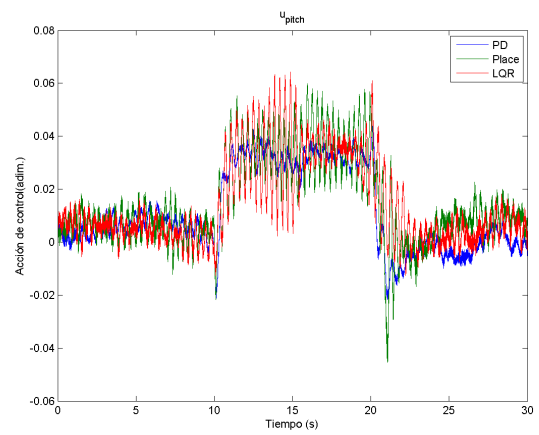


Figura 7.30: ADRC-up. Experimento rótula

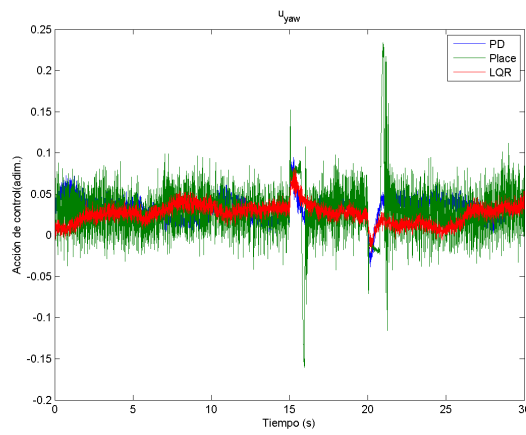


Figura 7.31: ADRC-uy. Experimento rótula

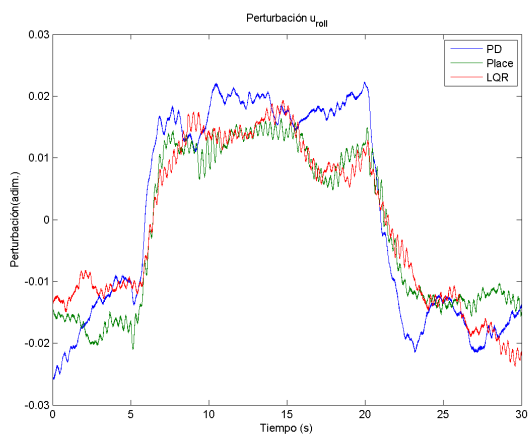


Figura 7.32: ADRC-Estimación perturbación ur. Experimento rótula

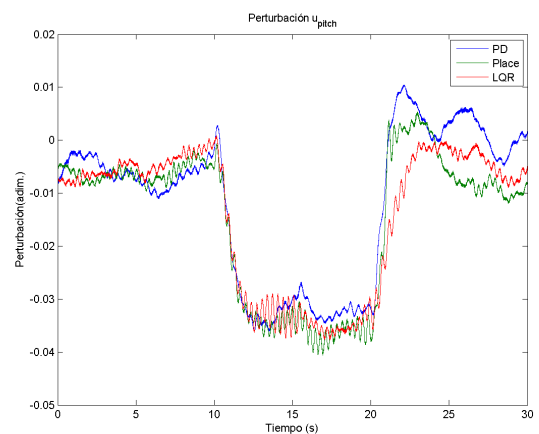


Figura 7.33: ADRC-Estimación perturbación up. Experimento rótula

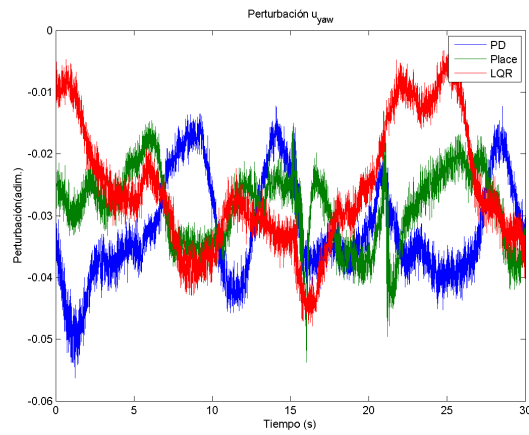


Figura 7.34: ADRC-Estimación perturbación uy. Experimento rótula

7.2.6 UDE

Control prealimentando la perturbación estimada mediante el UDE (subsección 6.6.2). Se han realizado tres experimentos, uno el control PD, otro el de asignación de polos y el LQR.

