



**UNIVERSIDAD EUROPEA DE MADRID**

**ESCUELA DE ARQUITECTURA, INGENIERÍA Y DISEÑO**

**Máster Universitario en Ingeniería Aeronáutica**

**Avances en el Desarrollo de un  
Simulador de RPAS Integrando Vehículo  
Hexacóptero y Octocóptero**

**Alejandro Fernández Guerrero**

**Curso 2022-2023**

Título: Avances en el Desarrollo de un Simulador de RPAS Integrando  
Vehículo Hexacóptero y Octocóptero.  
Autor: Alejandro Fernández Guerrero.



Título: Avances en el Desarrollo de un Simulador de RPAS Integrando Vehículo Hexacóptero y Octocóptero.  
Autor: Alejandro Fernández Guerrero.



**Título:** Avances en el Desarrollo de un Simulador de RPAS Integrando Vehículo Hexacóptero y Octocóptero

**Autor:** Alejandro Fernández Guerrero

**Tutor:** Federico Martínez de la Escalera

**Titulación:** Máster Universitario en Ingeniería Aeronáutica

**Curso:** 2022-2023

Título: Avances en el Desarrollo de un Simulador de RPAS Integrando  
Vehículo Hexacóptero y Octocóptero.  
Autor: Alejandro Fernández Guerrero.







# Índice de contenido

|                                                                     |     |
|---------------------------------------------------------------------|-----|
| Resumen.....                                                        | 8   |
| Chapter 1. Introducción .....                                       | 9   |
| 1.1 Objetivos .....                                                 | 9   |
| 1.2 Estado del arte .....                                           | 10  |
| 1.3 Descripción del contenido.....                                  | 15  |
| Chapter 2. Descripción del modelo.....                              | 17  |
| 2.1 Sistemas de referencia .....                                    | 17  |
| 2.2 Vector estado .....                                             | 21  |
| 2.3 Fuerzas y momentos que actúan sobre el modelo .....             | 22  |
| 2.4 Modelo dinámico de 6 grados de libertad .....                   | 31  |
| Chapter 3. Control del vehículo .....                               | 34  |
| 3.1 Controladores PID .....                                         | 37  |
| 3.2 Control mediante el modelo simplificado: Matriz de control..... | 38  |
| 3.3 Modelo dinámico completo y retroalimentación .....              | 42  |
| Chapter 4. Incorporación modelos hexacóptero y octocóptero.....     | 43  |
| 4.1 Hexacóptero .....                                               | 45  |
| 4.2 Octocóptero .....                                               | 59  |
| Chapter 5. Mejora del modelo.....                                   | 68  |
| 5.1 Modelo experimental de las baterías .....                       | 68  |
| 5.2 Efecto suelo basado en altura no en altitud .....               | 79  |
| 5.3 Condición suelo .....                                           | 80  |
| 5.4 Fallo rotor.....                                                | 80  |
| 5.5 Modo de operación básico y funciones automatizadas.....         | 82  |
| Chapter 6. Desarrollo en Matlab/Simulink .....                      | 84  |
| 6.1 MATLAB.....                                                     | 84  |
| 6.2 SIMULINK.....                                                   | 85  |
| Chapter 7. Interfaz gráfica .....                                   | 101 |
| 7.1 Menú de configuración .....                                     | 101 |
| 7.2 Interfaz gráfica .....                                          | 101 |
| Chapter 8. Resultados .....                                         | 104 |
| 8.1 Estudio comparativo del modelo propulsivo .....                 | 104 |
| 8.2 Estudio comparativo del modelo de batería.....                  | 105 |



|                                                  |     |
|--------------------------------------------------|-----|
| 8.3 Evidencias: Monitoreo de la simulación ..... | 109 |
| Chapter 9. Conclusiones.....                     | 123 |
| Chapter 10. Trabajos futuros .....               | 124 |
| Bibliografía .....                               | 125 |

## Índice de figuras

|                                                                                                                    |    |
|--------------------------------------------------------------------------------------------------------------------|----|
| Figura 1. Principales elementos del vehículo Ingenuity [Fuente: [4]].....                                          | 12 |
| Figura 2. Piloto manejando simulador RPAS [Fuente: [5]] .....                                                      | 13 |
| Figura 3. Fotograma obtenido de un simulados RPAS [Fuente: [5]] .....                                              | 14 |
| Figura 4. Sistemas de referencia asumidos en el modelo.....                                                        | 17 |
| Figura 5. Rotaciones que definen la orientación del vehículo [Fuente: [6]] .....                                   | 18 |
| Figura 6. Esquema de fuerzas presentes en el modelo multirrotor.....                                               | 22 |
| Figura 7. Desacoplamiento velocidad angular de los rotores de un multirrotor [Fuente: [9]]... 26                   |    |
| Figura 8. Esquema del control del modelo por lazo cerrado .....                                                    | 36 |
| Figura 9. Esquema de un controlador PID genérico.....                                                              | 38 |
| Figura 10. Distribución de rotores en un hexacóptero .....                                                         | 39 |
| Figura 11. Esquema de fuerzas y momentos que ejerce un rotor .....                                                 | 40 |
| Figura 12. Cuadricoptero real.....                                                                                 | 43 |
| Figura 13. Control de dimensiones de variables en Simulink-Model explorer .....                                    | 45 |
| Figura 14. Hexacóptero real .....                                                                                  | 46 |
| Figura 15. Motor hexacóptero MT4006-13.....                                                                        | 47 |
| Figura 16. Dimensiones motor hexacóptero MT4006-13 .....                                                           | 47 |
| Figura 17. Especificaciones técnicas motor hexacóptero MT4006-13 [Fuente: [10]] .....                              | 47 |
| Figura 18. Controlador de velocidad motor hexacóptero.....                                                         | 48 |
| Figura 19. Hélice del conjunto motor hexacóptero .....                                                             | 49 |
| Figura 20. Conjunto motor hexacóptero.....                                                                         | 49 |
| Figura 21. Resultados ensayo experimental similar al desarrollado en este documento .....                          | 50 |
| Figura 22. Bancada de ensayos para conjunto motor.....                                                             | 51 |
| Figura 23. Dispositivo para comprobar el estado de carga de la batería.....                                        | 52 |
| Figura 24. Coeficiente de empuje dimensional del hexacóptero en función del régimen de giro 54                     |    |
| Figura 25. Ley de empuje en función del cuadrado de la velocidad angular de giro del motor MT4006-13 .....         | 55 |
| Figura 26. Ajuste lineal por mínimos cuadrado y primeras aproximaciones lineales a la ley de empuje .....          | 56 |
| Figura 27. Ajuste polinómico de segundo grado por mínimos cuadrado y aproximación lineal de la ley de empuje ..... | 57 |
| Figura 28. Ajuste polinómico de tercer grado por mínimos cuadrado y aproximación lineal de la ley de empuje.....   | 58 |
| Figura 29. Octocóptero real .....                                                                                  | 59 |
| Figura 30. Motor octocóptero MN4014-9 .....                                                                        | 60 |
| Figura 31. Especificaciones técnicas motor octocóptero MN4014-9 [Fuente: [12]] .....                               | 60 |
| Figura 32. Controlador de velocidad motor octocóptero .....                                                        | 61 |
| Figura 33. Conjunto motor octocóptero .....                                                                        | 61 |

|                                                                                                                     |     |
|---------------------------------------------------------------------------------------------------------------------|-----|
| Figura 34. Coeficiente de empuje dimensional del octocóptero en función del régimen de giro .....                   | 63  |
| Figura 35. Ley de empuje en función del cuadrado de la velocidad angular de giro del motor MN4014-9 .....           | 64  |
| Figura 36. Zoom a bajas revoluciones de la ley de empuje del motor MN4014-9 .....                                   | 65  |
| Figura 37. Conjunto de aproximaciones lineales de la ley de empuje obtenidas tras diferentes ajustes. ....          | 66  |
| Figura 38. Representación de la ley de empuje asumida para el motor MT4006-13 junto a los datos experimentales..... | 67  |
| Figura 39. Baterías HRB tipo LiPo a bordo del hexacóptero real. ....                                                | 70  |
| Figura 40. Ajuste polinómico de cuarto grado de los datos experimentales del consumo de la batería HRB .....        | 71  |
| Figura 41. Ajuste polinómico de segundo grado de los datos experimentales de la potencia de la batería HRB.....     | 72  |
| Figura 42. Curva de consumo de la batería HRB en función del empuje que suministra el motor MT4006-13 .....         | 73  |
| Figura 43. Potencia consumida por el motor MT4006-13 en función del régimen de giro.....                            | 73  |
| Figura 44. Eficiencia empuje/potencia en función del régimen de giro del motor MT4006-13. 74                        |     |
| Figura 45. Baterías RC tipo LiPo a bordo del octocóptero real. ....                                                 | 75  |
| Figura 46. Ajuste polinómico de cuarto grado de los datos experimentales del consumo de la batería .....            | 76  |
| Figura 47. Ajuste polinómico de tercer grado de los datos experimentales de la potencia de la batería RC.....       | 77  |
| Figura 48. Curva de consumo de la batería RC en función del empuje que suministra el motor MN4014-9 .....           | 77  |
| Figura 49. Potencia consumida por el motor MN4014-9 en función del régimen de giro .....                            | 78  |
| Figura 50. Eficiencia empuje/potencia en función del régimen de giro del motor MN4014-9 ..                          | 78  |
| Figura 51. Flujo de aire alrededor de un cuadricoptero volando cerca del suelo .....                                | 79  |
| Figura 52. Esquema de efecto suelo basado en elevación .....                                                        | 79  |
| Figura 53. Modelo Simulink del multirrotores.....                                                                   | 86  |
| Figura 54. Módulo recibir datos .....                                                                               | 86  |
| Figura 55. Bloque control .....                                                                                     | 89  |
| Figura 56. Bloque PID del control manual.....                                                                       | 90  |
| Figura 57. Bloque PID del modo hover con realimentación del altura .....                                            | 91  |
| Figura 58. Bloque selección de modo de control.....                                                                 | 92  |
| Figura 59. Bloque control de la velocidad angular de los rotores.....                                               | 93  |
| Figura 60. Bloque motores .....                                                                                     | 94  |
| Figura 61. Bloque modelo dinámico .....                                                                             | 95  |
| Figura 62. Interior del bloque modelo dinámico .....                                                                | 96  |
| Figura 63. Bloque condición suelo .....                                                                             | 97  |
| Figura 64. Bloque selección de modo .....                                                                           | 98  |
| Figura 65. Bloque descarga de la batería .....                                                                      | 99  |
| Figura 66. Módulo enviar datos .....                                                                                | 100 |
| Figura 67. Menu de configuración .....                                                                              | 101 |



|                                                                                                      |     |
|------------------------------------------------------------------------------------------------------|-----|
| Figura 68. Interfaz gráfica con vehículo hexacóptero .....                                           | 101 |
| Figura 69. Interfaz gráfica con vehículo octocóptero .....                                           | 102 |
| Figura 70. Menu de opciones.....                                                                     | 102 |
| Figura 71. Funciones del menu de opciones.....                                                       | 103 |
| Figura 72. Modos de asistencia de vuelo .....                                                        | 103 |
| Figura 73. Cuadro informativo .....                                                                  | 103 |
| Figura 74. Reloj informativo y estado de la batería.....                                             | 104 |
| Figura 75. Estado del mando del piloto .....                                                         | 104 |
| Figura 76. Representación de las leyes de empuje y los datos empíricos obtenidos en el ensayo .....  | 105 |
| Figura 77. Representación de las curvas de consumo .....                                             | 106 |
| Figura 78. Relación consumo empuje .....                                                             | 107 |
| Figura 79. Potencia consumida en función del régimen de giro.....                                    | 108 |
| Figura 80. Eficiencia empuje/potencia en función del régimen de giro .....                           | 108 |
| Figura 81. Descarga de la batería recogida durante la simulación – Vuelo hexacóptero .....           | 109 |
| Figura 82. Velocidad angular media de los rotores durante la simulación – Vuelo hexacóptero .....    | 110 |
| Figura 83. Corriente media consumida por los rotores durante la simulación – Vuelo hexacóptero ..... | 110 |
| Figura 84. Descarga de la batería recogida durante la simulación – Vuelo octocóptero .....           | 111 |
| Figura 85. Velocidad angular media de los rotores durante la simulación – Vuelo octocóptero .....    | 111 |
| Figura 86. Corriente media consumida por los rotores durante la simulación – Vuelo octocóptero.....  | 112 |
| Figura 87. Valor de altitud recogido durante la simulación.....                                      | 113 |
| Figura 88. Valor de AGL recogido durante la simulación .....                                         | 113 |
| Figura 89. Valores de AGL inferiores a 40 cm durante la simulación.....                              | 114 |
| Figura 90. Incremento de empuje debido al efecto suelo .....                                         | 114 |
| Figura 91. Variación de altitud durante la simulación .....                                          | 115 |
| Figura 92. Variación de roll y pitch durante la simulación .....                                     | 115 |
| Figura 93. Evolución de la flag de la condición suelo durante la simulación .....                    | 116 |
| Figura 94. Velocidades angulares de los rotores del hexacóptero durante la simulación .....          | 117 |
| Figura 95. Número de rotores del hexacóptero comandados para parar durante la simulación.....        | 117 |
| Figura 96. Velocidades angulares de los rotores del octocóptero durante la simulación .....          | 118 |
| Figura 97. Número de rotores del octocóptero comandados para parar durante la simulación .....       | 118 |
| Figura 98. Evolución de la altitud de vuelo durante la simulación .....                              | 119 |
| Figura 99. Evolución de la coordenada X durante la simulación .....                                  | 120 |
| Figura 100. Evolución de la flag del modo avance frontal .....                                       | 120 |
| Figura 101. Evolución de la altitud de vuelo durante la simulación .....                             | 121 |
| Figura 102. Evolución de la coordenada Y durante la simulación.....                                  | 121 |
| Figura 103. Evolución de la flag del modo avance frontal .....                                       | 121 |
| Figura 104. Evolución de la altitud y la variable HoverTarget durante la simulación.....             | 122 |



## Resumen

Los sistemas de aeronaves no tripuladas RPAS (Remotely Piloted Aircraft Systems) son vehículos aéreos capaces de realizar vuelos autónomos y también pueden ser controlados a distancia por un operador humano. Estos sistemas se componen de diversos componentes esenciales, como sistemas de control, sensores y equipos especializados adaptados a su uso específico.

En el vertiginoso mundo de la tecnología RPAS, los avances van más allá del desarrollo y la mejora de los vehículos en sí. Los simuladores de vuelo desempeñan un papel fundamental en la formación y desarrollo de habilidades de los pilotos y operadores de RPAS. Estos sofisticados sistemas permiten recrear de manera virtual entornos y condiciones de vuelo, brindando una experiencia inmersiva y realista a el usuario.

Los simuladores de RPAS son programas informáticos diseñados para simular el vuelo de aeronaves no tripuladas. Estos sistemas reproducen con precisión aspectos técnicos y físicos del vuelo, como la dinámica de la aeronave, las condiciones meteorológicas, la geografía del terreno y la respuesta a las entradas de control. Los simuladores se ejecutan en ordenadores de alto rendimiento y suelen utilizarse junto con hardware adicional, como dispositivos de control y pantallas múltiples, para lograr una experiencia de vuelo más realista.

El objetivo principal de este documento es la descripción teórica del modelo heredado por el autor de antiguos alumnos de la universidad, y la justificación de los avances en el desarrollo del código, con el fin de hacer el simulador más versátil y realista.

El principal hito de este desarrollo es la integración de los vehículos hexacópteros y octocópteros a la simulación, esta integración se ha conseguido caracterizando los vehículos a partir de resultados empíricos obtenidos a partir de una bancada de ensayos para motores de dron suministrada por el tutor del trabajo. Con este ensayo se consigue modelar el sistema de propulsión de ambos vehículos, obteniéndose la ley de empuje teórica basada en estos datos. A partir de modificaciones realizadas en el modelo Matlab-Simulink, se han añadido otras mejoras a la simulación. Estas mejoras incluyen el desarrollo de un modelo de batería para ambos vehículos también basados en un ensayo experimental donde se analiza la descarga eléctrica de esta. Otros avances han sido la incorporación del efecto suelo basado en la altura en vez de en la altitud, la introducción de la condición de suelo para una simulación más realista, la inclusión de la condición de fallo de rotor disponible solo en multirrotores de más de cuatro rotores y la incorporación de funcionalidades básicas automatizadas para el control del vuelo del vehículo.

Además, se dispone de una interfaz gráfica programada en el motor gráfico Unity, desarrollada gracias a la colaboración con otro alumno de la universidad en este proyecto, esta interfaz nos servirá para demostrar todos los avances logrados en el desarrollo de este proyecto.

Como resumen de los principales hitos que se han logrado en este proyecto, se ha conseguido incorporar al modelo Matlab-Simulink dos nuevos vehículos con su planta propulsora y el modelo de descarga de su batería, además se incorpora la capacidad de simular la condición de fallo rotor, así como, otro enfoque de control diferente al manual a través de funciones automatizadas, entre las que destacamos, aterrizaje, avance frontal equilibrado, avance lateral equilibrado y hover automático basado en altitud.

# Chapter 1. Introducción

## 1.1 Objetivos

El tema de este proyecto ha sido propuesto por el tutor académico del mismo con la intención de aportar por parte del alumno cierto desarrollo adicional a un modelo de multirrotor RPAS elaborado en Matlab-Simulink por antiguos alumnos. Para más información del estado del modelo previo a este trabajo el lector puede consultar [1], [2] y [3].