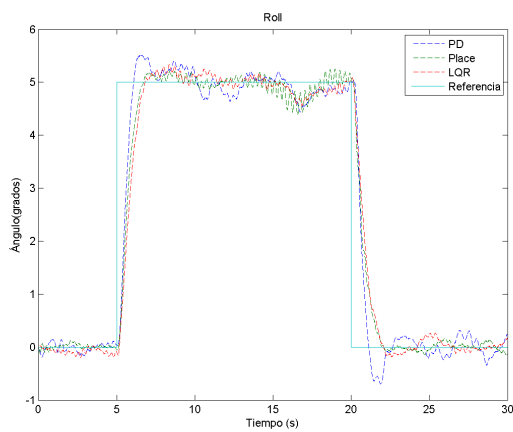


Figura 7.35: UDE-roll. Experimento rótula

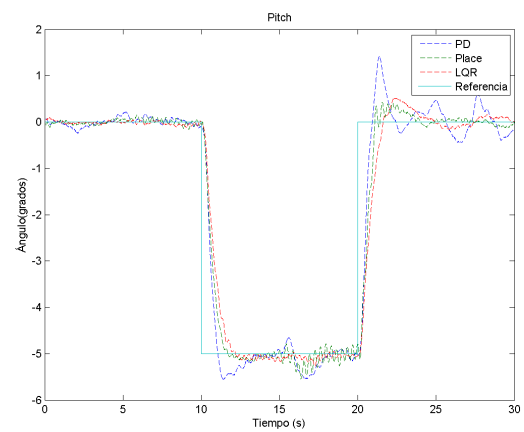


Figura 7.36: UDE-pitch. Experimento rótula

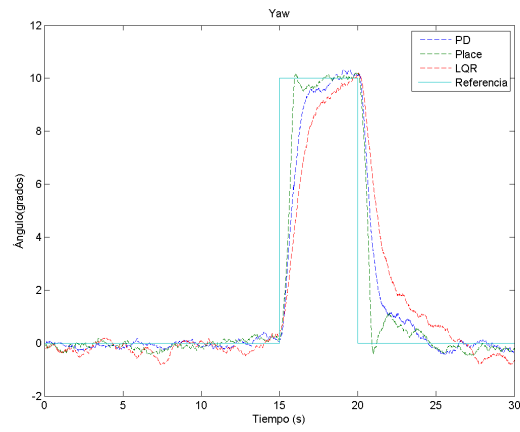


Figura 7.37: UDE-yaw. Experimento rótula

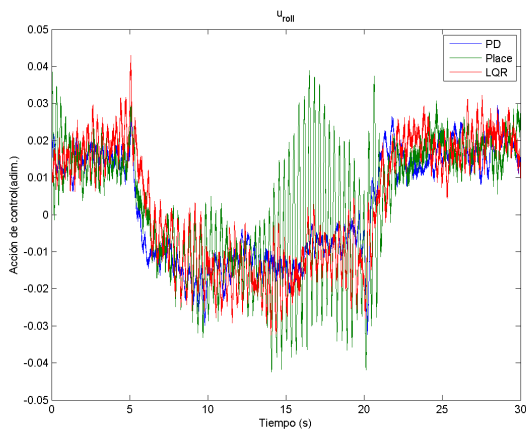


Figura 7.38: UDE-ur. Experimento rótula

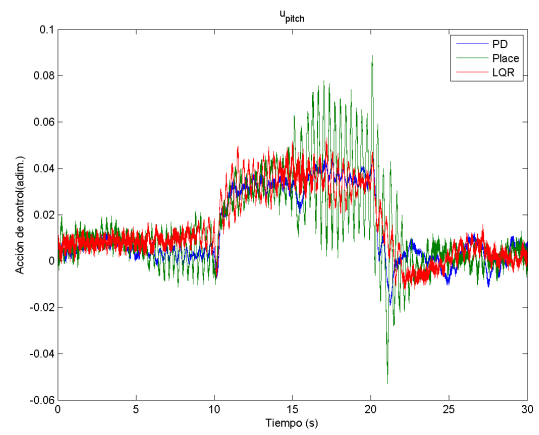


Figura 7.39: UDE-up. Experimento rótula

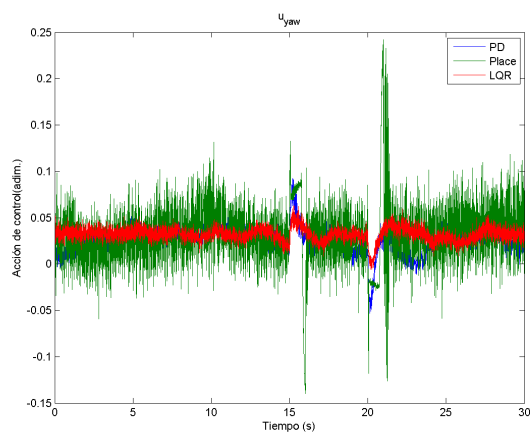


Figura 7.40: UDE-uy. Experimento rótula

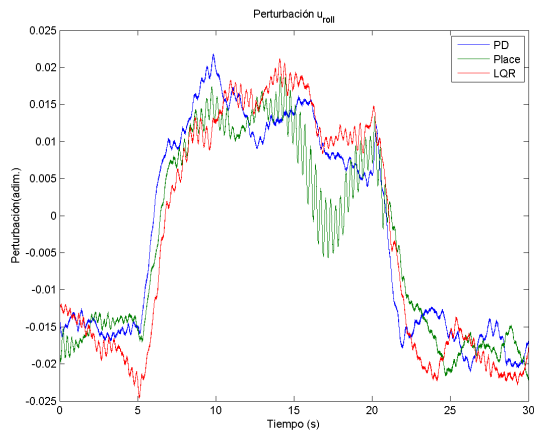


Figura 7.41: UDE-Estimación perturbación u_r . Experimento rótula

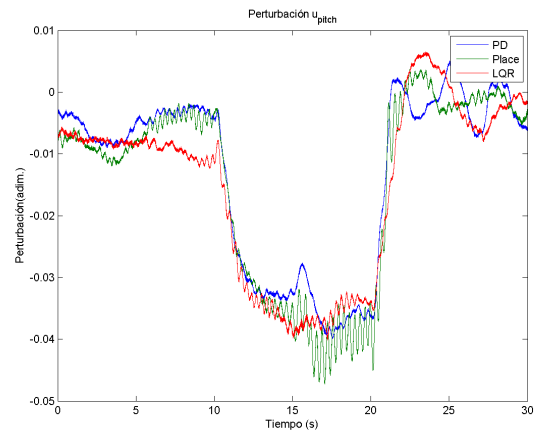


Figura 7.42: UDE-Estimación perturbación u_p . Experimento rótula

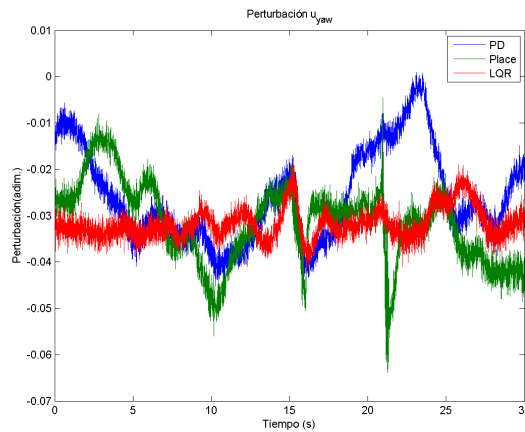


Figura 7.43: UDE-Estimación perturbación u_y . Experimento rótula

7.2.7 Análisis

En primer lugar se puede observar como los controladores PD y los dos de realimentación del estado no siguen la referencia, ya que el modelo real no es el doble integrador y además se ve sometido a diversas perturbaciones de diferentes índoles.

En el caso del PID, se observa que la pequeña acción integral (subsección 6.4.2) que tiene le permite seguimiento de la referencia, pero la dinámica presenta grandes oscilaciones y un tiempo de establecimiento muy mayor al requerido. Por el contrario, las técnicas de prealimentación de la perturbación permiten al sistema seguimiento de referencias sin que la dinámica del controlador original se vea relativamente afectada.

Los controladores PD y por realimentación del estado se adecua al comportamiento esperado de tiempo de establecimiento y oscilaciones de diseño y simulaciones. El controlador PD es el menos agresivo, mientras la realimentación del estado por asignación de polos es el más agresivo,

presentando oscilaciones cuando se ve sometido a perturbaciones considerables.

Entre el ADRC y el UDE, se observa que la estimación de la perturbación es más rápida en el caso del UDE, ya que en el ADRC se supone constante. No obstante, las dos técnicas permiten que el sistema real se comporte lo más parecido posible al modelo teórico considerado mediante la prealimentación de la perturbación estimada.

7.3 Comportamiento en vuelo libre

A partir del análisis del experimento de la rótula, para el vuelo libre se va a usar el controlador por realimentación del estado LQR, ya que el PD es poco agresivo y el de asignación de polos es demasiado. Por tanto, considerando el ajuste de la inercia (ver sección 4.8), se van a probar el controlador LQR, ADRC con LQR y UDE con LQR en vuelo libre.

Las referencias de posición (ver sección 6.7) se han establecido en un metro para la altura y cero para las restantes ($x = 0, y = 0, z = 1$) y se han aplicado manualmente diferentes perturbaciones (empujar y mover el quadrotor) para analizar su comportamiento. Para evaluar el control de orientación, el controlador de la posición x, y se ha decidido que sea poco agresivo. Cabe recordar que los controladores de x, y están conectados en cascada con el del roll y pitch.

7.3.1 Control LQR

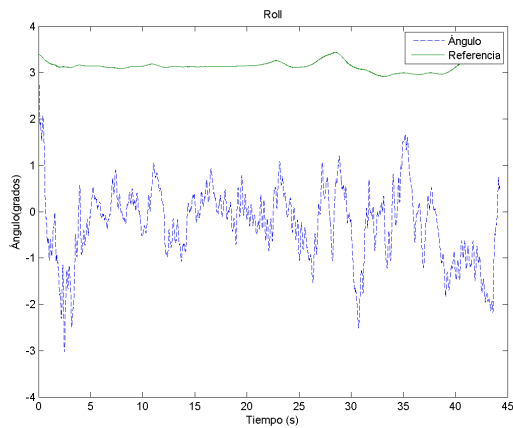


Figura 7.44: LQR-roll. Vuelo libre

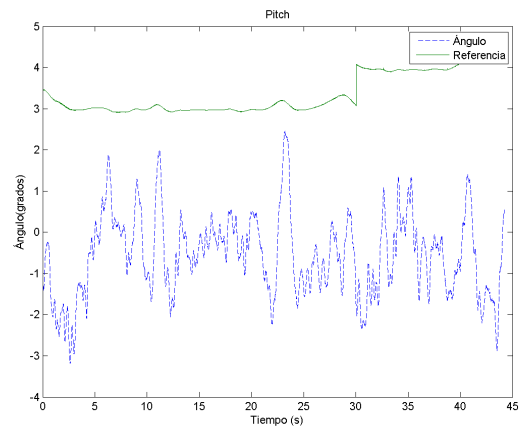


Figura 7.45: LQR-pitch. Vuelo libre

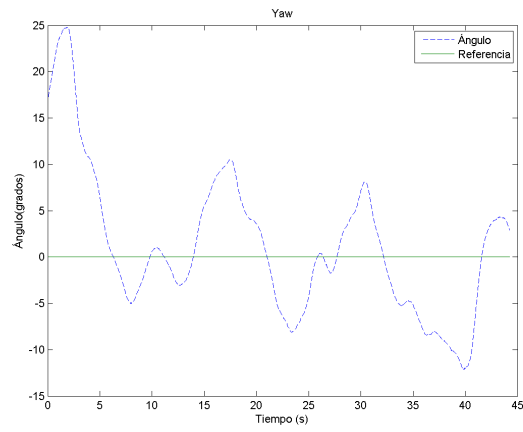


Figura 7.46: LQR-yaw. Vuelo libre

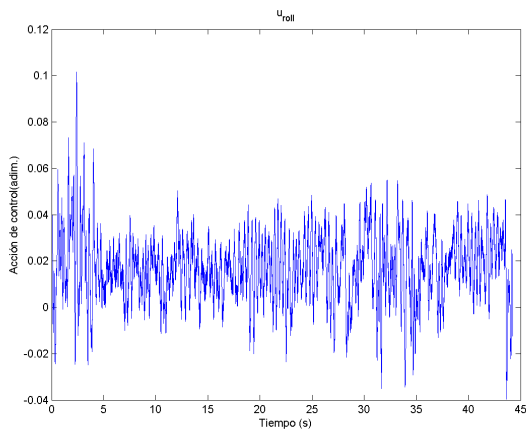


Figura 7.47: LQR-ur. Vuelo libre

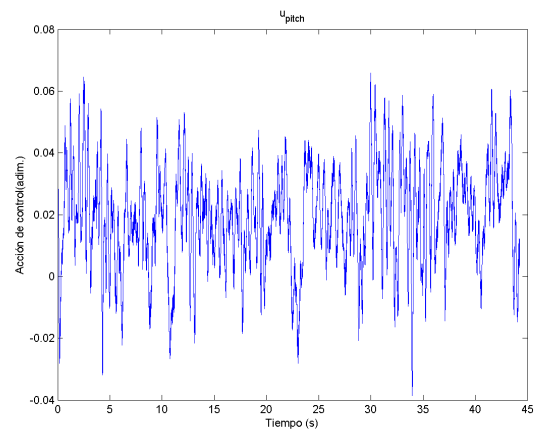


Figura 7.48: LQR-up. Vuelo libre

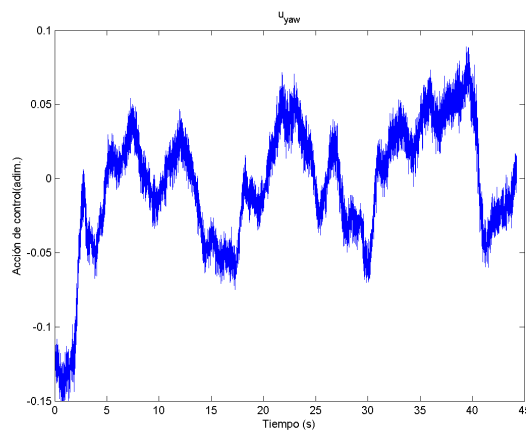


Figura 7.49: LQR-uy. Vuelo libre

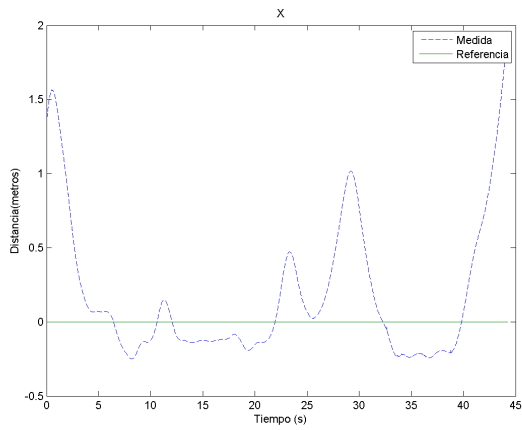


Figura 7.50: LQR X. Vuelo libre

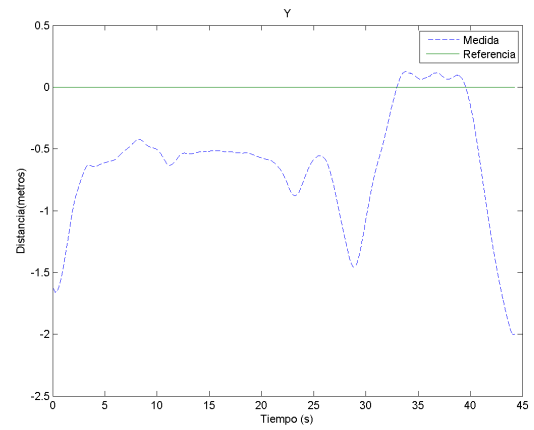


Figura 7.51: LQR Y. Vuelo libre

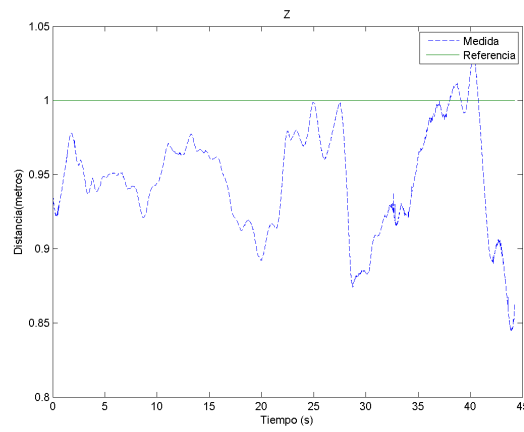


Figura 7.52: LQR Z. Vuelo libre

7.3.2 ADRC

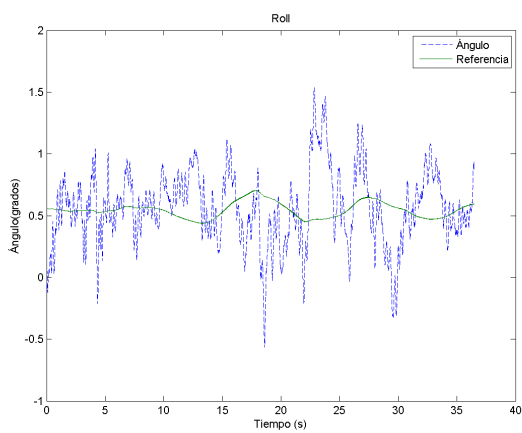


Figura 7.53: ADRC-roll. Vuelo libre

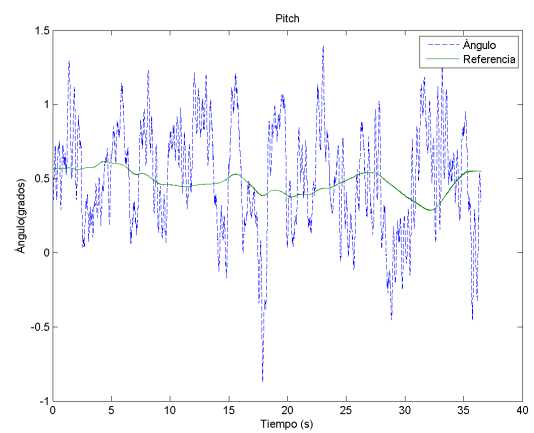


Figura 7.54: ADRC-pitch. Vuelo libre

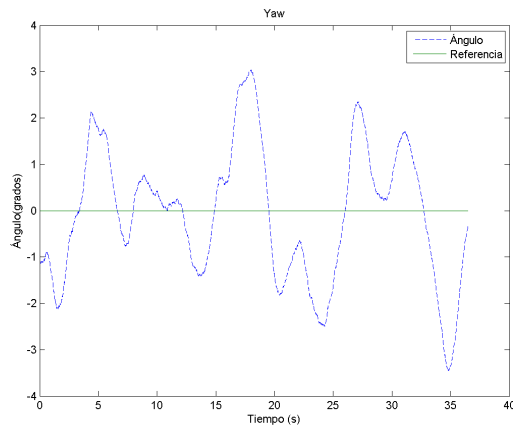


Figura 7.55: ADRC-yaw. Vuelo libre

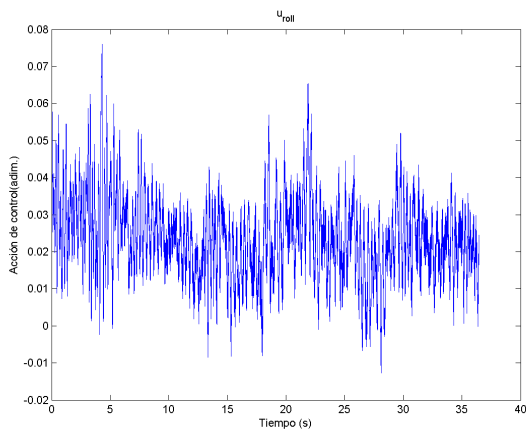


Figura 7.56: ADRC-ur. Vuelo libre

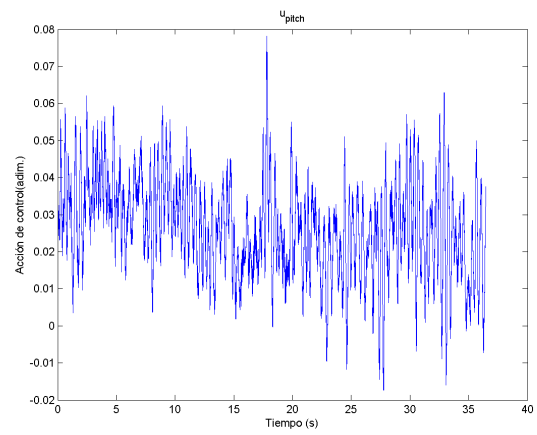


Figura 7.57: ADRC-up. Vuelo libre

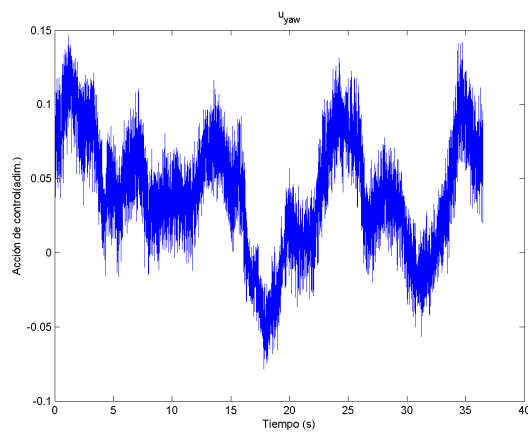


Figura 7.58: ADRC-uy. Vuelo libre

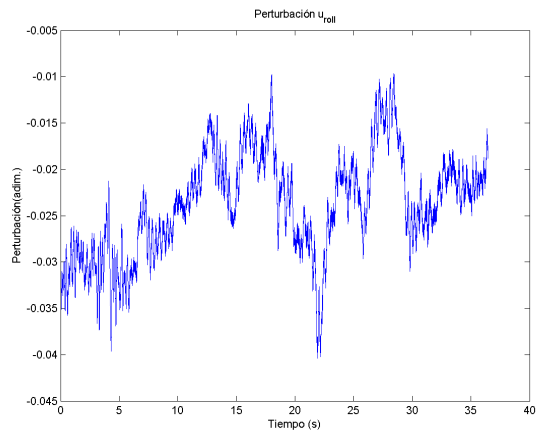


Figura 7.59: ADRC-Estimación perturbación u_r . Vuelo libre

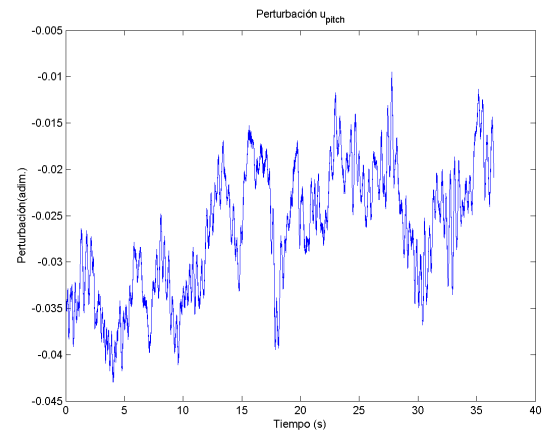


Figura 7.60: ADRC-Estimación perturbación u_p . Vuelo libre

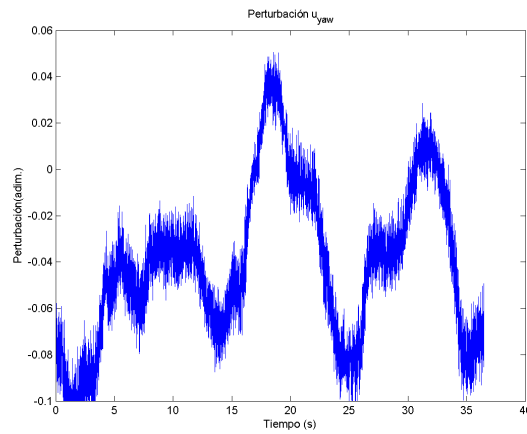


Figura 7.61: ADRC-Estimación perturbación u_y . Vuelo libre

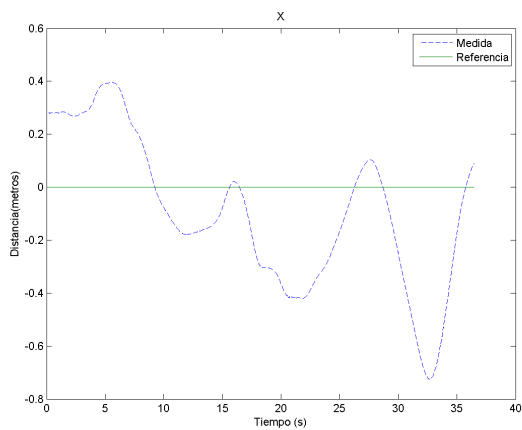


Figura 7.62: ADRC X. Vuelo libre

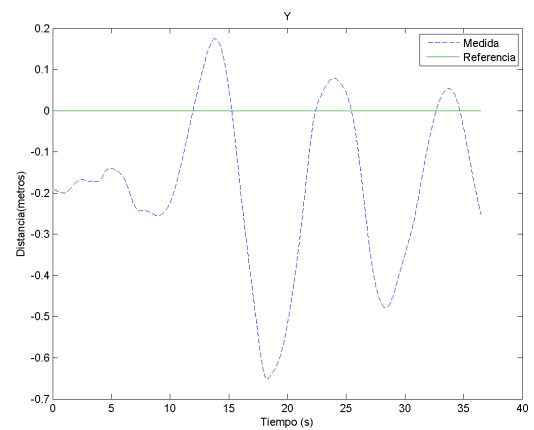


Figura 7.63: ADRC Y. Vuelo libre

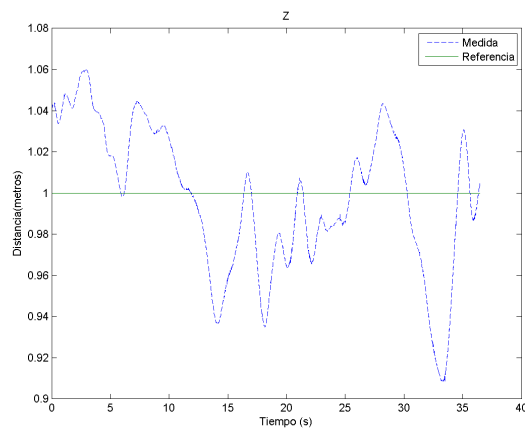


Figura 7.64: ADRC Z. Vuelo libre

7.3.3 UDE

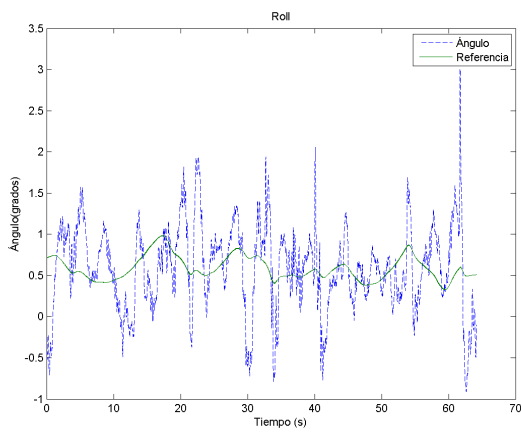