Entre los principales objetivos del trabajo podemos destacar:

1. Estudio del modelo del multirrotor parametrizado: en este trabajo se desarrollan todas las herramientas matemáticas necesarias para construir un modelo dinámico parametrizado en función del número de rotores, dimensiones y características geométricas del vehículo considerado.
2. Generalización del modelo Matlab-Simulink para multirrotores de N rotores: se pretende generalizar el código de la simulación para que, de forma realista, esta represente el movimiento dinámico del vehículo independientemente del número de rotores de los que disponga.
3. Caracterización experimental e incorporación al modelo del hexacóptero y octocóptero: en este proyecto haciendo uso de una bancada de ensayo se va a caracterizar la planta propulsora de los vehículos hexacóptero y octocóptero para después simular su vuelo.
4. Modelo experimental del consumo de dos baterías LiPo: haciendo uso de la misma bancada de ensayos se va a registrar el consumo de las baterías del hexacóptero y octocóptero para incorporar a la simulación el modelo de descarga de la batería y logra así una autonomía realista del vehículo en vuelo.
5. Aumentar el realismo de la simulación y añadir condición de fallo rotor: se van a añadir al modelo ciertos efectos que confieran realismo a la simulación tales como el efecto suelo basado en altura y no en altitud, la condición suelo cuando el vehículo repose sobre una superficie o la posibilidad de parar uno o más rotores simulando una avería en vuelo.
6. Implementar ciertos automatismos en el control del vehículo simulado: además de ampliar el modelo se ha querido modificar el control del vehículo, dotando el vehículo simulado de ciertas funcionalidades automatizadas, entre las que destacamos el avance frontal, el avance lateral, el aterrizaje automático o de emergencia y el “hover” dependiente del nivel de altura que indique el piloto.

## **1.2 Estado del arte**

Antes de comenzar con el desarrollo matemático e implementación del modelo desarrollado en este trabajo, en este apartado se va a definir el tipo de vehículo que se pretende simular, así como sus principales aplicaciones en la industria actual.

### **1.2.1 Tecnología RPAS**

En la última década, el mundo ha sido testigo de un rápido avance en el campo de la tecnología aeroespacial. Una de las innovaciones más notables en este ámbito es el desarrollo y la proliferación de los Sistemas de Aeronaves Pilotadas de Forma Remota (RPAS, por sus siglas en inglés). Los RPAS, comúnmente conocidos como drones, han revolucionado diversas industrias y están desempeñando un papel crucial en la transformación de la sociedad y la economía a nivel global.

Los RPAS, son aeronaves no tripuladas que bien son controladas de manera remota por un operador humano o bien pueden volar de manera autónoma.

Estas aeronaves están compuestas de diferentes componentes esenciales para su correcto funcionamiento, como el vehículo aéreo en sí mismo, los sistemas de control remoto, los sensores y otros equipos especializados según el uso específico del vehículo.

#### ***1.2.1.1 Principales aplicaciones de los RPAS***

Los RPAS han encontrado aplicaciones en una amplia gama de industrias y sectores, una de las áreas donde han tenido un impacto significativo es la industria del cine. Los drones equipados con cámaras de alta resolución y estabilizadores giroscópicos pueden capturar imágenes y videos aéreos impresionantes, ofreciendo perspectivas únicas y nuevas posibilidades creativas.

Además de la fotografía, los RPAS se utilizan cada vez más en el sector agrícola, estas aeronaves pueden realizar inspecciones precisas de cultivos, detectar plagas, y monitorear el estado de los suelos. Esta información permite a los agricultores tomar decisiones más adecuadas y optimizar sus prácticas de cultivo, lo que a su vez conduce a una mayor eficiencia y rendimiento en la producción agrícola.

Otra área en la que los RPAS están dejando su huella es la logística y el transporte, por ejemplo, empresas como Amazon y UPS están experimentando con la entrega de paquetes mediante drones, lo que promete una entrega más rápida y eficiente. Además, los RPAS se utilizan en misiones de búsqueda y rescate, inspección de infraestructuras y vigilancia, lo que ayuda a reducir el riesgo por factor humano y ampliar las capacidades de respuesta en situaciones críticas.

#### ***1.2.1.2 Desafíos y oportunidades***

Si bien los RPAS ofrecen una amplia gama de beneficios y oportunidades, también plantean desafíos significativos, la seguridad es uno de los principales temas de preocupación que rodea a la tecnología RPAS. El uso indebido de drones puede plantear riesgos para la seguridad de las personas. También existen preocupaciones sobre la posible interferencia con el tráfico aéreo, la colisión con aeronaves tripuladas y la invasión de la privacidad al capturar imágenes o datos sin el consentimiento adecuado.

Además, la regulación y la legislación en torno a los RPAS aún se están desarrollando en muchos países. Los gobiernos y las autoridades reguladoras se enfrentan al desafío de establecer políticas y normativas que equilibren la promoción del desarrollo de este tipo de vehículos con la protección de la seguridad y la privacidad.

Sin embargo, a pesar de estos desafíos, los RPAS también presentan oportunidades emocionantes para el futuro desarrollo tecnológico, el aumento de la capacidad de procesamiento en los dispositivos, cada vez mayor, permite la creación de drones más autónomos.

Su capacidad para acceder a áreas de difícil acceso o volar a altitudes elevadas los convierte en valiosas herramientas para la observación y el estudio de fenómenos naturales y cambios ambientales.

La tecnología RPAS tiene un enorme potencial en el ámbito de la investigación y la ciencia. Los drones pueden utilizarse para la recopilación de datos en áreas remotas o peligrosas, el monitoreo ambiental y la investigación geográfica incluso de otro planeta, por ejemplo, la NASA desplegó en la superficie de Marte el dron-helicóptero bautizado como Ingenuity en abril de 2021 a una distancia de aproximadamente 294 millones de kilómetros de distancia de la Tierra.

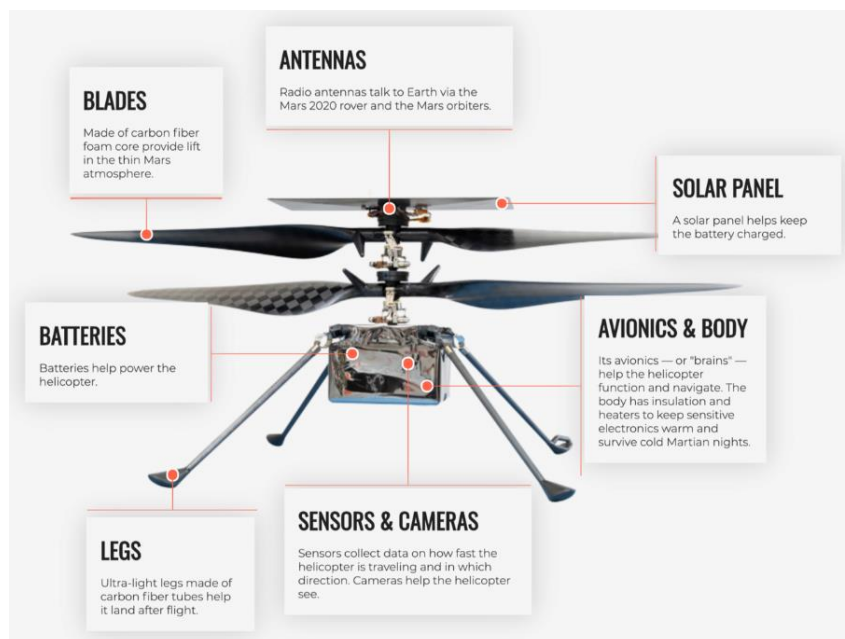


Figura 1. Principales elementos del vehículo Ingenuity [Fuente: [4]]

En conclusión, la tecnología RPAS ha revolucionado múltiples sectores y está impulsando una verdadera revolución aérea en la industria mundial, a medida que la tecnología continúa evolucionando y mejorando, los RPAS están desempeñando un papel cada vez más importante en nuestra sociedad y economía. Sin embargo, es crucial abordar los desafíos relacionados con la seguridad y la privacidad de las personas, así como establecer un marco regulatorio sólido. Al hacerlo, podemos aprovechar al máximo las oportunidades que los RPAS ofrecen y abrir nuevas posibilidades para el futuro desarrollo tecnológico.



### 1.2.2 Simuladores RPAS

En el vertiginoso mundo de la tecnología RPAS los avances no se limitan solo al desarrollo y la mejora de los propios vehículos, una herramienta fundamental en la formación y entrenamiento de pilotos y operadores de RPAS son los simuladores de vuelo.

Estos sofisticados sistemas permiten recrear de manera virtual el entorno y las condiciones de vuelo de los RPAS, ofreciendo una experiencia inmersiva y realista. A lo largo de esta introducción, exploraremos los simuladores de RPAS y sus diversas utilidades en la formación, entrenamiento y desarrollo de habilidades de los pilotos.

Los simuladores de RPAS son programas informáticos diseñados para simular el vuelo de aeronaves no tripuladas, estos sistemas reproducen con precisión los aspectos técnicos y físicos del vuelo, como la dinámica de la aeronave, las condiciones meteorológicas, la geografía del terreno y la respuesta a las entradas de control. Los simuladores se ejecutan en computadoras de alto rendimiento y a menudo se utilizan junto con hardware adicional, como mandos de control y pantallas múltiples, para una experiencia de vuelo más realista.

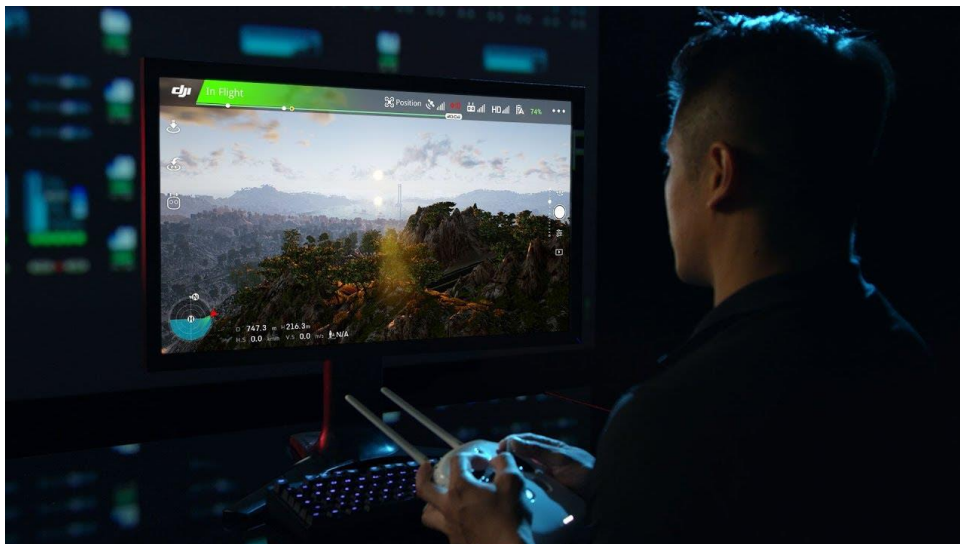


Figura 2. Piloto manejando simulador RPAS [Fuente: [5]]

#### 1.2.2.1 Aplicaciones de los simuladores de RPAS

Como se ha comentado los simuladores de RPAS desempeñan un papel fundamental en la formación y entrenamiento de pilotos y operadores de drones, ya que proporcionan un entorno seguro y controlado donde los usuarios pueden aprender y practicar diversas habilidades de vuelo sin los riesgos asociados con el vuelo real.

Algunas de las utilidades más destacadas de los simuladores de RPAS son:

a) Formación inicial: los simuladores permiten a los principiantes familiarizarse con los controles y la dinámica de vuelo de los RPAS. Los usuarios pueden aprender los conceptos básicos de operación, como despegue, aterrizaje, control de altitud y dirección, y maniobras básicas. Esto sienta las bases para una formación más avanzada y el desarrollo de habilidades específicas.

b) Entrenamiento de habilidades avanzadas: estos simuladores permiten a los pilotos practicar y perfeccionar una amplia gama de habilidades, como vuelo en condiciones climáticas adversas, vuelo en formación, navegación y operaciones de emergencia. Además, los simuladores pueden recrear escenarios complejos útiles para representar misiones de búsqueda y rescate, vigilancia de infraestructuras o inspección de cultivos, brindando a los pilotos la oportunidad de enfrentarse a desafíos realistas en un entorno seguro.

c) Evaluación y certificación: los simuladores de RPAS también se utilizan para evaluar las habilidades de los pilotos y operadores. Los programas de simulación pueden incluir métricas y criterios de evaluación para medir el rendimiento en áreas específicas. Esto es especialmente útil en el proceso de certificación y cumplimiento de los requisitos regulatorios establecidos por las autoridades de aviación civil.

d) Investigación y desarrollo: los simuladores también son herramientas valiosas para la investigación y el desarrollo de nuevas tecnologías y estrategias de vuelo. Los investigadores pueden utilizar los simuladores para probar y mejorar algoritmos de control, sistemas de detección y prevención de colisiones o para explorar futuras aplicaciones de estos vehículos.



Figura 3. Fotograma obtenido de un simulador RPAS [Fuente: [5]]

### 1.2.2.2 Ventajas de los simuladores de RPAS

El uso de simuladores de RPAS ofrece varias ventajas significativas en comparación con el vuelo real. Algunas de estas ventajas son:

a) Reducción de costes: el vuelo real de un RPAS implica gastos significativos, asociados a el gasto de combustible, a el mantenimiento de la aeronave y a los seguros asociados, los simuladores permiten reducir estos costes, ya que al no haber necesidad de desplegar los drones físicamente se evitan estos daños, accidentes y costes legales necesarios durante el entrenamiento real.



b) Seguridad y riesgos minimizados: los simuladores proporcionan un entorno seguro donde los pilotos pueden aprender y practicar sin correr riesgos asociados con el vuelo real. Esto es especialmente importante durante la fase inicial de formación y entrenamiento, donde el vuelo simulado permite cometer errores sin consecuencias graves.

c) Flexibilidad y control: los simuladores de RPAS permiten a los usuarios tener un control total sobre el entorno y las condiciones de vuelo. Como se ha comentado es posible simular una amplia gama de escenarios y condiciones climáticas, lo que proporciona una experiencia de entrenamiento más completa y adaptada a las necesidades específicas de cada piloto.

d) Análisis del vuelo: los simuladores registran datos detallados del vuelo virtual, lo que permite un análisis exhaustivo del rendimiento del piloto. Esto permite identificar áreas de mejora dónde corregir errores para facilitar el aprendizaje y el desarrollo de las habilidades del piloto.

Los simuladores de RPAS desempeñan un papel esencial en la formación, entrenamiento y desarrollo de habilidades de pilotos y operadores de drones. Estas herramientas permiten a los usuarios aprender, practicar y perfeccionar diversas habilidades de vuelo en un entorno seguro y controlado.

Además, los simuladores ofrecen flexibilidad, costos reducidos y una amplia gama de escenarios para una formación más completa y adaptada. A medida que la tecnología continúa avanzando, los simuladores de RPAS seguirán desempeñando un papel crucial en el desarrollo de la industria y en la promoción de prácticas de vuelo seguras y eficientes.

### 1.3 Descripción del contenido

Con la intención de ofrecer cierta orientación ante el contenido y distribución de la distinta información en este trabajo, a continuación, se hace un pequeño resumen de los principales temas que tratan cada capítulo.

**Capítulo 1: Introducción.** Al principio de este capítulo se ofrece una visión general de los objetivos propuestos para este trabajo. Después se hace una descripción general de la tecnología RPAS, sus principales aplicaciones en la industria y también se analiza el interés y utilidad del desarrollo de simuladores de vuelo de este tipo de vehículo. Por último, se describe el contenido de todo documento dividido en capítulos.

**Capítulo 2: Descripción del modelo.** Se describe todo el planteamiento matemático necesario para construir un simulador de vuelo parametrizado: Sistemas de referencia empleados, hipótesis asumidas por modelo dinámico de seis grados de libertad, ecuaciones generales del sólido rígido, fuerzas y momentos que actúan sobre el modelo.

**Capítulo 3: Control del vehículo.** En este apartado se desarrolla el control en lazo cerrado de la actitud y posición del vehículo, desglosándolo en sus tres principales elementos: El bloque de controladores PID, el modelo dinámico simplificado y el modelo dinámico real que retroalimenta variables al bloque PID. Se hace un análisis comparativo de la matriz de control para los diferentes multirrotores incorporados a el modelo.



**Capítulo 4: Incorporación vehículos hexacóptero y octocóptero.** Apartado dónde se hace un análisis general de los dos multirrotores que se añaden a el modelo Matlab-Simulink en este proyecto. Se describen y presentan los resultados de los ensayos experimentales llevados a cabo haciendo uso de una bancada de ensayos con el fin de modelar la planta propulsora de ambos vehículos para después, haciendo uso de estos datos, proceder a modelar de manera teórica su comportamiento.

**Capítulo 5: Mejora del modelo.** En este apartado se describen las mejoras implementadas a lo largo de este trabajo a el modelo Matlab-Simulink heredado por el auto de antiguos alumnos de la universidad. Entre estas mejoras podemos destacar el desarrollo de un modelo de batería para ambos vehículos basado también en resultados obtenidos experimentalmente, el efecto suelo baso en altura y no en altitud, la condición suelo para conferir mayor realismo a la simulación, la condición de fallo rotor disponible solo en multirrotores de más de cuatro rotores y la incorporación de funcionalidades de control básicas automatizadas por el modelo.

**Capítulo 6: Desarrollo en Matlab-Simulink.** Aplicación directa de todos los planteamientos matemáticos de capítulos anteriores. Descripción del contenido de los constructores y archivos de configuración desarrollados en Matlab y descripción completa del flujo de ejecución del modelo Simulink simulado en tiempo real. Además del flujo se entra a analizar a alto nivel la principal utilidad de cada bloque Simulink.

**Capítulo 7: Interfaz gráfica.** Presentación visual de la interfaz gráfica programada en el motor gráfico Unity, la cual ha sido desarrollada en [RAUL], con carácter colaborativo con este proyecto. En este apartado se analiza la interfaz y se da una breve explicación de cómo usarla con el fin de hacer un correcto uso de la simulación.