Figura 7.65: UDE-roll. Vuelo libre

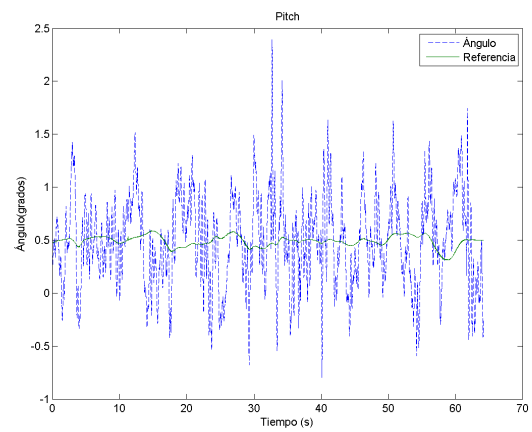


Figura 7.66: UDE-pitch. Vuelo libre

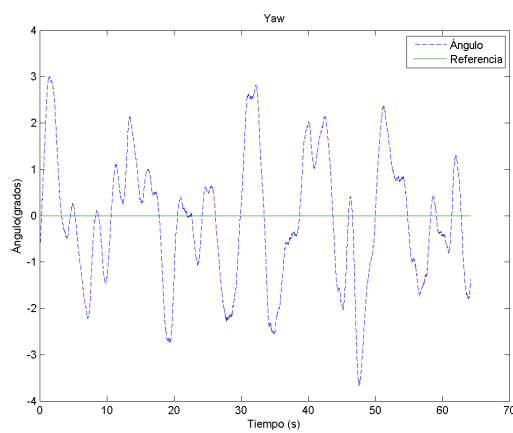


Figura 7.67: UDE-yaw. Vuelo libre

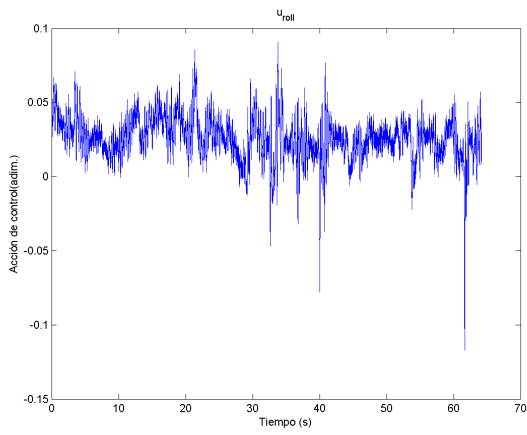


Figura 7.68: UDE-ur. Vuelo libre

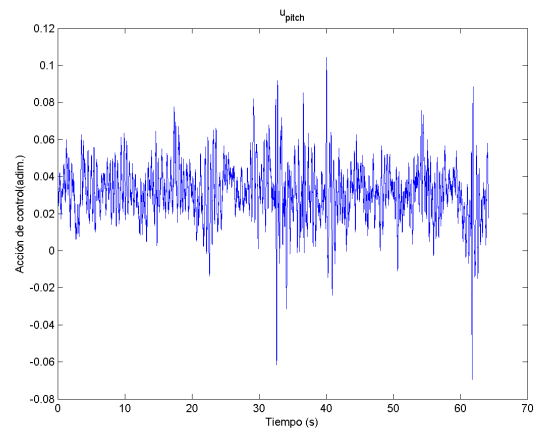


Figura 7.69: UDE-up. Vuelo libre

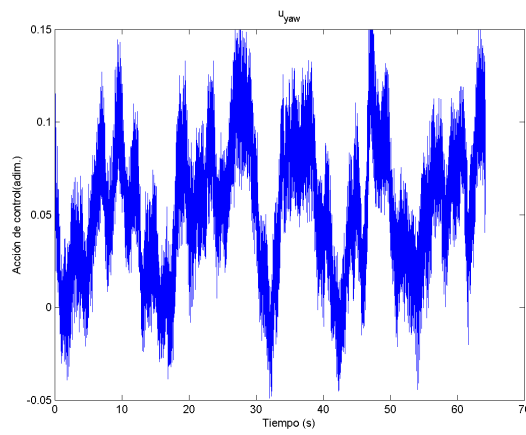


Figura 7.70: UDE-uy. Vuelo libre

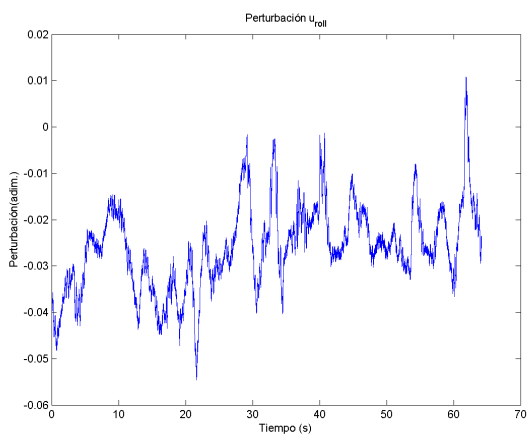


Figura 7.71: UDE-Estimación perturbación ur. Vuelo libre

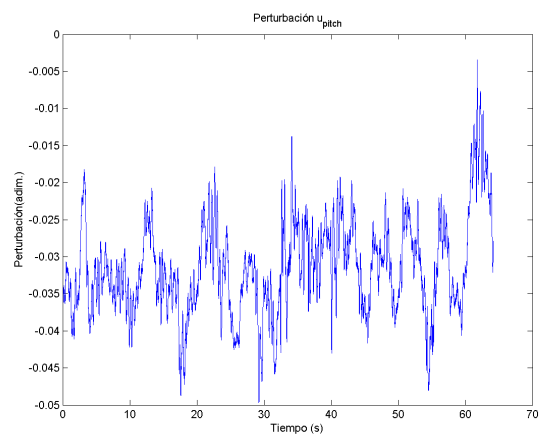


Figura 7.72: UDE-Estimación perturbación up. Vuelo libre

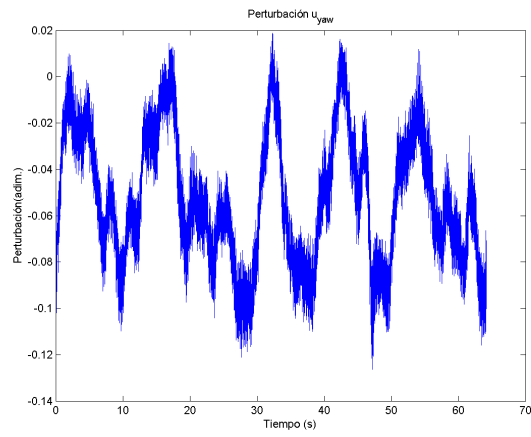


Figura 7.73: UDE-Estimación perturbación u_y . Vuelo libre

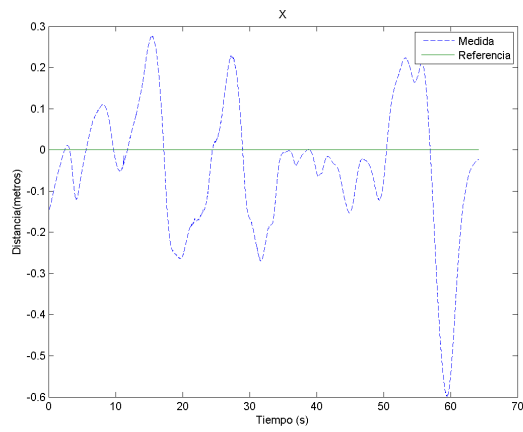


Figura 7.74: UDE X. Vuelo libre

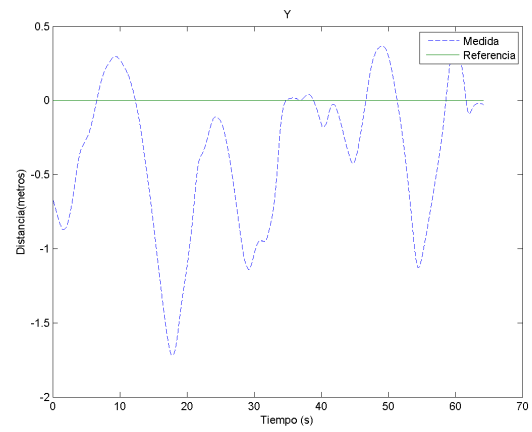


Figura 7.75: UDE Y. Vuelo libre

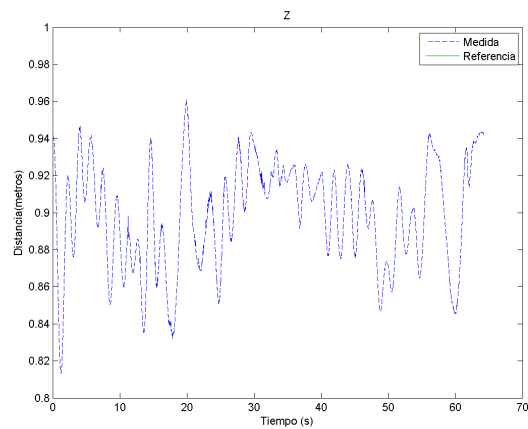


Figura 7.76: UDE Z. Vuelo libre

7.3.4 *Análisis*

Como puede observarse, las tres metodologías de control permiten controlar al sistema en vuelo libre, incluso con las fuertes perturbaciones a las que se ha visto sometido. Como en el caso de la rotula, el controlador LQR no es capaz de conseguir error de posición cero, pero mediante la realimentación de la perturbación estimada es posible compensarlo. El error de posición del controlador LQR junto con la realimentación del estado del control de posición, hacen que el error del seguimiento de las referencias de posición sea aún más elevado.

En lo referente al uso del ADRC y UDE, se extraen las mismas conclusiones que en la rotula. El UDE permite realizar una estimación más rápida de la perturbación al considerar que esta no es constante, pero en la práctica el control realizado no se ve demasiado afectado entre el uso de uno o del otro.