**Capítulo 8: Resultados.** En primer lugar, se describe el análisis comparativo utilizando los parámetros obtenidos experimentalmente, tanto de la planta propulsora como de las baterías de los multirrotores estudiados. Después con el objeto de mostrar ciertas evidencias extraídas de la simulación se han monitorizado parámetros clave que demuestren la correcta implantación de las mejoras propuestas para el modelo.

**Capítulo 9: Trabajos futuros.** En este apartado se recogen diferentes ideas o planteamientos que han ido surgiendo a lo largo del desarrollo del proyecto para mejorar el realismo de la simulación o las funcionalidades del vehículo en movimiento.

**Capítulo 10: Bibliografía.** Se recoge el conjunto de referencias a la literatura consultada para el desarrollo de este documento.

## Chapter 2. Descripción del modelo

Este apartado recoge toda la información relevante del modelo físico asumido por la simulación para representar el movimiento libre de un multirroto RPAS en un entorno gráfico digital.

Se presentan los sistemas de referencia elegidos, las matrices de giro entre sistemas, la matriz Jacobiana del sistema móvil y las hipótesis y ecuaciones asumidas en el modelo de seis grados de libertad para el cálculo de cada magnitud física que determinará el movimiento del vehículo por el entorno. Las ecuaciones empleadas por la simulación las cuales determinan las posición y orientación del vehículo en cada iteración serán las correspondientes a la teoría de sólido rígido.

### 2.1 Sistemas de referencia

En este tipo de problemas, se utilizan comúnmente dos sistemas de referencia para realizar los cálculos numéricos del modelo.

El primer sistema es un sistema de referencia fijo {0} que está ligado al suelo o a la estación de tierra. El origen de coordenadas de este sistema es un punto arbitrario que se tiene en cuenta al establecer los parámetros de telemetría y control (GPS). Se trata de un triedro definido a derechas con el eje Z apuntando hacia el centro de la Tierra.

El segundo sistema empleado es un sistema de referencia móvil {1} que es solidario al movimiento relativo del dron. Aunque no es esencial, se recomienda que el origen de coordenadas de este sistema coincida con el centro de masas del vehículo (CdM) y que la orientación de sus ejes coincida con los ejes de simetría del dron para facilitar los cálculos. Se trata de un triedro definido a derechas con el eje Z apuntando hacia abajo.

En la Figura 4 se muestra de manera esquemática como se definen ambos sistemas de referencia:

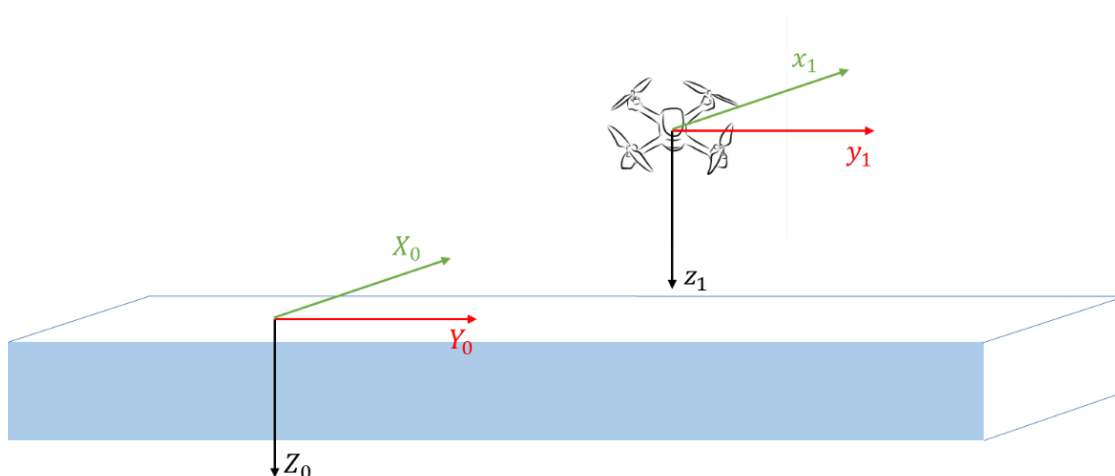


Figura 4. Sistemas de referencia asumidos en el modelo

### 2.1.1.1 Matrices de giro

Para transformar vectores de un sistema de referencia a otro, se necesita conocer la posición y la orientación relativa entre ambos. Esto se consigue mediante la definición de matrices de rotación.

Las matrices de rotación se utilizan para expresar una rotación en el espacio euclídeo alrededor de un determinado eje y se pueden encadenar consecutivamente para realizar múltiples rotaciones.

Para cambiar de base, se requieren tres rotaciones independientes en un orden determinado, y existen diversos criterios para elegir estas rotaciones. El criterio de rotación más común es el conocido como los ángulos de Euler, entre los que aparecen los siguientes giros:

1. Precesión
2. Nutación
3. Rotación propia.

Sin embargo y ante las numerosas posibilidades, hay que elegir un criterio acorde al modelo, por lo que en este trabajo se va a elegir la siguiente convención de ángulos de rotación típico para resolver problemas en aviación.

En el enfoque típico aeronáutico, se realizarán tres rotaciones sobre un vector arbitrario en una base fija  $\{0\}$ , tal y como muestra la Figura 5.

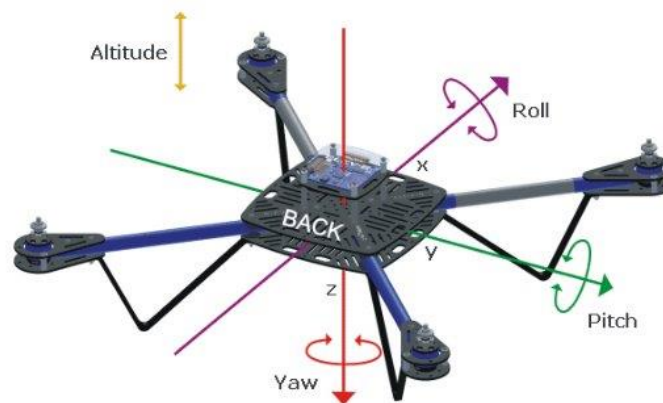


Figura 5. Rotaciones que definen la orientación del vehículo [Fuente: [6]]

Haciendo la analogía con los ángulos de Euler tenemos un giro de precesión alrededor del eje Z  $\{0\}$ . En la aviación, el ángulo asociado con esta rotación se conoce comúnmente como guiñada o “yaw”, a lo largo de este trabajo nos referiremos a este ángulo con la letra griega  $\varphi$ .

$$\begin{pmatrix} X' \\ Y' \\ Z' \end{pmatrix} = \begin{pmatrix} \cos \varphi & \sin \varphi & 0 \\ -\sin \varphi & \cos \varphi & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} X_0 \\ Y_0 \\ Z_0 \end{pmatrix}$$

Ecuación 1. Rotación por ángulo de guiñada

El giro nutación se lleva a cabo en torno al nuevo eje Y' obtenido tras la precesión, hallando así el ángulo conocido comúnmente como cabeceo o "pitch", a lo largo de este documento nos referiremos a este ángulo con la letra griega  $\Theta$ .

$$\begin{pmatrix} X'' \\ Y'' \\ Z'' \end{pmatrix} = \begin{pmatrix} \cos \Theta & 0 & -\sin \Theta \\ 0 & 1 & 0 \\ \sin \Theta & 0 & \cos \Theta \end{pmatrix} \begin{pmatrix} X' \\ Y' \\ Z' \end{pmatrix}$$

*Ecuación 2. Rotación por ángulo de cabeceo*

En cuanto al giro por rotación propia se consigue girando el ángulo comúnmente conocido como alabeo o "roll" sobre el eje X'' obtenido tras la nutación, a lo largo de este documento el ángulo de giro por alabeo se designará con la letra griega  $\psi$ .

$$\begin{pmatrix} X_0 \\ Y_0 \\ Z_0 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos \psi & \sin \psi \\ 0 & -\sin \psi & \cos \psi \end{pmatrix} \begin{pmatrix} X'' \\ Y'' \\ Z'' \end{pmatrix}$$

*Ecuación 3. Rotación por ángulo de alabeo*

Estos tres ángulos determinarán la orientación del vehículo respecto al sistema fijo, lo que al final se traduce en la orientación del dron respecto a tierra, concluyendo así, la transformación de los ejes fijos ligados a tierra a los ejes móviles ligados al dron.

Por lo tanto, suponiendo conocidos los tres ángulos "yaw", "pitch" y "roll" y concatenando las tres matrices de rotación obtenemos la matriz R que trasforma cualquier magnitud física vectorial expresada en el sistema fijo {0} al sistema móvil {1}:

$$R = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos \psi & \sin \psi \\ 0 & -\sin \psi & \cos \psi \end{pmatrix} \begin{pmatrix} \cos \Theta & 0 & -\sin \Theta \\ 0 & 1 & 0 \\ \sin \Theta & 0 & \cos \Theta \end{pmatrix} \begin{pmatrix} \cos \varphi & \sin \varphi & 0 \\ -\sin \varphi & \cos \varphi & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

*Ecuación 4. Matriz de rotación sistema fijo-móvil*

$$\begin{pmatrix} X_1 \\ Y_1 \\ Z_1 \end{pmatrix} = R \begin{pmatrix} X_0 \\ Y_0 \\ Z_0 \end{pmatrix}$$

*Ecuación 5. Transformación sistema de referencia fijo-móvil*

Conocidos los ángulos de rotación entre los sistemas anteriormente descritos, se puede obtener la matriz inversa a R que nos permitirá transformar las magnitudes vectoriales físicas ligadas a los ejes móviles en ejes fijos, con esta información es trivial demostrar que la inversa de la matriz R viene determinada por:



$${}^iR = R^{-1} = \begin{pmatrix} \cos \varphi & -\sin \varphi & 0 \\ \sin \varphi & \cos \varphi & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \cos \Theta & 0 & \sin \Theta \\ 0 & 1 & 0 \\ -\sin \Theta & 0 & \cos \Theta \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos \psi & -\sin \psi \\ 0 & \sin \psi & \cos \psi \end{pmatrix}$$

*Ecuación 6. Matriz de rotación sistema móvil-fijo*

$$\begin{pmatrix} X_0 \\ Y_0 \\ Z_0 \end{pmatrix} = R^{-1} \begin{pmatrix} X_1 \\ Y_1 \\ Z_1 \end{pmatrix}$$

*Ecuación 7. Transformación sistema de referencia móvil-fijo*

Las matrices R y su inversa,  $R^{-1}$ , permiten transformar magnitudes vectoriales que indiquen movimiento lineal. Este tipo de magnitudes vectoriales son las aceleraciones, las velocidades o las posiciones expresadas en los dos sistemas de referencia contemplados:

$$\begin{pmatrix} v_{x1} \\ v_{y1} \\ v_{z1} \end{pmatrix} = R \begin{pmatrix} v_{x0} \\ v_{y0} \\ v_{z0} \end{pmatrix} \quad \begin{pmatrix} a_{x1} \\ a_{y1} \\ a_{z1} \end{pmatrix} = R \begin{pmatrix} a_{x0} \\ a_{y0} \\ a_{z0} \end{pmatrix}$$

*Ecuación 8. Transformaciones magnitudes lineales fijo-móvil*

$$\begin{pmatrix} v_{x0} \\ v_{y0} \\ v_{z0} \end{pmatrix} = R^{-1} \begin{pmatrix} v_{x1} \\ v_{y1} \\ v_{z1} \end{pmatrix} \quad \begin{pmatrix} a_{x0} \\ a_{y0} \\ a_{z0} \end{pmatrix} = R^{-1} \begin{pmatrix} a_{x1} \\ a_{y1} \\ a_{z1} \end{pmatrix}$$

*Ecuación 9. Transformaciones magnitudes lineales móvil-fijo*

Estas matrices de rotación serán muy útiles en la resolución del sistema de fuerzas dinámico a través de Simulink, ya que nos permitirá transformar de manera rápida y eficiente magnitudes físicas expresadas en los diferentes sistemas según convenga en función de la ecuación que se quiere resolver.

### 2.1.1.2 Matriz Jacobiana

Como se ha comentado cuando queremos cambiar el sistema de referencia a una magnitud física que implica movimiento lineal basta con una matriz de rotación, sin embargo, cuando se trata de magnitudes físicas que implican movimiento angular entre sistemas, hay que utilizar otro tipo de matriz de transferencia, esta matriz concretamente es la matriz Jacobiana.

La velocidad de rotación del vehículo puede ser expresada según la velocidad angular del sistema de referencia ligado al CdM del dron, en ejes móviles {1}:  $\vec{\Omega} = (\Omega_x, \Omega_y, \Omega_z)$ , fácil de determinar, o también puede representarse en función de los ángulos de rotación que determinan la orientación de vehículo respecto a ejes tierra: "roll", "pitch" y "yaw" y de sus respectivas velocidades angulares, planteando las tres rotaciones completas del sistema móvil, se llega a el siguiente sistema: Ecuación 10 Ecuación 10, expresado en el sistema de referencia móvil {1}:



$$\Omega_x = -\dot{\varphi} \sin \theta + \dot{\psi}$$

$$\Omega_y = \dot{\varphi} \cos \theta \sin \psi + \dot{\theta} \cos \psi$$

$$\Omega_z = \dot{\varphi} \cos \theta \cos \psi - \dot{\theta} \sin \psi$$

*Ecuación 10. Velocidad angular de los ejes móviles en función de los ángulos de actitud*

Estando las velocidades angulares  $\vec{\Omega} = (\Omega_x, \Omega_y, \Omega_z)$  en el sistema móvil original, y las velocidades angulares  $\vec{n} = (\dot{\varphi}, \dot{\theta}, \dot{\psi})$  en los sistemas intermedios de giro

Si expresamos la Ecuación 11 de forma matricial obtenemos:

$$\begin{pmatrix} \Omega_x \\ \Omega_y \\ \Omega_z \end{pmatrix} = J \begin{pmatrix} \dot{\varphi} \\ \dot{\theta} \\ \dot{\psi} \end{pmatrix} = \begin{pmatrix} -\sin \theta & 0 & 1 \\ \cos \theta \sin \psi & \cos \psi & 0 \\ \cos \theta \cos \psi & -\sin \psi & 0 \end{pmatrix} \begin{pmatrix} \dot{\varphi} \\ \dot{\theta} \\ \dot{\psi} \end{pmatrix}$$

*Ecuación 11. Sistema matricial velocidad angular sistema referencia móvil*

Dónde J es la matriz Jacobiana de la transformación angular del sistema móvil, que nos permite obtener la velocidad angular del sistema de referencia fijo en función de los ángulos “roll”, “pitch”, “yaw” y sus velocidades angulares.

A su vez obtenemos la matriz inversa de la Jacobiana,  $J^{-1}$ , la cual nos permite obtener las velocidades angulares de “roll”, “pitch” y “yaw” en función de las velocidades angulares del sistema de referencia móvil:

$$J^{-1} = \begin{pmatrix} 0 & \sin \psi & \cos \psi \\ 0 & \cos \theta \cos \psi & -\cos \theta \sin \psi \\ \cos \theta & \sin \psi \sin \theta & \cos \psi \sin \theta \end{pmatrix}$$

*Ecuación 12. Inversa de la matriz jacobiana*

Esta inversa del Jacobiano nos va a ser muy útil para la resolución del modelo dinámico, ya que suponiendo conocido los ángulos de “roll”, “pitch” y “yaw” de la iteración anterior, podemos aplicar la ecuación para obtener las velocidades angulares de estos:

$$\left( \frac{d\vec{n}}{dt} \right)_1 = J^{-1}(\vec{\Omega})_1$$

*Ecuación 13. Velocidad angular de los ángulos de actitud en función de la velocidad angular del sistema de ref. móvil*

Dónde  $\vec{n} = (\varphi \ \theta \ \psi)$ , representa la orientación del dron respecto a el sistema de referencia móvil ligado a su CdM.

## 2.2 Vector estado

El vector estado se compone de  $\vec{p} = (X \ Y \ Z)$  coordenadas cartesianas del CdM en el sistema de referencia fijo y de  $\vec{n} = (\varphi \ \theta \ \psi)$  vector que determina la orientación del dron en el sistema de referencia móvil.

Por lo tanto, conociendo este vector en todo momento se conoce la orientación y posición del vehículo en cada iteración. Este vector será retroalimentado a el modelo dinámico, es decir será tanto salida como entrada del modelo, siendo la salida de una iteración la entrada de la siguiente, por lo tanto, es muy importante conocer en todo momento estas 6 variables.

$$\overrightarrow{estado} = (\vec{p}_0 \ \vec{n}_1) = ((X \ Y \ Z)_0 \ (\varphi \ \theta \ \psi)_1)$$

*Ecuación 14. Componentes vector estado*

Esta retroalimentación es muy importante para la simulación ya que permite integrar el movimiento haciendo que el valor integrado del vector estado de una iteración sean las condiciones iniciales de la siguiente. Esto implica que en el instante t=0 se tengan que definir unas condiciones iniciales para la primera iteración de la simulación, estas condiciones iniciales son definidas en la Ecuación 15, lo que indicaría que en ese instante el vehículo se encuentra en el origen de coordenadas del sistema de referencia fijo que coincide con el sistema de referencia móvil.

$$\overrightarrow{estado} = \vec{0} = ((0 \ 0 \ 0)_0 \ (0 \ 0 \ 0)_1)$$

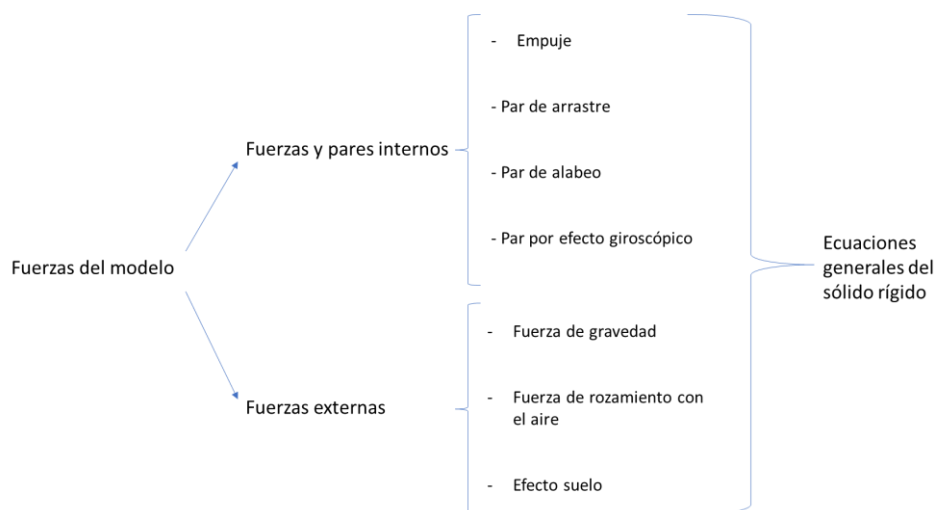
*Ecuación 15. Valor inicial del vector estado*

### 2.3 Fuerzas y momentos que actúan sobre el modelo

En este apartado, se presentan las principales expresiones analíticas de las fuerzas que gobiernan el modelo.

Vamos a distinguir principalmente las fuerzas según su origen, es decir, si son causadas propiamente por el vehículo: fuerzas y pares internos, o si su origen viene determinado por el entorno: fuerzas externas.

Las fuerzas que actúan sobre el vehículo simulado serán las que aparecen en la Figura 6. En los siguientes apartados se analizará el origen físico de cada fuerza o par, y se determinará la formula analítica utilizada para su modelización.



*Figura 6. Esquema de fuerzas presentes en el modelo multirroto*

### 2.3.1 Fuerzas y pares internos

Como se ha comentado estas fuerzas y pares serán consecuencia de la propia naturaleza del vehículo y en el caso particular del multirrotor son generados por la rotación de las hélices de cada uno de los rotores de los que dispone.

Las fuerzas y pares internos de mayor orden de magnitud presentes en la dinámica de este tipo de vehículos que se van a considerar en nuestro modelo, son principalmente cuatro:

1. Thrust o empuje
2. Par de arrastre
3. Par de alabeo y cabeceo
4. Par por efecto giroscópico

#### 2.3.1.1 Thrust $T$

La fuerza de empuje o fuerza vertical proporcionada por cada rotor es la responsable de la sustentación del dron. La dirección de esta fuerza va a seguir el eje de rotación del rotor, por lo que idealmente seguirá la dirección Z del sistema de referencia móvil {1}, en sentido negativo:

$$(0 \quad 0 \quad -T)_1$$

Esta magnitud comúnmente se suele expresar según indica la Ecuación 16.

$$T = C_t \rho A r^2 w^2$$

*Ecuación 16. Empuje de una hélice*

Dónde:

- $C_t$  – Coeficiente de empuje [adimensional]
- $\rho$  – Densidad de la atmósfera [ $kg/m^3$ ]
- $A$  – Area de referencia [ $m^2$ ]
- $r$  – Radio de la hélice [m]
- $w$  – Velocidad angular de giro del rotor [rad/seg]

El coeficiente de sustentación va a depender exclusivamente de la geometría y dimensiones de la hélice instalada en el rotor. Este se suele obtener experimentalmente mediante ensayos dónde se conocen las condiciones ambientales, las características geométricas de la hélice y la velocidad de rotación del rotor, así como el empuje que genera en todo momento.

La densidad dependerá de la atmósfera en la que vuela el multirrotor, esta recoge la influencia de las propiedades meteorológicas en el vuelo. Este valor de densidad va a ser dependiente de la zona del mundo donde vuele el vehículo, y también va a ser variable si las condiciones meteorológicas cambian durante el vuelo.

Otro motivo por el que puede variar la densidad del aire, es por el cambio de altura. En drones con alta capacidad de elevación sería conveniente un modelo de atmósfera preciso donde se proporcione un valor de densidad realista en función de la altura a la que vuela el dron, sin embargo, para este proyecto no hemos creído conveniente este enfoque ya que el rango de elevación de este dron no queremos que supere los 120 metros.

Según el modelo de atmósfera estándar internacional recogido en [7], para un día estándar sin incremento de temperatura se recoge una densidad de 1,225 kg/m<sup>3</sup> a el nivel del mar y de 1,21095 kg/m<sup>3</sup> a una altitud de 120 m. Esto representa un decremento de menos del 1.15% de su valor, en todo el rango de elevación, por este motivo se ha considerado irrelevante y se ha descartado su cálculo en el modelo.

Por lo tanto, ante la hipótesis de densidad constante durante todo el vuelo podemos asumir,  $\rho \cdot A \cdot r = \text{constante}$ , sin dejar de ser representativos de la realidad, lo que permite representar el empuje del rotor exclusivamente en función de la velocidad angular de rotación de este y de un coeficiente ahora con dimensiones, tal y como indica la Ecuación 17.

$$T = c_t w^2$$

*Ecuación 17. Empuje de una hélice asumiendo densidad constante*

Al igual que ocurre con el coeficiente de empuje adimensional este valor de  $c_t$  [kg m] puede obtenerse de manera experimental simplemente conociendo el empuje que proporciona la hélice a una determinada velocidad angular de giro, o bien se puede determinar analíticamente en función de los parámetros geométricos y dimensionales de la hélice.

#### 2.3.1.1.1 Método Teórico: Caracterización geométrica de la hélice:

A continuación, se presenta un método teórico expuesto en [8], donde se proponen la siguiente expresión para el coeficiente adimensional  $C_T$ .

$$C_T = \sigma a \left[ \left( \frac{1}{6} + \frac{1}{4} \mu^2 \right) \theta_0 - (1 + \mu^2) \frac{\theta_{tw}}{8} - \frac{\lambda}{4} \right]$$

*Ecuación 18. Coeficiente adimensional de empuje [Fuente: [8]]*

Esta expresión viene determinada por los diferentes parámetros geométricos de la hélice acoplada al motor.

Dónde cada parámetro de la Ecuación 18 indica:

$\sigma = \text{Ratio de solidez}$

$a = \text{Pendiente de sustentación}$

$\mu = \text{ratio de avance}$

$\lambda = \text{Ratio de flujo entrante}$

$\theta_0 = \text{Ángulo de inicio de la hélice}$

$\theta_{tw} = \text{Ángulo de curvatura}$

Con este coeficiente teórico y conociendo el valor de la densidad de la atmósfera, el radio de la hélice, el área de referencia y la velocidad angular de giro, podemos determinar la fuerza de empuje de cada uno de los rotores.

Las ecuaciones presentadas son solo un ejemplo válido, ya que hay otras ecuaciones más complejas que también pueden ser usadas. La existencia de varias ecuaciones para el mismo coeficiente se debe a las simplificaciones y estimaciones utilizadas en su cálculo.

También es importante tener en cuenta que las hélices experimentan diferentes estados de flujo según la dirección y orientación del dron, así como las condiciones del viento, lo que puede afectar su empuje.

Por lo tanto, cualquier método analítico va a tener sus limitaciones en la precisión de cálculo, cuantas más simplificaciones y suposiciones asuma el método más impreciso serán los resultados. Para conseguir una solución más cercana a la realidad deberíamos recurrir a métodos más precisos como pueden ser métodos CFD (Computational Fluids Dynamics).

### **2.3.1.2 Par de arrastre $Q$**

Este momento característico de hélices en rotación, es generado por la fuerza de resistencia a el avance de las palas. Por la tercera ley de Newton, toda fuerza ejercida sobre un medio físico tiene una reacción igual y opuesta, por lo tanto, una hélice rotando a gran velocidad en un sentido determinado, generará un momento de igual valor y sentido contrario en el fuselaje del vehículo.

El caso más típico de este fenómeno se puede apreciar en un helicóptero, dónde debido a sus hélices de gran tamaño es necesario un rotor de cola para compensar el momento de reacción generado en el fuselaje.

En el caso de un multirrotor, este par de arrastre es más pequeño debido a el tamaño de sus hélices, pero nunca se asumirá despreciable, es más, este momento será el responsable de la capacidad de guiñada del dron.

En conclusión, las hélice al rotar genera en su plano horizontal una fuerza de rozamiento con el aire que a su vez produce un momento  $Q$  en la dirección  $Z\{1\}$  del sistema de referencia móvil, es decir,  $(0 \ 0 \ \pm Q)_1$ , siendo positivo o negativo en función del sentido de giro de la hélice.

La magnitud de la fuerza  $Q$  la podemos expresar en función de la densidad, el área de referencia, el radio de la hélice y la velocidad angular de rotación de la hélice, según la Ecuación 19.

$$Q = -C_q \rho A r^3 \omega^2$$

*Ecuación 19. Par de arrastre de una hélice*

Dónde  $C_q$  es el coeficiente adimensional del par de arrastre.

El hecho de obtener un sentido u otro en función del eje de giro del rotor, y suponiendo que la única fuerza que va a generar momento en la dirección  $Z\{1\}$  sea esta fuerza de resistencia a el avance de la hélice, va a ser una característica relevante para permitir al dron controlar su guiñada.

Normalmente durante una operación nominal, los rotores del dron giran todos a la misma velocidad, lo que generará la misma resistencia en las palas y por consiguiente el mismo par de arrastre,  $Q$ . Si tienes a todas las hélices generando el mismo momento, basta con desacoplar dos a dos, en el caso de un cuadricóptero, tres a tres, en el hexacóptero o cuatro a cuatro, en el caso de un octocóptero, tal y como muestra la Figura 7, el sentido de giro de los rotores, para que la resultante de la suma de todos los pares rotores se anule, generando una situación estacionaria en el giro del dron sobre  $Z \{1\}$ .



Figura 7. Desacoplamiento velocidad angular de los rotores de un multirroto [Fuente: [9]]

Por el contrario, si lo que se desea es que gire el dron en la dirección  $Z\{1\}$ , bastará con modificar estas velocidades de giro de los rotores haciendo que dejen de ser iguales, hasta que se generado el giro deseado, donde se vuelve a solicitar a el control la condición de equilibrio en guiñada, es decir todas la velocidades angulares iguales lo que produce un  $\sum Q = 0$ .

De nuevo asumiendo la hipótesis,  $\rho \cdot A \cdot r = \text{constante}$ , se puede obtener la expresión de este par de arrastre en función únicamente de un coeficiente de dimensiones  $\text{kg} \cdot \text{m}$ , obtenido en nuestro caso de manera teórica, y de la velocidad angular de la hélice en rotación.

$$Q = -c_q \omega^2$$

Ecuación 20. Par de arrastre de una hélice asumiendo densidad constante

#### 2.3.1.2.1 Método Teórico – Caracterización geométrica de la hélice

Siguiendo el mismo método expuesto en [8], se propone la siguiente expresión para el coeficiente adimensional  $C_Q$ .

$$C_Q = \sigma a \left[ \left( \frac{1}{8a} (1 + \mu^2) \right) C_d - \lambda \left( \frac{1}{6} \theta_0 - \frac{1}{8} \theta_{tw} - \frac{1}{4} \lambda \right) \right]$$

Ecuación 21. Coeficiente adimensional de para de arrastre [Fuente: [8]]

Donde los parámetros son los mismos que los recogidos en el apartado 2.3.1.1.1.

Conociendo el valor de la densidad del entorno, del radio de la hélice, el área de referencia y la velocidad angular de giro, podemos determinar el momento generado en cada uno de los rotores por la fuerza de resistencia, también conocido como par de arrastre.

### 2.3.1.3 Par de alabeo y cabeceo

Momento generado en el plano horizontal producido por la fuerza de empuje que aparece en cada rotor, consecuencia de la distancia de este al centro de masas del vehículo, por lo tanto, se puede obtener multiplicando vectorialmente la fuerza de empuje de cada rotor por la distancia de este al CdM del dron, tal y como indica la Ecuación 22.

$$\vec{\tau}_{xy} = \vec{d}_i \times \vec{T}_i$$

*Ecuación 22. Par de alabeo y cabeceo producido por el empuje de los rotores*

Como el empuje idealmente seguirá la dirección  $(0 \ 0 \ -T)_1$  el par de alabeo y cabeceo va a estar contenido en plano XY del sistema de referencia móvil {1}, por lo que sus componentes serán:  $(\tau_x \ \tau_y \ 0)_1$

Este momento será el responsable de la inclinación del dron respecto al plano horizontal, es decir, será el responsable de inducir ángulos de “roll” y “pitch” en el vehículo durante el vuelo. La posición de equilibrio en el plano horizontal se obtendrá cuando este vector  $\tau_{xy} = \vec{0}$ , siendo la componente  $\tau_x$  responsable de inducir “roll” y la componente  $\tau_y$  la responsable de inducir “pitch”.

### 2.3.1.4 Efecto giroscópico

El efecto giroscópico es la capacidad que tienen los objetos en rotación de mantener una dirección constante en su eje de giro.

Se trata de un efecto estabilizador ya que cuando un objeto en rotación experimenta una fuerza que trata de desviarlo de su eje de rotación, el efecto giroscópico genera una resistencia opuesta a ese cambio y mantiene el eje de giro en su posición original.

Este fenómeno se basa en el principio físico del momento angular y su conservación. Cuando un objeto está girando, su momento angular se mantiene constante a menos que se apliquen fuerzas externas.

Por lo tanto, este efecto se produce en cada brazo del dron, lo que genera un momento perpendicular al eje de giro del rotor, es decir, en el plano horizontal del sistema de referencia móvil {1}:  $(\tau'_x \ \tau'_y \ 0)_1$

Este momento generado por la perturbación se puede expresar según la Ecuación 23 y dependerá de la velocidad angular horizontal del sistema móvil en ese instante,  $\vec{\Omega}_{xy} = (\Omega_x \ \Omega_y)$ , de la velocidad angular y del rotor de inercia de las hélices,  $I_r$ .

$$\tau'_x = -I_r \Omega_y \sum w_i$$
$$\tau'_y = I_r \Omega_x \sum w_i$$

*Ecuación 23. Par de alabeo y cabeceo producido por el efecto giroscópico*

Por lo tanto, cuando se intenta alterar la dirección del eje de giro de un objeto en rotación, el efecto giroscópico produce una fuerza contraria que se opone al cambio, logrando así estabilizar la dirección del eje.

Teniendo en cuenta que el sentido de giro de los n-rotores del dron están desacoplados, se deduce de la Ecuación 23 que este efecto desaparecerá cuando todos los rotores giren a la misma velocidad angular o cuando no exista rotación relativa del sistema de referencia móvil respecto al fijo:  $\vec{\Omega}_{xy} = \vec{0}$

### 2.3.2 Fuerzas externas

A continuación, se presentan las fuerzas inducidas por influencia del entorno sobre el vehículo. Entre todas las posibles fuerzas externas presentes en la naturaleza, hemos decidido incluir en el modelo tres de las más influyentes para el caso particular de un dron:

1. Fuerza de gravedad
2. Incremento de empuje por el efecto suelo
3. Fuerza de fricción con el aire

#### 2.3.2.1 Fuerza de gravedad

Para nuestro modelo se va a asumir que el valor de la gravedad es constante a lo largo de toda la simulación, es decir no cambia con la altura. Fijaremos su valor en  $9.81 \text{ m/s}^2$ , aunque al ser un modelo parametrizado será fácilmente modificable en el constructor de cada vehículo.

La dirección de esta magnitud será la dirección Z {0} positiva del sistema de referencia fijo. Por lo tanto, conociendo la masa, M, del dron es constante en todo el vuelo, podemos concluir que el peso,  $\vec{P}$ , del dron será en todo instante de la simulación igual a  $\vec{P} = (0 \ 0 \ M \cdot g)_0$

#### 2.3.2.2 Fuerza de fricción con el aire

La fuerza de rozamiento con el aire, también conocida como resistencia aerodinámica, se produce debido a la interacción entre el vehículo en movimiento y las moléculas de aire circundantes.

Este fenómeno físico está condicionado por los siguientes fenómenos físicos o características del vehículo en movimiento:

1. Colisiones y transferencia de momento: A medida que el vehículo se mueve, choca con las moléculas de aire. Estas colisiones provocan una transferencia de momento entre el objeto y las moléculas de aire, lo que genera una fuerza opuesta al avance del vehículo.
2. Arrastre viscoso: El aire se comporta como un fluido viscoso, lo que significa que tiene una resistencia interna al flujo. Cuando el objeto se mueve a través del aire, las capas de aire cercanas a su superficie se adhieren al objeto y forman una "capa límite" de aire en movimiento. Esta capa límite crea una resistencia adicional al movimiento del objeto.



3. Forma del objeto: La forma y el perfil del objeto también influyen en la fuerza de rozamiento con el aire. Objetos con formas aerodinámicas, como las gotas de agua, experimentan menos resistencia debido a que el aire puede fluir más suavemente alrededor de ellos. Por otro lado, objetos con formas más angulosas o irregulares generan flujos turbulentos y una mayor resistencia.
4. Velocidad del objeto: La fuerza de rozamiento con el aire aumenta con la velocidad del objeto. A medida que la velocidad aumenta, más moléculas de aire chocan con el objeto en un período de tiempo dado, lo que resulta en una mayor resistencia.

En aerodinámica la fuerza de rozamiento con el aire se expresa comúnmente según la Ecuación 24.

$$F_{roz} = C_{roz} \rho A v_{rel}^2$$

*Ecuación 24. Fuerza de rozamiento con el aire*

Donde  $\rho$  es la densidad del aire,  $A$  el área de referencia respecto del cual se calcula el coeficiente adimensional,  $C_{roz}$  es el coeficiente adimensional obtenido normalmente de manera experimental y  $v_{rel}$  es el módulo de la velocidad relativa del aire en el sistema de referencia móvil {1}.

La dirección de esta fuerza será opuesta al movimiento del vehículo, por lo que llevará la misma dirección que la velocidad relativa del dron, pero sentido opuesto.