Finalmente, tanto el control de altura como el de x, y no siguen su referencia al no poseer acción integral o corrección del modelo. En el caso del control de altura se ha conseguido un desempeño adecuado. No obstante, para el caso del control x, y sería necesario en un trabajo futuro estudiar diferentes tipos de controladores para evaluar su desempeño y realizar posteriormente seguimiento de trayectorias.

Conclusiones y trabajos futuros

8.1 Conclusiones

A lo largo del trabajo de fin de máster realizado se ha desarrollado una plataforma para la validación segura de algoritmos de control en vehículos aéreos no tripulados tipo quadrotors. Se ha diseñado un software que permite su uso en cualquier tipo de vehículo de multirrotor y totalmente modulable para que sea más sencilla su modificación o la implementación de las estrategias de control.

Se ha programado una interfaz hombre maquina totalmente multiplataforma que permite la comunicación bidireccional con el quadrotor y enviar los datos del Optitrack al vehículo. Una HMI con multitud de opciones como la configuración de modos, envío de parámetros, envío de referencias mediante joystick o la posibilidad de enviar un patrón de referencias.

La estructura de la rotula ha sido definida y se ha obtenido el modelo teórico general, así como el modelo del quadrotor en vuelo libre. A partir de la estructura del modelo se ha podido construir un simulador y los parámetros físicos considerados de interés. También se ha analizado comparativamente el caso de la rótula y el vuelo libre, llegando a la conclusión que mediante multiplicando los parámetros de control por la relación de las inercias se pueden aplicar las conclusiones en el caso de la rótula al vuelo libre.