Respecto a el cálculo de  $C_{roz}$  cabe destacar que existen métodos no empíricos mucho más precisos para obtener este coeficiente, normalmente, haciendo software de computación que resuelven la mecánica de fluidos de un caso concreto.

En nuestro modelo por falta de los medios necesarios para obtenerlo experimentalmente y porque el uso de software CFD quedaba fuera de los límites de este trabajo se ha decidido estimar razonablemente este coeficiente de resistencia aerodinámica basándonos en el coeficiente de rozamiento utilizado para el vehículo cuadricóptero por los antiguos desarrolladores del modelo.

#### 2.3.2.2.1 Velocidad del aire variable

Debido a que la velocidad relativa se define como indica la Ecuación 25, se ha considerado oportuno añadir la posibilidad de modificar la velocidad horizontal del viento a través de  $V_x$  y  $V_y$ .

$$\vec{v}_{rel} = \vec{v}_{aire} - \vec{v}_{dron}$$

*Ecuación 25. Velocidad relativa del vehículo*

De la Ecuación 25 podemos deducir que, si el dron avanza en el mismo sentido que el aire y a la misma velocidad, la velocidad relativa será nula y no existirá fuerza de rozamiento, por el contrario, si el dron avanza en sentido opuesto al aire teniendo el aire de cara, esta velocidad

relativa será superior a la velocidad de dron ya que se le sumará la del aire al tener ambas velocidades sentidos opuestos.

### ***2.3.2.3 Incremento de empuje por efecto suelo***

El efecto suelo en un multirrotoir puede tener un impacto significativo en su rendimiento y comportamiento dinámico.

Este efecto se refiere a la interacción entre la superficie sobre la cual vuela el multirrotoir y las corrientes de aire generadas por sus hélices. Cuando un dron vuela cerca del suelo, se forma un espacio de aire comprimido entre el vehículo y la superficie, lo que crea una especie de "almohadilla" de aire.

A continuación, se presentan dos consecuencias, una positiva y otra negativa del efecto suelo en drones.

1. Incremento de fuerza de sustentación: El efecto suelo aumenta la sustentación del dron, ya que la presión del aire bajo este en estas condiciones será superior lo que ayuda a contrarrestar la fuerza de la gravedad y proporcionar un mayor soporte.  
  
Esto permitirá al dron volar de manera más eficiente, debido a que necesitas una menor velocidad angular de rotación las hélices para estabilizar el vuelo.
2. Influencia en el control y la maniobrabilidad: El efecto suelo puede afectar la respuesta y la estabilidad del dron. En algunos casos, puede hacer que el dron sea más sensible a los cambios en los controles y más propenso a oscilaciones o vibraciones no deseadas.  
  
En situaciones extremas, el efecto suelo puede dar lugar a oscilaciones inestables conocidas como "ground resonance" (resonancia del suelo) en drones de gran tamaño. Estas oscilaciones pueden ser peligrosas y potencialmente dañinas para el dron y su estructura. Por lo tanto, es importante que los pilotos de drones tengan en cuenta estos efectos y ajusten su vuelo en consecuencia.

En resumen, el efecto suelo puede tener tanto efectos positivos como negativos en el vuelo de un dron. Es importante comprender cómo afecta el rendimiento del dron y adaptar las técnicas de vuelo en consecuencia para garantizar un vuelo seguro y controlado.

En nuestro modelo, solo se considerará el incremento de empuje, descartando el estudio de las inestabilidades producidas por este efecto. Este incremento de empuje solo se considerará relevante cuando la altura del dron respecto a la superficie sólida más cercana sea inferior a 10 veces el radio de la hélice, siendo despreciable a alturas superiores.

Este incremento se modela mediante la Ecuación 26, donde  $T_{libre}$  es el empuje ideal determinado analíticamente con la Ecuación 16.

$$\frac{T_{con\ suelo}}{T_{sin\ suelo}} = \frac{1}{1 - \frac{r^2}{16 z^2}}$$

*Ecuación 26. Incremento de empuje por efecto suelo en función de la altitud del vehículo*

## 2.4 Modelo dinámico de 6 grados de libertad

El modelo dinámico planteado, el cual se va a resolver en la simulación tendrá 6 grados de libertad. Una vez definidas el origen, las hipótesis y la magnitud de cada una de las fuerzas que determinan el movimiento de nuestro vehículo podemos plantear las ecuaciones generales del sólido rígido que serán utilizadas en Simulink para obtener el vector estado en cada iteración.

### 2.4.1 Hipótesis ecuaciones generales del sólido rígido

Las ecuaciones generales del sólido rígido están basadas en algunas hipótesis fundamentales que simplifican el comportamiento dinámico de los cuerpos sólidos.

A continuación, se presentan las principales hipótesis asociadas a estas ecuaciones:

1. Rigidez del vehículo: Se asume que el cuerpo sólido es completamente rígido, es decir, que no existen deformaciones internas ni cambios en la forma del objeto durante el movimiento. Esta hipótesis implica que las distancias entre los puntos del sólido se mantienen constantes a lo largo del tiempo.
2. Partículas puntuales: Se considera que el sólido está formado por una colección de partículas puntuales, es decir, se ignora el tamaño y la extensión física real de las partículas individuales. Esta simplificación permite tratar el cuerpo sólido como un sistema discreto de masas concentradas en puntos específicos.

En nuestro caso para resolver estas ecuaciones se va suponer que todo el dron se reduce a una masa puntual de masa  $M$ , localizado en su centro de masas, origen del sistema de referencia móvil  $\{1\}$ .

3. Conservación del momento angular,  $\vec{L} = cte$ : Se supone que el momento angular total del sólido rígido se conserva en ausencia de fuerzas externas. Esto implica que no hay momentos externos aplicados al sistema,  $\vec{M}_{ext} = 0$  o que los momentos aplicados son equilibrados por momentos internos dentro del sólido, ya que  $\frac{d\vec{L}}{dt} = \vec{M}_{ext}$

Estas hipótesis son fundamentales para derivar las ecuaciones generales del sólido rígido, como las ecuaciones del movimiento lineal y angular, sin embargo, es importante tener en cuenta que, en ciertas situaciones, algunos de estos supuestos pueden no ser válidos, y se requieren modelos más complejos para describir de una manera realista por ejemplo el comportamiento de cuerpos sólidos deformables o con movimientos más complejos.

Por lo tanto, después de definir las hipótesis asumidas por el modelo de sólido rígido vamos a hacer un repaso de las ecuaciones diferenciando entre la resolución del movimiento lineal, con el que obtendremos el vector  $\vec{p} = (X \ Y \ Z)$ , y entre el movimiento angular con el que

obtendremos el vector  $\vec{n} = (\varphi \ \theta \ \psi)$ , con ambos vectores queda determinado el vector estado en cada iteración.

### 2.4.2 Movimiento lineal

Durante el movimiento lineal o de traslación de un sólido, todos los puntos del objeto se desplazan en trayectorias paralelas. Además, la velocidad de cualquiera de estos puntos durante el movimiento de traslación es idéntica a la velocidad del centro de masas del sólido. Esto implica que todos los puntos del objeto se desplazan a la misma velocidad y en la misma dirección que el centro de masas, sin importar su posición relativa dentro del sólido.

La ecuación que describe este movimiento lineal será derivada de la segunda ley de Newton, donde para nuestro caso particular tendremos:

$$\left(\frac{d\vec{v}}{dt}\right)_0 = \frac{1}{M} \left( (\vec{F})_0 + iR (\vec{T})_1 \right)$$

*Ecuación 27. Ecuación diferencial del movimiento lineal*

Donde  $\vec{v}$ , representa la velocidad absoluta del dron en ejes fijos,  $\vec{F}$  representa las fuerzas externas comentadas anteriormente,  $M$  es la masa del dron y  $\vec{T}\{1\}$  es la suma de las fuerzas de empuje de cada rotor en el sistema de referencia móvil, multiplicado por  $iR$ , inversa de la matriz de rotación descrita en el apartado 2.1.1.1, obtenemos esta magnitud en el sistema de referencia fijo  $\{0\}$ .

Integrando obtenemos la velocidad absoluta del dron en ejes fijos  $\{0\}$ :  $\vec{v}$

$$\left(\frac{d\vec{p}}{dt}\right)_0 = \vec{v}$$

*Ecuación 28. Ecuación diferencial de la velocidad*

Según la Ecuación 28 integrando la velocidad obtenemos las coordenadas cartesianas en ejes fijos:  $\vec{p} = (X \ Y \ Z)_0$ , por lo tanto, con solo estas dos ecuaciones diferenciales definimos el movimiento de traslación del dron en cada iteración.

### 2.4.3 Movimiento angular

Las ecuaciones de movimiento angular del sólido rígido describen cómo cambia el movimiento de rotación de un objeto en función de las fuerzas que actúan sobre él. Estas ecuaciones se derivan a partir de las ecuaciones generales del sólido rígido, que describen el movimiento del objeto como una combinación de traslación y rotación.

Asumiendo las hipótesis planteadas al principio de esta sección, podemos decir que, debido a la segunda hipótesis la gravedad afecta a todas las partículas del dron por igual y gracias a la tercera se acepta que el entorno no genera momentos exteriores que inducen sobre el dron rotaciones adicionales.

Por último, un diseño simétrico del vehículo implica que el viento genera una fuerza uniforme sobre todo el vehículo, lo que tampoco induce momento externo.

Con todas estas suposiciones es válido afirmar que los únicos momentos que va a experimentar el dron son los comentados en el apartado 2.3.1, es decir el par de arrastre,  $(0 \ 0 \ Q)_1$ , el par de alabeo y cabeceo,  $(\tau_x \ \tau_y \ 0)_1$  y el par por efecto giroscópico:  $(\tau'_x \ \tau'_y \ 0)_1$ , todos ellos expresados en el sistema de referencia móvil.

Por definición, la variación de la cantidad de momento angular  $\vec{L}$  se puede expresar como la suma total de los momentos que experimenta el vehículo en movimiento, hecho que se puede expresar según la Ecuación 29.

$$\frac{d\vec{L}}{dt} = \frac{d(I_0\vec{\Omega})}{dt} = \sum \vec{\tau}$$

*Ecuación 29. Variación del momento angular*

Donde,  $\vec{\tau}$  representa el conjunto de momentos que actúan sobre el dron,  $\vec{\Omega}$  es la velocidad angular del sistema de referencia móvil {1} e  $I_0$ , es la matriz de inercia del cuerpo expresada en el sistema de referencia fijo {0}, esta matriz de inercia será desconocida ya que es variable con la posición angular del vehículo y por lo tanto cambiará con el tiempo, sin embargo, si conoceremos la matriz de inercia  $I_1$ , en el sistema de referencia móvil {1}, constante en el tiempo.

En términos generales, la Ecuación 29 establece que la aceleración angular,  $\alpha = \frac{d(\vec{\Omega})}{dt}$ , del sólido rígido es igual a la suma de los momentos o pares externos que actúan sobre él, dividida por su momento de inercia.

Esta matriz de inercia las obtenemos de cualquier software CAD, en este proyecto se ha utilizado Solid Edge, pero existen otras alternativas comerciales como CATIA o SolidWorks. Para los dos nuevos vehículos incorporados en el modelo hemos obtenido mediante este software las matrices de inercia respecto a el sistema de referencia móvil {1}.

La cantidad de movimiento angular  $I_0\vec{\Omega}$  respecto a la base fija {0}, vendría determinada por la Ecuación 30.

$$\left(\frac{d(I_0\vec{\Omega})}{dt}\right)_0 = \left(\frac{d(I_1\vec{\Omega})}{dt} + \vec{\Omega} \times I_1\vec{\Omega}\right)_1$$

*Ecuación 30. Cantidad de momento angular*

Incorporando la Ecuación 29 a este desarrollo obtenemos la Ecuación 31 expresada en el sistema de referencia móvil {1}:

$$(\vec{\tau})_1 = \left(\frac{d(I_1\vec{\Omega})}{dt} + \vec{\Omega} \times I_1\vec{\Omega}\right)_1$$

*Ecuación 31. Momentos que actúan sobre el multirroto*

Despejando  $\frac{d(\vec{\Omega})}{dt}$  obtenemos la Ecuación 32, que nos permite obtener la orientación del vehículo.

$$\left(\frac{d(\vec{\Omega})}{dt}\right)_1 = I_1^{-1}((\vec{\tau})_1 - \vec{\Omega} \times I_1 \vec{\Omega})_1$$

*Ecuación 32. Variación de la velocidad angular del sistema referencia móvil*

Asumiendo conocidos todos los parámetros de la Ecuación 32, es decir, la matriz de inercia en ejes móviles  $I_1$ , el conjunto de momento que actúan sobre el vehículo  $\vec{\tau}$ , y la velocidad angular del sistema de referencia móvil {1} en la iteración anterior, se puede integrar esta ecuación para obtener la velocidad angular de la base relativa en esa iteración.

Conociendo la matriz Jacobiana y haciendo uso de la Ecuación 13 se obtiene la velocidad angular de los ejes intermedios en la base móvil {1}, lo que corresponde a las aceleraciones angulares de los ángulos “roll”, “pitch” y “yaw”.

Integrando este vector obtenemos los ángulos “roll”, “pitch” y “yaw” obteniéndose así la orientación del vehículo respecto a el sistema de referencia fijo y que junto a las coordenadas cartesianas en estos ejes determinan el vector estado en esa iteración de la simulación, solución de nuestro modelo dinámico.

## Chapter 3. Control del vehículo

Después de la descripción matemática del modelo de fuerzas que gobiernan el movimiento del vehículo es necesario implementar algoritmos de control para manejarlo adecuadamente en vuelo. Estos algoritmos actúan como intermediarios entre los motores y las órdenes de vuelo recibidas, ya sean comandadas por el piloto a través del mando o mediante cualquier sistema de autopiloto.

En secciones anteriores se ha descrito el comportamiento del multirroto desarrollando un modelo matemático parametrizado y se han presentado todas las fuerzas que gobiernan dicho modelo. El propósito de esta descripción es mostrar una forma precisa de recrear su realidad con el objetivo final de simular el comportamiento dinámico del vehículo dentro de un entorno virtual.

Los algoritmos de control que se presentarán a continuación son implementados en el dron real mediante diversos dispositivos electrónicos por lo que existen múltiples maneras de implementarlos.

En una primera aproximación, se pueden distinguir varios ejemplos de implementación de control. Uno de ellos es el uso de una placa programable sencilla como Arduino o Raspberry Pi, que reciben las órdenes de control y retroalimentación de la Unidad de Medición Inercial (IMU) y envían órdenes a los motores BLDC. Sin embargo, en la práctica, los drones de uso industrial suelen incorporar un microcontrolador dedicado que incluye los algoritmos de control. La programación de estos algoritmos puede variar según el modelo y el modo de uso del dron.



Existen diferentes dispositivos esenciales para el control de actitud de un dron con el objetivo de mantener y controlar su orientación en el espacio. Algunos de los más comunes son:

1. IMU (Unidad de Medición Inercial): Este dispositivo combina varios sensores, como acelerómetros, giróscopos y magnetómetros, para medir la aceleración lineal, la velocidad angular y la orientación del dron en el espacio. Estos datos son retroalimentados al dispositivo de control y después son implementado en los diferentes algoritmos para ajustar los motores y mantener la actitud deseada.
2. Controladores PID (Proporcional-Integral-Derivativo): Los controladores PID son algoritmos de control clásicos ampliamente utilizados en drones. Estos controladores obtienen una señal de control en función del error entre la actitud deseada y la actitud real medida por la IMU. El controlador PID ajusta continuamente los motores para reducir este error y mantener la estabilidad y la orientación del dron solicitada por el control.
3. Controladores de retroalimentación de estado: Estos controladores utilizan técnicas más avanzadas, como la teoría de control moderna y el control óptimo, para controlar la actitud del dron. Estos controladores utilizan modelos matemáticos del sistema y mediciones en tiempo real para calcular y ajustar las señales de control de los motores. Estos controladores pueden ofrecer un rendimiento más preciso y adaptativo en comparación con los controladores PID. Entre ellos podemos destacar los controladores LQR (Linear Quadratic Regulator), los cuales se basan en técnicas de control óptimo y son especialmente efectivo cuando se cuenta con un modelo matemático preciso del sistema. El objetivo del control de actitud por LQR es diseñar un controlador lineal que minimice una función cuadrática que tiene en cuenta tanto el estado del sistema como las señales de control.
4. Algoritmos de control basados en aprendizaje automático: En los últimos años, se han utilizado técnicas de aprendizaje automático, como redes neuronales o algoritmos de aprendizaje por refuerzo, para desarrollar algoritmos de control de actitud más avanzados. Estos algoritmos pueden aprender de los datos recopilados durante el vuelo y ajustar su comportamiento de control en función de las condiciones y los objetivos específicos del vehículo.

Estos son solo algunos ejemplos de dispositivos de control de posición y actitud utilizados en los drones. La elección del dispositivo dependerá de la aplicación, el nivel de precisión requerido y las capacidades del dron en particular.

La descripción que se presenta a continuación se centrará únicamente en los aspectos matemáticos del control sin recrear la electrónica necesaria para llevar a cabo estos algoritmos, sin embargo, esto será suficiente para su la implementación en el simulador obteniéndose así resultados similares a los proporcionados por la electrónica real de un multirroto.

El control elegido para el modelo va a ser implementado en Simulink suponiendo que el multirrotores dispone un sistema de control de actitud PID y una IMU “perfecta” con datos actualizados en todo momento de la posición, velocidad, aceleración, orientación, etc.

Cabe destacar que esta condición de IMU “perfecta” es una situación completamente ideal, ya que un IMU real proporciona mediciones que necesitan ser filtradas debido a la incertidumbre de sus mediciones y el ruido asociado a cada uno de los sensores. Un filtro típico utilizado en este tipo de mediciones es el filtro de Kalman, este es un algoritmo de estimación utilizado para combinar de manera óptima las mediciones de una IMU (Unidad de Medición Inercial) con el fin de obtener una estimación más precisa del estado del sistema, como la actitud, la velocidad y la posición de un dron, este enfoque queda fuera de los límites del trabajo, pero sería interesante considerarlo como futuro desarrollo para el modelo.

Por lo tanto, en nuestra simulación estos datos no salen de ninguna IMU, sino que son la solución al modelo dinámico descrito en el apartado 2.4, el cual se va a resolver en cada iteración.

Con la solución del modelo tenemos la información precisa de la posición y actitud del dron, estas variables retroalimentan controladores los PID. Además de estas variables estos controladores reciben las variables objetivo impuestas por el “controller” o por el autopiloto.

Según diseño el controlador PID tenderá a reducir el error entre la variable objetivo y la variable solución del modelo dinámico. Cuando este error sea aproximadamente cero significará que el dron ha cumplido la orden descrita por el piloto y el PID deja de modificar la variable solución del modelo dinámico.

Este tipo de control se llama control en lazo cerrado ya que retroalimentas la variable que quieres modificar salida del modelo dinámico, tal y como muestra la Figura 8.

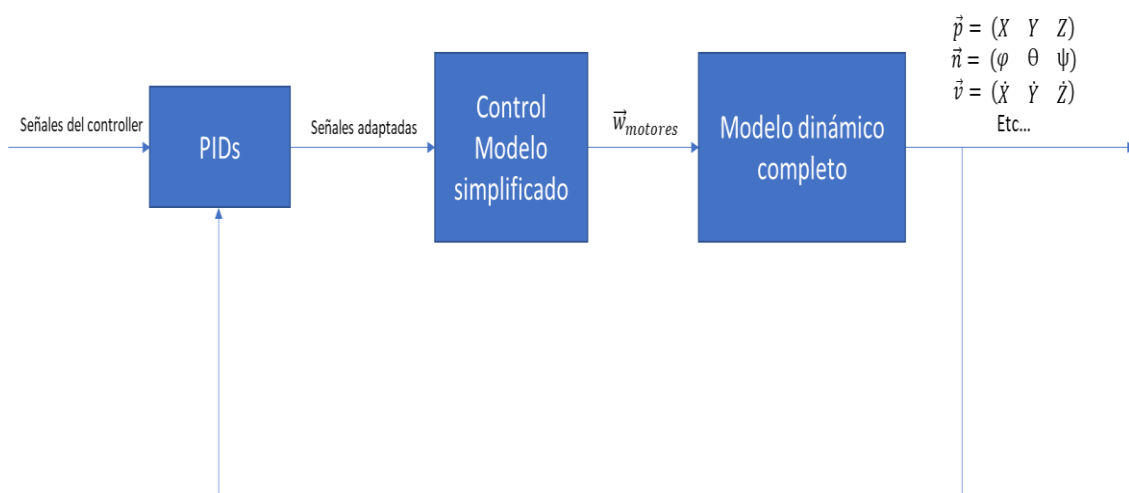


Figura 8. Esquema del control del modelo por lazo cerrado

El control de actitud del dron implementado seguirá tres bloques claramente diferenciados.



### 3.1 Controladores PID

Los controladores PID se compone de tres términos principales: proporcional (P), integral (I) y derivativo (D). Cada término contribuye de manera diferente al control del sistema y tiene un papel específico en la respuesta del controlador.

1. Término Proporcional (P): Este término es proporcional al error entre la variable controlada y el valor deseado,  $K_p \cdot E(t)$ . Cuanto mayor sea el error, mayor será la contribución proporcional al control. El término P proporciona una respuesta rápida del controlador y ayuda a reducir el error en estado estacionario. Sin embargo, puede llevar a oscilaciones y “overshooting” si se usa de manera excesiva.
2. Término Integral (I): El término integral acumula el error a lo largo del tiempo y lo utiliza para corregir el control,  $K_I \cdot \int E(t) dt$ . Ayuda a eliminar el error en estado estacionario y mejora la precisión del control. El término I es particularmente útil cuando el sistema tiene errores persistentes. Sin embargo, un exceso de término integral puede generar una respuesta lenta y provocar oscilaciones.
3. Término Derivativo (D): El término derivativo se basa en la tasa de cambio del error,  $K_D \frac{dE(t)}{dt}$ . Ayuda a predecir las tendencias futuras y estabiliza la respuesta del controlador. El término D es útil para reducir oscilaciones y mejorar la estabilidad. Sin embargo, puede amplificar el ruido en las mediciones, por lo que debe utilizarse con cuidado.

En resumen, los controladores PID utilizan una combinación de términos proporcional, integral y derivativo para lograr un control preciso y estable de un sistema. El término proporcional responde al error actual, el término integral elimina el error en estado estacionario y el término derivativo estabiliza la respuesta del controlador.

La combinación de los tres términos (P, I y D) permite un control más preciso y adaptable en diferentes situaciones, pudiendo prescindir uno o más obteniéndose así los controladores P, PI, PD etc.

Los coeficientes del controlador ( $K_p$ ,  $K_I$  y  $K_D$ ) se ajustan para lograr un rendimiento óptimo, considerando las características y requisitos del sistema controlado. La elección de las constantes PID depende de las características del sistema controlado y los requisitos de control específicos. En general, el ajuste de las constantes PID es un proceso iterativo que implica realizar pruebas y ajustes para lograr un rendimiento óptimo del controlador.

Es importante destacar que las constantes PID no son independientes entre sí y ajustar una constante puede afectar la respuesta de las otras. Por lo tanto, es necesario encontrar un equilibrio adecuado entre las constantes para lograr un control estable, preciso y eficiente del sistema controlado. El ajuste adecuado de los coeficientes del controlador es esencial para obtener un rendimiento óptimo del control del dispositivo.

El funcionamiento de un controlador PID puede esquematizarse según la Figura 9. En resumen, este funcionamiento consiste en la producción de la señal  $u(t)$  que entre al modelo para obtener la señal  $y(t)$  que haga que el error,  $e(t)$ , sea lo más próximo a cero posible.

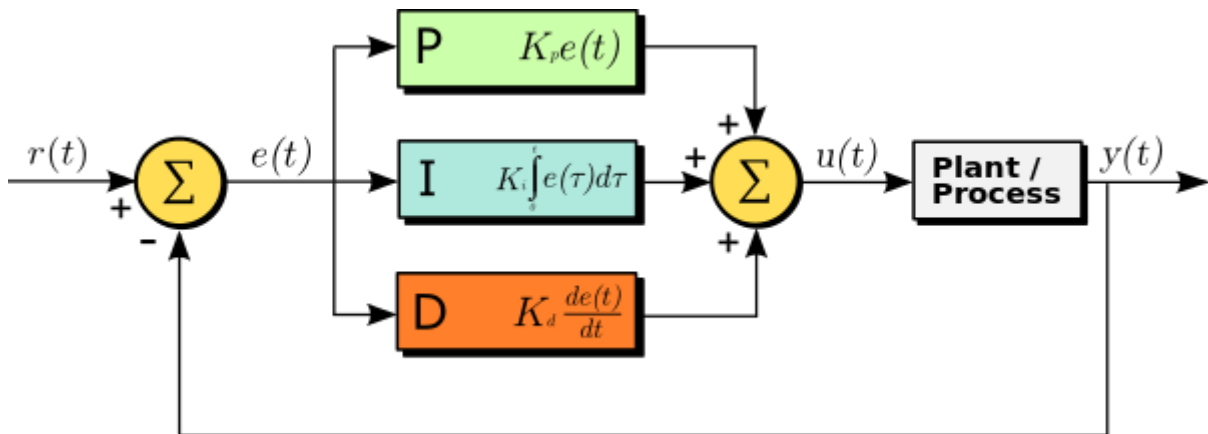


Figura 9. Esquema de un controlador PID genérico

Particularmente en nuestro bloque PID van a entrar las señales normalizadas sin amplificar del mando del piloto y saldrán las señales adaptadas a conseguir la variable deseada estipulada previamente gracias a las ganancias y a la actuación de los controladores PID.

Se han modificado iterativamente tanto las ganancias de las señales que vienen del mando del piloto cómo los coeficientes de los controladores PID hasta que se ha obtenido un control adecuado de los vehículos en vuelo.

### 3.2 Control mediante el modelo simplificado: Matriz de control

La matriz de control,  $A$ , va a ser la herramienta matemática utilizada para el resolver en cada iteración un modelo dinámico simplificado ante una orden impuesta por el bloque PID. Esta orden viene determinada por el vector  $\vec{H}$ .

A continuación, se recoge las hipótesis de este modelo dinámico simplificado para un multirroto generalizado de  $N$  motores, en el cual vamos a asumir las siguientes:

1. La fuerza de empuje de cada uno de los motores es la única fuerza que actúa sobre el vehículo y siempre irá en la dirección negativa  $Z \{1\}$ .
2. Los pares o momentos a los que se ve sometidos el vehículo son únicamente:
  - a. Par de arrastre  $Q$ , producido por la resistencia a el avance de la hélice siempre en la dirección del sistema de referencia móvil  $Z \{1\}$ .
  - b. Par de alabeo,  $\tau_x$ , producido por la fuerza de empuje, siempre en la dirección del sistema de referencia móvil  $X \{1\}$ .
  - c. Par de cabeceo,  $\tau_y$ , producido por la fuerza de empuje, siempre en la dirección del sistema de referencia móvil  $Y \{1\}$ .
3. Los coeficientes adimensionales  $c_t$  y  $c_q$  son constantes independientemente de la velocidad angular a la que gire el motor.
4. El dron es simétrico respecto de a los planos  $XZ$  e  $YZ$  del sistema de referencia móvil  $\{1\}$ , por lo que, todos los rotores están colocados a la misma distancia, vertical y horizontal, del centro de masas.
5. La densidad del entorno es constante en todo el vuelo.

Como conclusión a estas hipótesis se puede afirmar que este modelo dinámico simplificado se compone únicamente de una fuerza y tres momentos, de las que disponemos de sus expresiones analíticas, representadas en la Ecuación 35, Ecuación 36 y Ecuación 37.

Resulta conveniente construir una matriz de distancia de los n rotores a el centro de masas, obteniéndose así la matriz D:

$$D = \begin{pmatrix} d \cos \alpha_1 & d \cos \alpha_2 & \dots & d \cos \alpha_N \\ d \sin \alpha_1 & d \sin \alpha_2 & \dots & d \sin \alpha_N \\ h & h & \dots & h \end{pmatrix}$$

Ecuación 33. Matriz de distancias de los rotores del vehículo

Donde:

- d es la distancia horizontal del brazo del multirroto al centro de masas, igual para todos los rotores.
- h es la distancia vertical de la posición del multirroto al centro de masas, igual para todos los rotores
- $\alpha_i$ , es el ángulo entre el rotor localizado en la posición i con el eje X {1} del modelo. Puede calcularse según la Ecuación 34, dónde N es el número de rotores del vehículo y  $\alpha_0$  es el ángulo que se elige arbitrariamente para la localización del primer rotor.

$$\alpha_i = \alpha_0 + \frac{i - 1}{N} 360$$

Ecuación 34. Ángulo de localización del rotor respecto al eje X.

Por ejemplo, la Figura 10 representa la distribución de rotores en un hexacóptero, donde se ha elegido arbitrariamente un  $\alpha_0 = 20^\circ$ , por lo tanto, tendremos los siguientes ángulos para cada rotor:  $\alpha_1 = 20^\circ$ ,  $\alpha_2 = 80^\circ$ ,  $\alpha_3 = 140^\circ$ ,  $\alpha_4 = 200^\circ$ ,  $\alpha_5 = 260^\circ$  y  $\alpha_6 = 320^\circ$ .

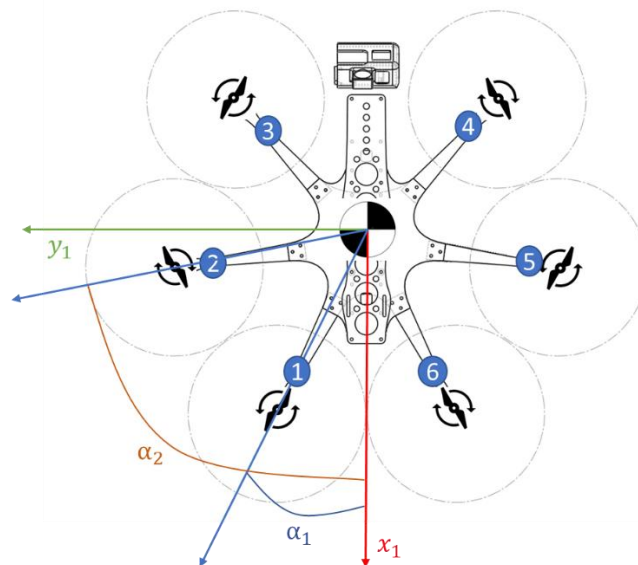


Figura 10. Distribución de rotores en un hexacóptero

En la Figura 11 se esquematizan las fuerzas y momentos, así como las magnitudes de las que dependen, es cierto que en el caso particular de la figura el rotor  $i$ , esta sobre el eje  $Y \{1\}$  por lo que  $\alpha_i = 90^\circ$  y el momento de cabeceo sería nulo, pero se ha decidido representar en la figura por mantener la generalización de la explicación.

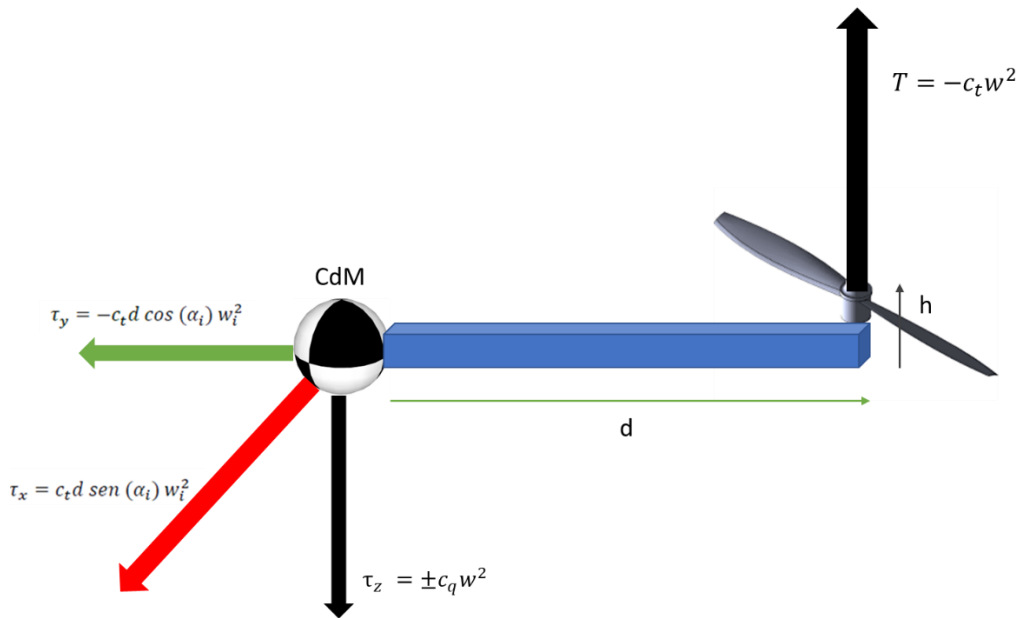


Figura 11. Esquema de fuerzas y momentos que ejerce un rotor

Donde se puede apreciar que el valor de los momentos de cabeceo y alabeo vienen condicionados por la distancia de la posición del rotor al centro de masas del dron, origen del sistema de referencia móvil.

Con todo esto podemos estimar las fuerzas y momentos producidas por los  $N$  motores del modelo simplificado, en función de valores constantes y de una única variable: La velocidad angular de los  $n$  motores.

El vector velocidad angular,  $W$ , será nuestra única variable de control, incógnita de nuestro sistema compatible resuelto en cada iteración, donde para los vehículos simulados el número de incógnitas del sistema siempre será mayor o igual a el número ecuaciones.

Este sistema se define mediante las la Ecuación 35, Ecuación 36 y Ecuación 37:

$$T = \sum c_{ti} w^2 = c_t \sum w^2$$

Ecuación 35. Empuje total que ejercen los rotores sobre el vehículo

$$\tau_z = c_q \sum \operatorname{signo}(w_i) w_i^2$$

Ecuación 36. Par de arrastre total sobre el vehículo

$$\vec{\tau}_{xy} = \sum \bar{D}_i \times \vec{T}_i$$

$$\tau_x = c_t d \sum \sin(a_i) w_i^2$$

$$\tau_y = -c_t d \sum \cos(a_i) w_i^2$$

*Ecuación 37. Pares de cabeceo y alabeo totales producidos por los rotores*

El sistema del modelo simplificado generalizado a N rotores se puede expresar matricialmente mediante la Ecuación 38:

$$H = A W^2$$

*Ecuación 38. Sistema de fuerzas del modelo dinámico simplificado*

$$\begin{pmatrix} T \\ \tau_x \\ \tau_y \\ \tau_z \end{pmatrix} = \begin{pmatrix} c_t & c_t & c_t & c_t & \dots & c_t \\ c_t d \sin \alpha_1 & c_t d \sin \alpha_2 & c_t d \sin \alpha_3 & c_t d \sin \alpha_4 & \dots & c_t d \sin \alpha_N \\ -c_t d \cos \alpha_1 & -c_t d \cos \alpha_2 & -c_t d \cos \alpha_3 & -c_t d \cos \alpha_4 & \dots & -c_t d \cos \alpha_N \\ -c_q & c_q & -c_q & c_q & \dots & c_q \text{signo}(w_N) \end{pmatrix} \begin{pmatrix} w_1^2 \\ w_2^2 \\ w_3^2 \\ w_4^2 \\ \dots \\ w_N^2 \end{pmatrix}$$

*Ecuación 39. Sistema de fuerzas del modelo dinámico simplificado desarrollado para un multirroto genérico*

En la Ecuación 38 aparece por primera vez la matriz de control de nuestra simulación, A, esta matriz de control se compone de valores constantes debido a las hipótesis asumidas, y aunque el modelo simplificado no representa fielmente la naturaleza dinámica del dron, su acción conjunta al control PID comentado en la sección anterior, va a ser suficiente para controlar la actitud del dron.