Se ha analizado la problemática de identificar en bucle cerrado, ya que al tratarse de un sistema inestable no es posible realizar experimentos sin un controlador. Se ha seleccionado el método de identificación X-CLOE para el modelo ARMAX supuesto y se ha obtenido el modelo experimental. A partir de dicho modelo, se han recalculado diferentes parámetros físicos difíciles de obtener por medida.

Con el modelo identificado, se han diseñado diferentes controladores. En un primer lugar se han simulado en Simulink mediante el mismo código a ejecutar en el quadrotor. En segundo

lugar se han probado en la r tula y en ultimo, los controladores seleccionados se han probado en vuelo libre. Se ha determinado que para realizar el control es necesario realimentar el  ngulo y su derivada. Entre los controladores, cabe destacar el buen desempe o que han tenido los basados en estimaci n de perturbaciones, los cuales permit an seguimiento de referencias con escaso efecto sobre la din mica deseada.

Como conclusi n, el presente trabajo de fin de m ster ha sido un proyecto multidisciplinar abarcando desde la programaci n de aplicaciones de escritorio o empotradas de tiempo real, hasta metodolog as de identificaci n y control de sistemas inestables.

8.2 Trabajos futuros

A continuaci n se proponen diversos posibles trabajos futuros a realizar partiendo de la plataforma desarrollada en el presente trabajo de fin de m ster.

- Cooperaci n entre quadrotors. El Optitrack permite identificar m ltiples cuerpos r gidos, por lo que se podr an crear aplicaciones donde los robots cooperen entre s  para un determinado fin.
- Mejora de la identificaci n. Se podr an emplear t cnicas m s avanzadas basadas en variables instrumentales, as  como obtener una cota del error realizado en la identificaci n.
- Mejora del control de posici n. Se podr an estudiar diferentes t cnicas de control de la posici n y realizar seguimientos de trayectorias en el espacio.
- Desarrollo de otras tipolog as de v hculos a reos no tripulados adecuados a la plataforma para realizar comparaciones entre los diferentes tipos de v hculos.
- Estudio de otras t cnicas de control. A parte de las utilizadas existen multitud de t cnicas a considerar para su control.
- Generar las se ales PWM por hardware en una placa adicional, para mejorar la precisi n de la se al actual generada por software.
- Implementaci n de un sensor GPS para permitir el control de posici n en exteriores.
- Comparativa entre las medidas obtenidas de la IMU y las que el Optitrack proporciona o la implementaci n de algoritmos de fusi n sensorial entre las dos medidas.
- Dise ar sistemas adicionales de seguridad, implementar redundancia en el control del quadrotor o sistemas de aterrizaje seguro.

Parte II

Presupuesto

Presupuesto

9.1 Introducción

El presente presupuesto forma parte del trabajo de fin de máster “Diseño e implementación de una plataforma para validación segura de algoritmos de control avanzados en vehículos aéreos no tripulados tipo quadrotors”.

El presupuesto se ha dividido en tres capítulos. El primero se refiere al conjunto de elementos físicos que forman la plataforma. El segundo capítulo corresponde con la programación de los elementos, limitándose a las partes que hacen posible su funcionamiento. El tercer y último capítulo comprende el desarrollo de la metodología de identificación y el diseño de los diferentes controladores empleados. A continuación se muestran los diferentes capítulos con sus respectivas unidades de obra.

1. Capítulo 1: Elementos físicos de la plataforma.

- Quadrotor.
- Rótula.
- Instalación Optitrack.

2. Capítulo 2: Software de la plataforma.

- Software del quadrotor.
- Interfaz hombre maquina.

3. Capítulo 3: Identificación y control.

- Diseño del Simulador.
- Identificación del sistema.

- Diseño y validación de controladores.

9.2 Cuadro de precios

Para el cálculo del precio por hora de la mano de obra, se ha utilizado como referencia un precio base mensual de 2500€. En el caso del trabajo de ingeniería, considerando que el trabajador trabaja por cuenta ajena mediante las estimaciones de [59] se ha obtenido el precio hora aproximado de dicho trabajador. Para el trabajo realizado por el técnico, se ha calculado de manera similar considerando un sueldo base de 2000€ mensuales.

Las estimaciones de tiempo se han realizado considerando una dedicación de media jornada a partir de los días dedicados a cada tarea.

Tabla 9.1: Cuadro de precios de la mano de obra

Código	Unidades	Descripción	Precio (€)
MO.MII	h	Ingeniero industrial	37,00
MO.TEC	h	Técnico	28,00

El quadrotor y la rótula base utilizada se suponen construidas en los trabajos previos, obteniendo su precio de [26] y [27] respectivamente. No obstante, ha sido necesario realizar diferentes reajustes y en el caso de la rótula, sustituirla por otra de menor fricción. Por tanto se añadirán los componentes extras y las horas necesarias para su sustitución y ajuste. Con el Optitrack se han añadido los componentes de la instalación con el precio de su puesta en marcha. Se han considerado independientemente la licencia del software Motive de Optitrack y la licencia de Matlab con acceso a Simulink. Se supone que todos los elementos del trabajo se utilizan únicamente para el mismo, asumiendo el precio completo de los mismos.

Tabla 9.2: Cuadro de precios de los materiales

Código	Unidades	Descripción	Precio (€)
MA.QUAD	u	Quadrotor base	4808,61
MA.ROT	u	Rótula base	1380,47
MA.EROT	u	Elemento rótula fotográfica	14,99
MA.CAM	u	Cámaras Optitrack	600,00
MA.OPT	u	Unidad de procesamiento Optitrack	1100,00
MA.CABL	u	Conexión del sistema Optitrack	50,00
MA.LOPT	u	Licencia uso Motive	800,00
MA.LMAT	u	Licencia Matlab	2000,00

Para la realización del conjunto de tareas de software, ha sido necesaria la utilización de un ordenador portátil. Se supone que el ordenador se dedica exclusivamente al presente trabajo, por tanto para un precio base de 700€, se obtiene el precio hora a partir de todas sus horas de uso.

Tabla 9.3: Cuadro de precios de la maquinaria

Código	Unidades	Descripción	Precio (€)
M.ORD	h	Ordenador portátil	2,50

A partir de los presupuestos de las unidades de obra, se considerarán un 1% de costes directos complementarios y un 2% de costes indirectos. Para la obtención del presupuesto de ejecución por contrata se ha utilizado un 15% de gastos generales y un 6% de beneficio industrial. Finalmente, se ha aplicado el IVA actual del 21%. Todos estos valores son los valores medios recomendados para este tipo de proyectos.

9.3 Cuadro de precios unitarios

En la presente sección se indican los precios de las diferentes unidades de obra que forman los capítulos.

Tabla 9.4: Capítulo 1

Código	Unidad de obra	Precio(€)
C1.UO1	Quadrotor	5202,11
C1.UO2	Rótula	1670,57
C1.UO3	Instalación Optitrack	7474,10

Tabla 9.5: Capítulo 2

Código	Unidad de obra	Precio(€)
C2.UO1	Software del quadrotor	3109,66
C2.UO2	Interfaz hombre máquina	2092,34

Tabla 9.6: Capítulo 3

Código	Unidad de obra	Precio(€)
C3.UO1	Diseño del simulador	3077,72
C3.UO2	Identificación del sistema	3167,35
C3.UO3	Diseño y validación de controladores	2150,03

9.4 Cuadro de precios descompuestos

En la presente sección se indican la descomposición de las diferentes unidades de obra.

Tabla 9.7: C1.UO1. Quadrotor

Código	Unidades	Descripción	Precio (€)	Rendimiento	Importe (€)
MA.QUAD	u	Quadrotor base	4808,61	1,00	4808,61
MO.MII	h	Ingeniero industrial	37,00	5,00	185,00
MO.TEC	h	Técnico	28,00	2,00	56,00
					5049,61
Costes directos complementarios (1 %)					50,50
Costes indirectos (2 %)					102,00
Total unidad de obra					5202,11

Tabla 9.8: C1.UO2. Rótula

Código	Unidades	Descripción	Precio (€)	Rendimiento	Importe (€)
MA.ROT	u	Rótula base	1380,47	1,00	1380,47
MA.EROT	u	Elemento rótula fotográfica	14,99	5,00	74,95
MO.MII	h	Ingeniero industrial	37,00	3,00	111,00
MO.TEC	h	Técnico	28,00	1,00	28,00
					1594,42
Costes directos complementarios (1 %)					15,94
Costes indirectos (2 %)					32,21
Total unidad de obra					1670,57