Haciendo la inversa, en el caso que se trate de una matriz cuadrada, o pseudoinversa de A, podemos obtener nuestro vector  $W^2$ , salida de este bloque. Si se trata de una matriz cuadrada el sistema es compatible determinado donde tenemos el mismo número de incógnitas que de ecuaciones la solución es única, este es el caso del cuadricóptero donde tenemos 4 velocidades angulares que resolver y viene representado en la Ecuación 40.

$$\begin{pmatrix} T \\ \tau_x \\ \tau_y \\ \tau_z \end{pmatrix} = \begin{pmatrix} c_t & c_t & c_t & c_t \\ c_t d \sin \alpha_1 & c_t d \sin \alpha_2 & c_t d \sin \alpha_3 & c_t d \sin \alpha_4 \\ -c_t d \cos \alpha_1 & -c_t d \cos \alpha_2 & -c_t d \cos \alpha_3 & -c_t d \cos \alpha_4 \\ -c_q & c_q & -c_q & c_q \end{pmatrix} \begin{pmatrix} w_1^2 \\ w_2^2 \\ w_3^2 \\ w_4^2 \end{pmatrix}$$

*Ecuación 40. Sistema de fuerzas del modelo dinámico simplificado desarrollado para un cuadricóptero*

En cambio, si el número de rotores aumenta, hexacóptero u octocóptero, el sistema que queremos resolver se convierte uno compatible indeterminado dónde existen infinitas soluciones, por eso se recurre a la pseudoinversa de Moore-Penrose, que encuentra la solución por mínimos cuadrados que hace que el valor de las velocidades angulares obtenidas sea el mínimo posible, lo que en términos de eficiencia energética resulta adecuado ya que cuanto menor sea la velocidad de giro de los motores menor será el consumo de la batería.

Los sistemas de ecuaciones del modelo simplificado para el vehículo hexacoptero y octocóptero vienen determinados matricialmente por la Ecuación 41 y la Ecuación 42, respectivamente.

$$\begin{pmatrix} T \\ \tau_x \\ \tau_y \\ \tau_z \end{pmatrix} = \begin{pmatrix} c_t & c_t & c_t & c_t & c_t & c_t & c_t & c_t \\ c_t d \sin \alpha_1 & c_t d \sin \alpha_2 & c_t d \sin \alpha_3 & c_t d \sin \alpha_4 & c_t d \sin \alpha_5 & c_t d \sin \alpha_6 & -c_t d \cos \alpha_4 & -c_t d \cos \alpha_6 \\ -c_t d \cos \alpha_1 & -c_t d \cos \alpha_2 & -c_t d \cos \alpha_3 & -c_t d \cos \alpha_4 & -c_t d \cos \alpha_5 & -c_t d \cos \alpha_6 & c_q & c_q \\ -c_q & c_q & -c_q & c_q & -c_q & c_q & & \end{pmatrix} \begin{pmatrix} W_1^2 \\ W_2^2 \\ W_3^2 \\ W_4^2 \\ W_5^2 \\ W_6^2 \end{pmatrix}$$

*Ecuación 41. Sistema de fuerzas del modelo dinámico simplificado desarrollado para un hexacóptero*

$$\begin{pmatrix} T \\ \tau_x \\ \tau_y \\ \tau_z \end{pmatrix} = \begin{pmatrix} c_t & c_t & c_t & c_t & c_t & c_t & c_t & c_t & c_t & c_t \\ c_t d \sin \alpha_1 & c_t d \sin \alpha_2 & c_t d \sin \alpha_3 & c_t d \sin \alpha_4 & c_t d \sin \alpha_5 & c_t d \sin \alpha_6 & c_t d \sin \alpha_7 & c_t d \sin \alpha_8 & -c_t d \cos \alpha_4 & -c_t d \cos \alpha_8 \\ -c_t d \cos \alpha_1 & -c_t d \cos \alpha_2 & -c_t d \cos \alpha_3 & -c_t d \cos \alpha_4 & -c_t d \cos \alpha_5 & -c_t d \cos \alpha_6 & -c_t d \cos \alpha_7 & -c_t d \cos \alpha_8 & c_q & c_q \\ -c_q & c_q & -c_q & c_q & -c_q & c_q & -c_q & c_q & & \end{pmatrix} \begin{pmatrix} W_1^2 \\ W_2^2 \\ W_3^2 \\ W_4^2 \\ W_5^2 \\ W_6^2 \\ W_7^2 \\ W_8^2 \end{pmatrix}$$

*Ecuación 42. Sistema de fuerzas del modelo dinámico simplificado desarrollado para un octocóptero*

Generalizando, la solución del sistema matricial viene dada por la Ecuación 43.

$$W^2 = A + H$$

*Ecuación 43. Solución al sistema de fuerzas del modelo dinámico simplificado*

El objetivo principal de este bloque es obtener un vector  $W^2$ , que sea capaz de determinar el empuje y los momentos que el piloto solicita desde el mando, esto no significa que el dron vaya a experimentar exactamente ese conjunto de fuerzas y momentos, sino que es una mera aproximación que sirve al dron a acercarse al objetivo.

En conclusión, de este módulo saldrán las velocidades angulares de los rotores, entrarán al modelo dinámico real donde se calculará ahora sí, el movimiento dinámico real del dron, considerando todos las fuerzas y momentos comentados en el apartado 2.3.

### 3.3 Modelo dinámico completo y retroalimentación

Como se ha comentado, una vez hemos obtenido las velocidades angulares al resolver el modelo simplificado, estas entran en el modelo dinámico completo, mucho más realista.

Integrando las ecuaciones del modelo de solido rígido descritas en el apartado 2.4, obtenemos el vector estado, las velocidades, aceleración y demás variables de control que podemos retroalimentar a nuestro bloque PID. En este bloque si está bien calibrado, se va a conseguir reducir a cero el error entre la variable deseada y la obtenida del modelo dinámico realista simplemente modificando la orden, H, que entrará a el modelo simplificado, y con esto cerraríamos el lazo. Este tipo de control se conoce como control de lazo cerrado y es un control típico utilizado en aplicaciones “software in the loop”.

Con la intención de simplificar la idea del control de lazo cerrado, se podría resumir en el siguiente proceso iterativo:

1. El piloto comanda una orden
2. El bloque PID compara la variable actual con la requerida por el piloto, si difieren envía una orden  $\vec{H}$  basada en el error al sistema de control del modelo dinámico simplificado.
3. El modelo simplificado resuelve el sistema compatible y obtiene unas velocidades angulares que envía al modelo dinámico realista.
4. El modelo realista obtiene la nueva posición y orientación del dron actualizando así el valor realimentado al bloque PID.

Llegará un momento en el cual la posición y actitud del dron sean la que ha solicitado el piloto, momento en el cual el bloque PID estabiliza la orden  $\vec{H}$  o señal de control y se consigue una situación estacionaria.

## Chapter 4. Incorporación modelos hexacóptero y octocóptero

El punto de partida de este modelo Matlab-Simulink contaba con la simulación de dos cuadricópteros muy similares con la particularidad de que uno de ellos tenía gimbal y otro de ellos no, de ahora en adelante cuadricóptero WG y cuadricóptero, respectivamente.



*Figura 12. Cuadricoptero real*

La principal motivación de este trabajo era generalizar este modelo dinámico válido para los cuadricópteros,  $N=4$ , y adaptarlo a el hexacóptero,  $N=6$ , y octocóptero,  $N=8$ . No solo se ha obtenido este objetivo si no que se ha parametrizado completamente para un dron simétrico de  $N$  rotores.

Si bien no tiene sentido añadir a el modelo un vehículo del cual no tenemos datos empíricos o teóricos de su planta propulsora y/o de sus características aerodinámicas, se ha decidido dejar el modelo completamente parametrizado en función del número de rotores para futuros proyectos donde incorporar nuevos multirrotores a el modelo.

Esta generalización del modelo se ha conseguido principalmente con:

1. Generalización en función del número de rotores, N, de la matriz de control dentro del constructor de cada vehículo:

```
%MATRIZ DE CONTROL:
%Prealocación:
dron.C=zeros(4,N);

for i=1:N
dron.C(1,i)= dron.b;                               %Thrust
dron.C(2,i)= sin((a0)+((i-1)/N)*2*pi)*dron.b*dron.d; %taux = tau roll
dron.C(3,i)= -1*cos((a0)+((i-1)/N)*2*pi)*dron.b*dron.d; %tauy = tau pitch
dron.C(4,i)=(-1)^(i)*dron.k;                       %tauz = tau yaw
end

%Inversa o pseudoinverta
dron.iC = pinv(dron.C);
```

Utilizando la Ecuación 43 el vector  $W^2$  obtenido tendrá tantos elementos como columnas tenga la matriz A (llamada en el código C). Como se ha comentado este número de elementos viene determinado por N.

2. Generalización en función del número de rotores, N, de las fuerzas y momentos que generan estos. Esto se consigue inicializando con ceros las matrices fuerzas y momentos de 3 filas y N columnas.

Después mediante un bucle desde 1 hasta N, almacenamos en estas matrices las fuerzas y momentos generadas por cada rotor del vehículo.

Cuando el bucle termina, se suman todas las columnas de las matrices y tienes el sistema de fuerzas y momentos total que experimenta el vehículo en movimiento en esa iteración de la simulación.

```
T = zeros(3,dron.nrotors);
Q = zeros(3,dron.nrotors);
tau = zeros(3,dron.nrotors);

for i = 1:dron.nrotors
    %Cálculos de fuerzas y momentos de cada rotor
```

3. Junto a esto es necesario permitir la herencia de las dimensiones de las variables entre bloques de Simulink, ya que algunas de estas dimensiones van a cambiar en función del vehículo simulado. Esto se hace desde el "Model Explorer" con el valor -1 en la casilla "SIZE" de la variable que quieres que herede la dimensión que se le asigna, tal y como indica la Figura 13.



|  | Name    | Scope     | Port | Resolve Signal | DataType                  | Size | InitialValue | CompiledType | CompiledSize |
|--|---------|-----------|------|----------------|---------------------------|------|--------------|--------------|--------------|
|  | Fuerzas | Input     | 1    |                | Inherit: Same as Simulink | 4    |              | unknown      |              |
|  | wmots   | Output    | 1    |                | Inherit: Same as Simulink | -1   |              | unknown      |              |
|  | dron    | Parameter |      |                | Inherit: Same as Simulink | -1   |              | unknown      |              |

Figura 13. Control de dimensiones de variables en Simulink-Model explorer

Por ejemplo, aquí la variable Fuerzas es un vector de dimensiones fijas ya que este vector siempre tendrá los 4 elementos de  $\vec{H} = (T \quad \tau_x \quad \tau_y \quad \tau_z)$ , independientemente del vehículo simulado, en cambio el vector wmots, será de cuatro elementos en el caso del cuadricóptero, de seis elementos en el caso del hexacóptero y de ocho en el caso del octocóptero. Por lo tanto, para que un mismo código Simulink nos sirva para los diferentes vehículos es esencial habilitar esta herencia de dimensiones.

Pero para simular un multirroto de manera fidedigna no solo basta con ampliar su número de rotores en el modelo, es necesario adaptar el constructor de la simulación con las características específicas de cada multirroto para que la simulación sea representativa de las circunstancias reales del vehículo en vuelo.

Por ello se debe hacer un estudio completo de:

1. Masa del vehículo.
2. Dimensiones y características geométricas del dron.
3. Dimensiones y características geométricas de la hélice.
4. Caracterización de la planta propulsora.

A continuación, se analizan estos 4 puntos para los nuevos vehículos implementados en el modelo Matlab-Simulink.

#### 4.1 Hexacóptero

Todas las dimensiones y características geométricas del vehículo que se exponen a continuación han sido calculadas a través del modelo geométrico del hexacóptero gracias al software CAD SolidEdge, resaltar que existen otros métodos mucho más fiables como mediciones más precisas, pero es un buen punto de partida para hacer un primer enfoque más realista.



Figura 14. Hexacóptero real

#### 4.1.1 Masa del vehículo.

La masa del hexacóptero es de 4.178 kg.

#### 4.1.2 Dimensiones y características geométricas del dron.

Las dimensiones geométricas más relevantes para el modelo son la distancia horizontal y vertical de los rotores al centro de masas del dron, estos valores los denominamos como “d” y “h”, respectivamente y el caso del hexacóptero miden:

- $d = 0.41595 \text{ m}$
- $h = 0.05529 \text{ m}$

Otra de las características geométricas muy importante del vehículo sería el rotor de inercia de dron en ejes móviles, para el hexacóptero se ha obtenido a través de SolidEdge.

$$I_1 = \begin{pmatrix} 0,536 & -0,001 & 0 \\ -0,001 & 0,513 & 0 \\ 0 & 0 & 0,967 \end{pmatrix}$$

Ecuación 44. Rotor de inercia del vehículo hexacóptero

#### 4.1.3 Dimensiones y características geométricas de la hélice.

La dimensión principal que utiliza nuestro modelo de la hélice es su radio y este en el caso del hexacóptero mide:  $r = 0.1651 \text{ m}$

Y su rotor de inercia de la hélice en ejes móviles vendría determinado por:

$$I_H = \begin{pmatrix} 0,0001285 & 0 & 0 \\ 0 & 0,0001285 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

Ecuación 45. Rotor de inercia de la hélice del vehículo hexacóptero

#### 4.1.4 Caracterización de la planta propulsora

En este apartado se explica cómo se construye el modelo, a través de datos experimentales, de la planta propulsora del hexacóptero real fotografiado en la Figura 15.



Figura 15. Motor hexacóptero MT4006-13

El motor que propulsa este vehículo ha sido facilitado por el tutor del proyecto y se trata de un MT4006-13 de la marca T-MOTOR de 740 KV. La geometría y dimensiones del motor pueden verse en la Figura 16.

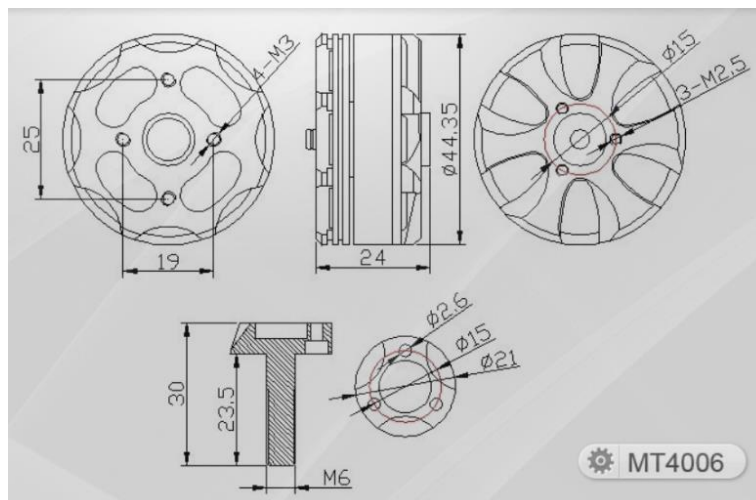


Figura 16. Dimensiones motor hexacóptero MT4006-13

Otras características técnicas importantes del motor MT4006-13 se reflejan en la Figura 17.

**T-Motor MT4006-13 740KV Outrunner Brushless Motor for Multi-copter (3-4S)**

|                                    |             |
|------------------------------------|-------------|
| KV.....                            | 740         |
| Configuration.....                 | 18N24P      |
| Stator Diameter.....               | 40mm        |
| Stator Length.....                 | 6mm         |
| Shaft Diameter.....                | 4mm         |
| Motor Dimensions(Dia.*Len).....    | Φ44.35×24mm |
| Weight (g).....                    | 92g         |
| Idle current(10v)(A).....          | 00.5A       |
| No. of Cells(Lipo).....            | 3-4S        |
| Max Continuous current(A)180S..... | 27A         |
| Max Continuous Power(W)180S.....   | 500W        |
| Max. efficiency current.....       | (5-18A)>84% |
| internal resistance.....           | 136mΩ       |

Figura 17. Especificaciones técnicas motor hexacóptero MT4006-13 [Fuente: [10]]

Este valor KV significa el número de revoluciones por minuto que el motor es capaz de generar en el eje de giro en un régimen sin carga por cada voltio de entrada, es decir, para el caso particular de la batería que alimenta el hexacóptero a plena carga, con un voltaje de 16.8V teóricamente se podrían conseguir aproximadamente 12800 rpm en el eje de este motor.

En el ensayo empírico que se mostrará en este apartado podremos ver que esta velocidad angular no es representativa de la real, esto se debe a que la planta propulsora de cualquier multirrotores lleva asociado un variador de velocidad adecuado para el motor y la batería, que limita este valor teórico máximo de rpm el cual puede generar un motor por motivos de durabilidad del motor y seguridad en el vuelo.

El variador de velocidad del vehículo hexacóptero que se quiere simular es un variador brushless HOBBYWING Platinum PRO 30A 2-6sLiPO.



Figura 18. Controlador de velocidad motor hexacóptero

El variador de velocidad del rotor en un dron es un componente esencial para controlar la velocidad de rotación de los motores. Su función principal es convertir la señal de control proveniente del control del piloto en una señal eléctrica que ajusta la velocidad de los motores, permitiendo al piloto controlar su velocidad de rotación de forma precisa.

Al controlar la velocidad de los motores puede equilibrar las fuerzas de sustentación generada por los rotores, permitiendo que el dron se mantenga estable en el aire, realice maniobras precisas y controle su altura y dirección, todo esto es importante para lograr un vuelo estable,

controlar la altitud, aumentar la maniobrabilidad del vehículo y ajustar la velocidad de respuesta del dron según las necesidades específicas de vuelo.