Tabla 9.9: C1.UO3. Instalación Optitrack

Código	Unidades	Descripción	Precio (€)	Rendimiento	Importe (€)
MA.CAM	u	Cámaras Optitrack	600,00	6,00	3600,00
MA.OPT	u	Un. Proc. Optitrack	1100,00	2,00	2200,00
MA.CABL	u	Con. Optitrack	50,00	1,00	50,00
MA.LOPT	u	Licencia uso Motive	800,00	1,00	800,00
MO.MII	h	Ingeniero industrial	37,00	5,00	185,00
MO.TEC	h	Técnico	28,00	15,00	420,00
					7255,00
Costes directos complementarios (1 %)					72,55
Costes indirectos (2 %)					146,55
Total unidad de obra					7474,10

Tabla 9.10: C2.UO1. Software del quadrotor

Código	Unidades	Descripción	Precio (€)	Rendimiento	Importe (€)
MO.MII	h	Ingeniero industrial	37,00	75,00	2775,00
M.ORD	h	Ordenador portátil	2,5	75,00	187,50
MO.TEC	h	Técnico	28,00	2,00	56,00
					3018,50
Costes directos complementarios (1%)					30,19
Costes indirectos (2%)					60,97
Total unidad de obra					3109,66

Tabla 9.11: C2.UO2. Interfaz hombre máquina

Código	Unidades	Descripción	Precio (€)	Rendimiento	Importe (€)
MO.MII	h	Ingeniero industrial	37,00	50,00	1850,00
M.ORD	h	Ordenador portátil	2,5	50,00	125,00
MO.TEC	h	Técnico	28,00	2,00	56,00
					2031,00
Costes directos complementarios (1%)					20,31
Costes indirectos (2%)					41,03
Total unidad de obra					2092,34

Tabla 9.12: C3.UO1. Diseño del simulador

Código	Unidades	Descripción	Precio (€)	Rendimiento	Importe (€)
MO.MII	h	Ingeniero industrial	37,00	25,00	925,00
M.ORD	h	Ordenador portátil	2,5	25,00	62,50
MA.LMAT	u	Licencia Matlab	2000,00	1,00	2000,00
					2987,50
Costes directos complementarios (1%)					29,88
Costes indirectos (2%)					60,35
Total unidad de obra					3077,72

Tabla 9.13: C3.UO2. Identificación del sistema

Código	Unidades	Descripción	Precio (€)	Rendimiento	Importe (€)
MO.MII	h	Ingeniero industrial	37,00	75,00	2775,00
M.ORD	h	Ordenador portátil	2,5	75,00	187,50
MO.TEC	h	Técnico	28,00	4,00	112,00
					3074,50
Costes directos complementarios (1 %)					30,75
Costes indirectos (2 %)					62,10
Total unidad de obra					3167,35

Tabla 9.14: C3.UO3. Diseño y validación de controladores

Código	Unidades	Descripción	Precio (€)	Rendimiento	Importe (€)
MO.MII	h	Ingeniero industrial	37,00	50,00	1850,00
M.ORD	h	Ordenador portátil	2,50	50,00	125,00
MO.TEC	h	Técnico	28,00	4,00	112,00
					2087,00
Costes directos complementarios (1 %)					20,87
Costes indirectos (2 %)					42,16
Total unidad de obra					2150,03

9.5 Presupuesto parcial

En la presente sección se indican el presupuesto parcial de los capítulos del presupuesto.

Tabla 9.15: Capítulo 1

Código	Unidad de obra	Precio(€)	Cantidad	Coste total (€)
C1.UO1	Quadrotor	5202,11	1,00	5202,11
C1.UO2	Rótula	1670,57	1,00	1670,57
C1.UO3	Instalación Optitrack	7474,10	1,00	7474,10
				14346,78

Tabla 9.16: Capítulo 2

Código	Unidad de obra	Precio(€)	Cantidad	Coste total (€)
C2.UO1	Software del quadrotor	3109,66	1,00	3109,66
C2.UO2	Interfaz hombre máquina	2092,34	1,00	2092,34
				5201,99

Tabla 9.17: Capítulo 3

Código	Unidad de obra	Precio(€)	Cantidad	Coste total (€)
C3.UO1	Diseño del simulador	3077,72	1,00	3077,72
C3.UO2	Identificación del sistema	3167,35	1,00	3167,35
C3.UO3	Diseño y validación de controladores	2150,03	1,00	2150,03
				8395,10

9.6 Presupuesto de ejecución, por contrata y base de licitación

En la presente sección se indican el presupuesto de ejecución material, de ejecución por contrata y base de licitación.

Tabla 9.18: Presupuesto por capítulos

Capítulo	Precio (€)
Capítulo 1	14346,78
Capítulo 2	5201,99
Capítulo 3	8395,10
	27943,88

Tabla 9.19: Presupuesto de ejecución, por contrata y base de licitación

PRESUPUESTO DE EJECUCIÓN MATERIAL	27943,88€
Gastos generales (15 %)	4191,58
Beneficio industrial (6 %)	1676,63
PRESUPUESTO DE EJECUCIÓN POR CONTRATA	33812,09€
IVA(21 %)	7100,54
PRESUPUESTO BASE DE LICITACIÓN	40912,63€

Asciende el presupuesto total de ejecución material a la siguiente cantidad:

VEINTISIETE MIL NOVECIENTOS CUARENTA Y TRES EUROS CON OCHENTA Y OCHO CÉNTIMOS

Asciende el presupuesto total de ejecución por contrata a la siguiente cantidad:

TREINTA Y TRES MIL OCHOCIENTOS DOCE EUROS CON NUEVE CÉNTIMOS

Asciende el presupuesto base de licitación a la siguiente cantidad:

CUARENTA MIL NOVECIENTOS DOCE EUROS CON SESENTA Y TRES CÉNTIMOS

Bibliografía

- [1] Recurso Web. (2017). Asociación Española de RPAS, dirección: <http://www.aerpas.es/>.
- [2] Recurso Web. (2017). Programa Marco de Investigación e Innovación de la Unión Europea, dirección: <http://www.eshorizonte2020.es/>.
- [3] I. García, “Estudio sobre vehículos aéreos no tripulados y sus aplicaciones”, *Universidad de Valladolid*, 2017.
- [4] P. Fahlstrom y T. Gleason, *Introduction to UAV systems*. John Wiley & Sons, 2012.
- [5] P. Castillo, R. Lozano y A. E. Dzul, *Modelling and control of mini-flying machines*. Physica-Verlag, 2005.
- [6] Recurso Web. (2013). Drone Origins: World War II and Vietnam-era Remotely Piloted Vehicles, dirección: <https://understandingempire.wordpress.com/2013/12/02/drone-origins-world-war-ii-and-vietnam-era-remotely-piloted-vehicles/>.
- [7] R. A. Carver, “Wernher von Braun and the German rocket team in America”, Tesis de maestría., East Tennessee State University, 2001.
- [8] J. G. Leishman, “The breguet-richet quad-rotor helicopter of 1907”, *Vertiflite*, vol. 47, n.º 3, págs. 58-60, 2002.
- [9] C. Gablehouse, *Helicopters and autogiros: a history of rotating-wing and V/STOL aviation*. Lippincott, 1969.
- [10] P. McKerrow, “Modelling the Draganflyer four-rotor helicopter”, en *Robotics and Automation, 2004. Proceedings. ICRA'04. 2004 IEEE International Conference on*, IEEE, vol. 4, 2004, págs. 3596-3601.
- [11] N. Michael, D. Mellinger, Q. Lindsey y V. Kumar, “The grasp multiple micro-uav testbed”, *IEEE Robotics & Automation Magazine*, vol. 17, n.º 3, págs. 56-65, 2010.

- [12] Recurso Web. (2017). Driverless flying taxi service set to launch in Dubai, dirección: <http://edition.cnn.com/2017/03/17/tech/ehang-passenger-drone-dubai/index.html>.
- [13] P. Castillo, A. Dzul y R. Lozano, “Real-time stabilization and tracking of a four rotor mini-rotorcraft”, en *European Control Conference (ECC), 2003*, IEEE, 2003, págs. 3123-3128.
- [14] P. Pounds, R. Mahony, P. Hynes y J. M. Roberts, “Design of a four-rotor aerial robot”, en *Proceedings of the 2002 Australasian Conference on Robotics and Automation (ACRA 2002)*, Australian Robotics & Automation Association, 2002, págs. 145-150.
- [15] Z. Chen y J. Huang, “Attitude tracking and disturbance rejection of rigid spacecraft by adaptive control”, *IEEE Transactions on Automatic Control*, vol. 54, n.º 3, págs. 600-605, 2009.
- [16] T. Lee, M. Leok y N. H. McClamroch, “Nonlinear robust tracking control of a quadrotor UAV on SE (3)”, *Asian Journal of Control*, vol. 15, n.º 2, págs. 391-408, 2013.
- [17] G. M. Hoffmann, S. L. Waslander y C. J. Tomlin, “Quadrotor helicopter trajectory tracking control”, en *AIAA guidance, navigation and control conference and exhibit*, 2008, págs. 1-14.
- [18] A Castillo, “Desarrollo integral de un quadrotor: diseño de un algoritmo de control para la posición x-y basado en señales GPS”, *Universitat Politècnica de València. TFG*, 2014.
- [19] T. Bresciani, “Modelling, identification and control of a quadrotor helicopter”, *MSc Theses*, 2008.
- [20] D Schermuk, “Diseño e implementación de un controlador para la orientación de un quadrotor”, *Facultad de Ingeniería de la Universidad de Buenos Aires. TFC*, 2012.
- [21] I. Stanculeanu y T. Borangiu, “Quadrotor black-box system identification”, *World Academy of Science, Engineering and Technology*, vol. 5, págs. 276-279, 2011.
- [22] Boletín Oficial del Estado. (2014). Real Decreto 552/2014, dirección: <https://www.boe.es/boe/dias/2014/07/01/pdfs/BOE-A-2014-6856.pdf>.
- [23] Recurso Web. (2016). Nueva legislación de drones en España para 2017, dirección: <http://www.dronair.es/actualizacion-de-la-ley-sobre-el-uso-de-drones-en-espana>.
- [24] L Ródenas, “Plataforma de desarrollo para el control de estabilidad en tiempo real de un vehículo aéreo tipo quadrotor”, *Universitat Politècnica de València. TFC*, 2013.
- [25] V Balaguer, “Sistema de Tiempo Real basado en Raspberry PI para el control de un Quadrotor”, *Universitat Politècnica de València. TFG*, 2015, <http://hdl.handle.net/10251/53624>.

- [26] F. Casas, “Desarrollo de un software de control basado en Raspberry Pi II para la implementación de algoritmos de control en una plataforma 3D (con 3 grados de libertad).”, *Universitat Politècnica de València. TFG*, 2016, <http://hdl.handle.net/10251/65984>.
- [27] J. Reina, “Desarrollo de una plataforma de control basada en raspberry pi para la implementación y ensayo del control de orientación mediante soporte de pruebas indoor”, *Universitat Politècnica de València. TFG*, 2016, <http://hdl.handle.net/10251/70071>.
- [28] R. Machado, “Diseño de una plataforma mecánica para el control de orientación 3D de un sistema Quadrotor basado en Raspberry Pi 2.”, *Universitat Politècnica de València. TFG*, 2016, <http://hdl.handle.net/10251/65985>.
- [29] R. R.P. M. J. Rico-Azagra M. Gil-Martinez, “Plataforma didáctica de bajo coste para el control de actitud y orientación de UAVs multirrotor”, *Jornadas de Automática 2016*, págs. 1140-1147, 2016.
- [30] P Albiol, “Desarrollo integral de un quadrotor: diseño e implementación de los sistemas de telemetría e interfaz gráfica (HMI) para la monitorización y control mediante entorno QT”, *Universitat Politècnica de València. TFG*, 2014.
- [31] N Diaz-Otero, “Desarrollo de un sistema de telemetría e interfaz gráfica para el control de un vehículo aéreo pilotado remotamente (RPAS)”, *Universitat Politècnica de València. TFG*, 2015, <http://hdl.handle.net/10251/56050>.
- [32] R. Sanz, L. Ródenas, P. García y P. Albertos, “Attitude Estimation using Low-Cost Sensors: a comparative analysis”, en *Intelligent Control and Automation (WCICA), 2014 11th World Congress on*, IEEE, 2014, págs. 4261-4266.
- [33] R. Sanz, L. Rodenas, P. Garcia y P. Castillo, “Improving attitude estimation using inertial sensors for quadrotor control systems”, en *Unmanned Aircraft Systems (ICUAS), 2014 International Conference on*, IEEE, 2014, págs. 895-901.
- [34] Recurso Web. (2017). MAVLink Micro Air Vehicle Communication Protocol, dirección: <http://qgroundcontrol.org/mavlink/start>.
- [35] Recurso Web. (2017). The pigpio library, dirección: <http://abyz.co.uk/rpi/pigpio/>.
- [36] Recurso Web. (2017). Optitrack - Motion Capture System, dirección: <http://optitrack.com/>.
- [37] P. Albertos y S. Antonio, *Multivariable control systems: an engineering approach*. Springer Science & Business Media, 2006.
- [38] K. Ogata, *Ingeniería de control moderna*. Pearson Educación, 2003.

- [39] I. D. Landau y G. Zito, *Digital control systems: design, identification and implementation*. Springer Science & Business Media, 2005.
- [40] L. Sun, H. Ohmori y A. Sano, “Direct closed-loop identification approach to unstable plant”, en *Decision and Control, 2000. Proceedings of the 39th IEEE Conference on*, IEEE, vol. 2, 2000, págs. 1148-1153.
- [41] X. Zhang, X. Li, K. Wang e Y. Lu, “A survey of modelling and identification of quadrotor robot”, en *Abstract and Applied Analysis*, Hindawi Publishing Corporation, vol. 2014, 2014.
- [42] L. Sun, Y. Miyake, H. Ohmori y A. Sano, “New Direct Closed-loop Identification Method for Unstable Systems and Its Applications to Magnetic Suspension System”, *Transactions of the Society of Instrument and Control Engineers*, vol. 36, n.º 4, págs. 348-355, 2000.
- [43] K. J. Åström y B. Wittenmark, *Computer-controlled systems: theory and design*. Prentice-Hall, 1997.
- [44] Recurso Web. (1997). Generation of pseudorandom numbers by a shift register, dirección: http://www.cs.miami.edu/home/burt/learning/Csc609.022/random_numbers.html.
- [45] J. Åkesson, “Safe manual control of unstable systems”, *MSc Theses*, 2000.
- [46] R. Miranda Colorado y G. Contreras Castro, “Closed-loop identification applied to a DC servomechanism: controller gains analysis”, *Mathematical Problems in Engineering*, vol. 2013, 2013.
- [47] Q.-G. Wang, H.-Q. Zhou, Y. Zhang e Y. Zhang, “A comparative study on control of unstable processes with time delay”, en *Control Conference, 2004. 5th Asian*, IEEE, vol. 3, 2004, págs. 1996-2004.
- [48] A. S. Rao y M Chidambaram, “Enhanced two-degrees-of-freedom control strategy for second-order unstable processes with time delay”, *Industrial & engineering chemistry research*, vol. 45, n.º 10, págs. 3604-3614, 2006.
- [49] R. P. Sree, M. Srinivas y M Chidambaram, “A simple method of tuning PID controllers for stable and unstable FOPTD systems”, *Computers & chemical engineering*, vol. 28, n.º 11, págs. 2201-2218, 2004.
- [50] A. O’Dwyer, *Handbook of PI and PID controller tuning rules*. World Scientific, 2009.
- [51] S. Skogestad, “Simple analytic rules for model reduction and PID controller tuning”, *Journal of process control*, vol. 13, n.º 4, págs. 291-309, 2003.
- [52] E. Bernabeu y M. Martínez, “Diseño algebraico de controladores discretos”, *SP Universidad Politécnica de Valencia*, 1999.

- [53] H Shibasaki, J Endo, Y Hikichi, R Tanaka, K Kawaguchi y Y Ishida, “A modified internal model control for an unstable plant with an integrator in continuous-time system”, *International Journal of Information and Electronics Engineering*, vol. 3, n.º 4, pág. 357, 2013.
- [54] H. Wang, H. Dong, L. He, Y. Shi e Y. Zhang, “Design and simulation of LQR controller with the linear inverted pendulum”, en *Electrical and Control Engineering (ICECE), 2010 International Conference on*, IEEE, 2010, págs. 699-702.
- [55] E. V. Kumar y J. Jerome, “Robust LQR controller design for stabilizing and trajectory tracking of inverted pendulum”, *Procedia Engineering*, vol. 64, págs. 169-178, 2013.
- [56] P. Albertos, R. Sanz y P. Garcia, “Disturbance rejection: A central issue in process control”, en *Systems and Control (ICSC), 2015 4th International Conference on*, IEEE, 2015, págs. 1-8.
- [57] R. Sanz, P. Garcia y P. Albertos, “Active disturbance rejection by state feedback: Experimental validation in a 3-DOF quadrotor platform”, en *Society of Instrument and Control Engineers of Japan (SICE), 2015 54th Annual Conference of the*, IEEE, 2015, págs. 794-799.
- [58] R. Sanz, P. Garcia, Q.-C. Zhong y P. Albertos, “Robust control of quadrotors based on an uncertainty and disturbance estimator”, *Journal of Dynamic Systems, Measurement, and Control*, vol. 138, n.º 7, pág. 071006, 2016.
- [59] Recurso Web. (2017). Estimación de salario freelance, dirección: <http://www.calculadorafreelance.com/>.