Además de controlar la velocidad, los variadores de velocidad del rotor se encargan de la regulación de la corriente eléctrica por lo que también ofrecen otras funciones fundamentales como la protección contra sobrecargas, la protección térmica y la retroalimentación de la posición de los motores. Estas características adicionales ayudan a garantizar un funcionamiento seguro y confiable del dron.

Por último, nos queda describir la hélice que lleva acoplada el motor, esta se trata de una P13\*4.4, donde el primer dígito 13, significa la longitud total de la hélice en pulgadas, es decir esta hélice tiene un diámetro de 330.2 mm, y el segundo dígito 4.4, significa la curvatura que tiene la hélice lo que se denomina paso de la hélice, este valor también se mide en pulgadas y se define como la distancia vertical teórica que avanzaría la hélice tras una revolución completa si esta fuera un tornillo en un medio solo, este es un factor muy importante para el rendimiento y eficiencia de una planta propulsora.



Figura 19. Hélice del conjunto motor hexacóptero

El conjunto motor-hélice modelizado en este proyecto se puede ver en la Figura 20.



Figura 20. Conjunto motor hexacóptero

Una vez que ya tenemos descritos todos los componentes de la planta propulsora del hexacóptero vamos a describir el proceso seguido para su incorporación al modelo.



En primer lugar y antes de empezar con el ensayo empírico se buscaron ensayos similares para la misma planta propulsora en internet. Tras una pequeña investigación, encontramos la Figura 21 en [10] con características muy similares a nuestro ensayo ya que era el mismo sistema motor-hélice MT4006 – 13\*4.4, el mismo voltaje de la batería 14.8V, pero con la salvedad de que no teníamos información del variador de velocidad que se había utilizado en el ensayo.

| Item No.            | Volts (V)         | Prop                | Throttle | Amps (A) | Watts (W) | Thrust (g) | RPM   | Efficiency (g/W) | Operating temperature( °C) |    |    |
|---------------------|-------------------|---------------------|----------|----------|-----------|------------|-------|------------------|----------------------------|----|----|
| MT4006<br>KV740     | 11.1              | T-MOTOR<br>12*4CF   | 50%      | 2.6      | 29        | 360        | 4000  | 12.47            | 43                         |    |    |
|                     |                   |                     | 65%      | 4.1      | 46        | 480        | 4700  | 10.55            |                            |    |    |
|                     |                   |                     | 75%      | 5.5      | 61        | 600        | 5100  | 9.83             |                            |    |    |
|                     |                   |                     | 85%      | 7.4      | 82        | 750        | 5750  | 9.13             |                            |    |    |
|                     |                   | T-MOTOR<br>13*4.4CF | 50%      | 2.9      | 32        | 360        | 3700  | 11.18            |                            | 46 |    |
|                     |                   |                     | 65%      | 4.7      | 52        | 510        | 4500  | 9.78             |                            |    |    |
|                     |                   |                     | 75%      | 6.4      | 71        | 620        | 5000  | 8.73             |                            |    |    |
|                     |                   |                     | 85%      | 8.4      | 93        | 800        | 5600  | 8.58             |                            |    |    |
|                     |                   | T-MOTOR<br>14*4.8CF | 50%      | 3.6      | 40        | 460        | 3400  | 11.51            |                            |    | 50 |
|                     |                   |                     | 65%      | 6.3      | 70        | 690        | 4100  | 9.87             |                            |    |    |
|                     |                   |                     | 75%      | 8.6      | 95        | 860        | 4600  | 9.01             |                            |    |    |
|                     |                   |                     | 85%      | 11.4     | 127       | 1040       | 5100  | 8.22             |                            |    |    |
|                     | T-MOTOR<br>15*5CF | 50%                 | 4.3      | 48       | 580       | 3100       | 12.15 | 54               |                            |    |    |
|                     |                   | 65%                 | 7.8      | 87       | 860       | 3900       | 9.93  |                  |                            |    |    |
|                     |                   | 75%                 | 10.6     | 118      | 1020      | 4300       | 8.67  |                  |                            |    |    |
|                     |                   | 85%                 | 14.1     | 157      | 1240      | 4700       | 7.92  |                  |                            |    |    |
|                     | 14.8              | T-MOTOR<br>11*3.7CF | 50%      | 3.5      | 52        | 380        | 5400  |                  | 7.34                       | 46 |    |
|                     |                   |                     | 65%      | 4.9      | 73        | 520        | 6200  |                  | 7.17                       |    |    |
|                     |                   |                     | 75%      | 6.7      | 99        | 670        | 6800  |                  | 6.76                       |    |    |
|                     |                   |                     | 85%      | 8.9      | 132       | 820        | 7600  |                  | 6.23                       |    |    |
|                     |                   | T-MOTOR<br>12*4CF   | 50%      | 3.9      | 58        | 540        | 5000  |                  | 9.36                       |    | 48 |
|                     |                   |                     | 65%      | 6.1      | 90        | 790        | 5800  |                  | 8.75                       |    |    |
|                     |                   |                     | 75%      | 8.5      | 126       | 980        | 6600  |                  | 7.79                       |    |    |
|                     |                   |                     | 85%      | 11.1     | 164       | 1190       | 7200  |                  | 7.24                       |    |    |
| T-MOTOR<br>13*4.4CF |                   | 50%                 | 4.3      | 64       | 590       | 4750       | 9.27  | 53               |                            |    |    |
|                     |                   | 65%                 | 6.9      | 102      | 800       | 5700       | 7.83  |                  |                            |    |    |
|                     |                   | 75%                 | 9.4      | 139      | 1080      | 6300       | 7.76  |                  |                            |    |    |
|                     |                   | 85%                 | 12.6     | 186      | 1260      | 7000       | 6.76  |                  |                            |    |    |
| T-MOTOR<br>14*4.8CF | 50%               | 5.6                 | 83       | 780      | 4400      | 9.41       | 63    |                  |                            |    |    |
|                     | 65%               | 9.4                 | 139      | 1090     | 5200      | 7.83       |       |                  |                            |    |    |
|                     | 75%               | 12.6                | 186      | 1340     | 5700      | 7.19       |       |                  |                            |    |    |
|                     | 85%               | 16.9                | 250      | 1560     | 6300      | 6.24       |       |                  |                            |    |    |
| 100%                | 20                | 296                 | 1800     | 6600     | 6.08      |            |       |                  |                            |    |    |

Notes: The test condition of temperature is motor surface temperature in 100% throttle while the motor run 10 min.

Figura 21. Resultados ensayo experimental similar al desarrollado en este documento

Aunque no se trate de el mismo ensayo exactamente, estos datos experimentales no dan una idea del comportamiento de la planta propulsora a modelizar y no pueden servir como valores de referencia para nuestro ensayo.

En la tabla del ensayo descrito en [10] podemos observar que se recogen las siguientes variables de conjunto motor: Empuje en gramos [g], consumo en amperios [A], potencia suministrada en vatios [W], régimen de giro en revoluciones por minuto [rpm], eficiencia energética en gramos/vatios [g/W], todo esto para 5 posiciones del “throttle”.

Se realiza un ensayo similar en este proyecto, pero con un enfoque algo diferente siguiendo el método descrito en [11]. El método de ensayo que se describe a continuación se utiliza también para modelizar la descarga de la batería, los resultados relativos a este se recogen en el apartado 5.1.

Antes de describir el método seguido, se va a presentar la instrumentación de medida, se trata de una bancada de ensayo suministrada por el tutor del trabajo compuesto por los siguientes elementos marcados en la Figura 22según el siguiente orden:

1. Contador de revoluciones del rotor: Independiente al circuito y que sirve para determinar el régimen de giro del motor.
2. Vatímetro: Conectado en la batería y al variador de velocidad del motor. En este dispositivo podemos ver los parámetros eléctricos más importantes para saber el consumo energético del motor, así como el estado de la batería. Los parámetros que recoge el vatímetro en tiempo real son:
  - a) Voltaje remanente de la batería en voltios.
  - b) Intensidad suministrada por la batería en amperios.
  - c) Potencia suministrada por la batería en vatios.
  - d) Carga eléctrica por unidad de tiempo en miliamperios hora.
3. Báscula: Mide la cantidad de gramos de empuje que proporciona la hélice, lo hace a través del martillo conectado a la misma estructura que la hélice. Al ser esta estructura diseñada en forma de L, cuando la hélice ejerza un empuje en un extremo de la L en la dirección de su eje de giro, el martillo, en el otro extremo, ejercerá el mismo empuje sobre la báscula.
4. Receptor de radio: Dispositivo conectado al variador de velocidad con fuente de alimentación propia y que sirve para comunicar el mando del piloto con el variador de velocidad del motor, con el único fin de modificar el régimen de giro del motor a través de “throttle” del mando del piloto.

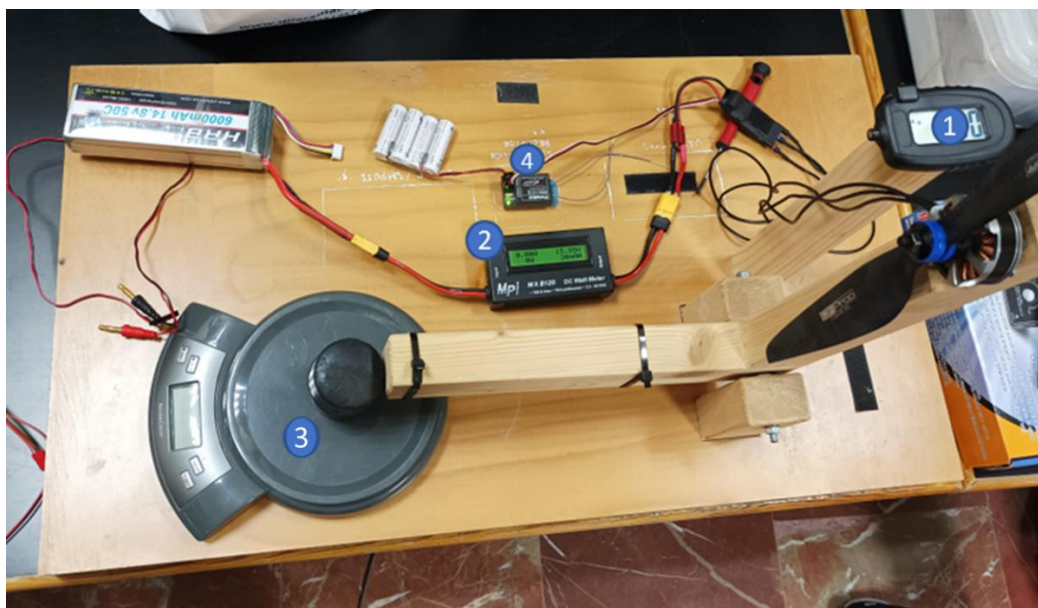


Figura 22. Bancada de ensayos para conjunto motor

En este apartado se quiere recoger el empuje que proporciona el rotor del hexacóptero, por lo que solo se aportarán los datos recogidos del ensayo relacionados con este empuje, es decir los datos recogido con la báscula.

Como se ha comentado el método seguido para recoger los datos de esta planta propulsora, así como de la batería del sistema es el descrito en [11]

En primer lugar, se realiza el montaje del circuito eléctrico, se carga la batería al 100% y antes de alimentar el motor se comprueba que el voltaje de la batería y sus celdas es el correcto a través del “digital battery checker” que aparece en la Figura 23.



Figura 23. Dispositivo para comprobar el estado de carga de la batería

La recogida de valores medidos a distintos regímenes de giro del rotor, se hace de la siguiente manera:

Se comenzarán a recoger datos del ensayo a la velocidad mínima a la que pueda girar el motor, en el caso del hexacóptero esta velocidad mínima es de aproximadamente 1000 revoluciones por minuto, se recogerán medidas cada 10 segundos durante un tiempo aproximado de 1 minuto y medio, por lo tanto, se recogerán unas 9-10 medidas. Una vez se tengan todas estas mediciones incrementaremos la velocidad de giro del rotor en 500 rpm y se repetirá el proceso hasta llegar a aproximadamente el 90% del “throttle” del mando del piloto. Este ensayo se repite 3 veces, siempre iniciando el ensayo con la batería al 100% de carga.

Así por cada régimen de giro en intervalos de 500 rpm tendremos el empuje en gramos que proporciona este sistema propulsor (conjunto motor-variador de velocidad-hélice).

Una vez se han recogido del ensayo una muestra de medidas lo suficientemente grande, casi 30 mediciones por cada régimen de giro estudiado, y haciendo un promedio de todas las medidas obtenidas en cada régimen de giro obtenemos la Tabla 1, donde podemos ver el promedio del empuje suministrado por el motor en función del promedio de la velocidad angular que ha conseguido en cada posición del “throttle” del mando.

El régimen de saturación de este motor estará en torno a 7000 rpm. No obstante, no se ha querido llegar a estas velocidades en el ensayo para no poner en peligro el motor, por lo tanto,



el valor del empuje a esta velocidad angular se estima mediante extrapolación de los datos experimentales obtenidos a menores velocidades de giro, representados en la Figura 25, los cuales mediante un ajuste polinómico de grado 2 por mínimos cuadrados sigue la expresión:

$$T = 2.40308856E - 12 * (w^2)^2 + 0.0000178887744 * w^2 - 0.068919828$$

*Ecuación 46. Ajuste polinómico de segundo grado del empuje obtenido experimentalmente del motor MT4006-13*

Obteniéndose así que el empuje cuando el motor gira a 6990 rpm es de 10.21 N, los valores obtenidos por extrapolación se mostrarán con otro color en las figuras de este apartado para denotar que no han sido obtenidos experimentalmente como el resto.

Haciendo uso de la Ecuación 17, podemos obtener la tercera columna de la Tabla 1, es decir, el valor del coeficiente de empuje en [Kg·m] para cada régimen de giro estudiado.

| <b>w [rad/s]</b> | <b>T [N]</b> | <b>Ct [kg ·m]</b> |
|------------------|--------------|-------------------|
| 111              | 0.17         | 1.381E-05         |
| 160              | 0.38         | 1.475E-05         |
| 211              | 0.70         | 1.562E-05         |
| 260              | 1.14         | 1.685E-05         |
| 316              | 1.68         | 1.680E-05         |
| 368              | 2.39         | 1.761E-05         |
| 418              | 3.17         | 1.815E-05         |
| 472              | 4.02         | 1.808E-05         |
| 527              | 5.05         | 1.821E-05         |
| 577              | 6.13         | 1.842E-05         |
| 634              | 7.58         | 1.887E-05         |
| 680              | 8.78         | 1.896E-05         |
| 732              | 10.21        | 1.893E-05         |

*Tabla 1. Resultados experimentales del ensayo del motor MT4006-13*

Como muestra la Figura 24 el valor del coeficiente dimensional de empuje,  $c_t$ , va subiendo a medida que aumenta el régimen de giro hasta que parece estabilizarse en torno del 1.85E-5 a altos regímenes.

Esto indica que, para considerar la hipótesis impuesta en el apartado 3.2, donde  $c_t$  se considera constante, hay que hacer cierto ajuste de los datos empíricos obtenidos.

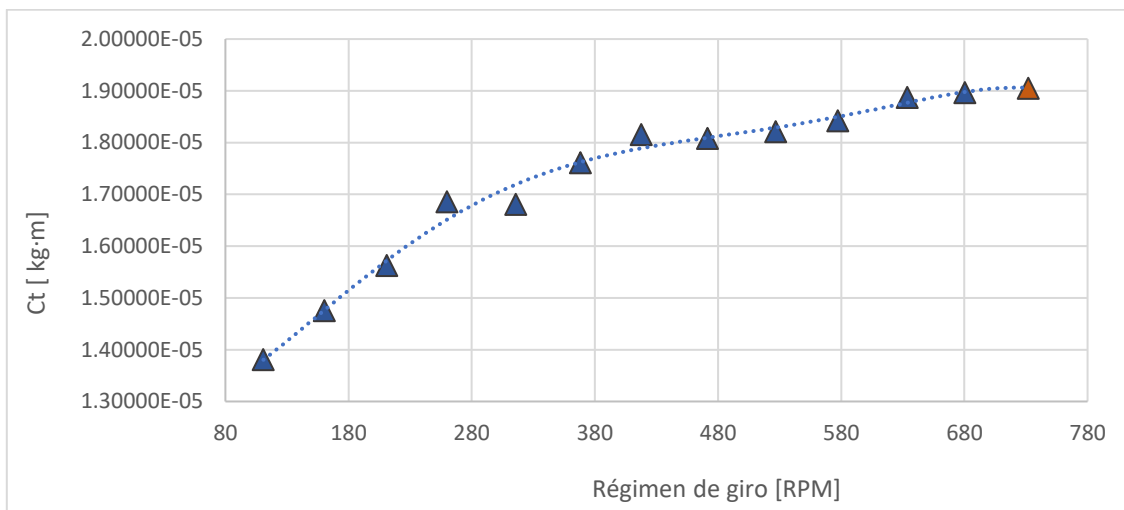


Figura 24. Coeficiente de empuje dimensional del hexacóptero en función del régimen de giro

La Ecuación 17 tiene carácter lineal de pendiente  $c_t$ , donde, según la tercera hipótesis del modelo dinámico simplificado, este valor de pendiente es constante en todo el rango de operación. A continuación, se realiza un ajuste de los datos experimentales obtenidos en el ensayo con el fin de encontrar ese valor del coeficiente tratando de cubrir de manera realista el máximo número de regímenes de giro del motor. Partiendo de los datos medidos en el ensayo que aparecen en la Tabla 2. En la figura podemos ver el carácter lineal de la evolución del empuje con el cuadrado de la velocidad angular a la que gira el rotor.

| $W^2$ [rad/s] | T [N] |
|---------------|-------|
| 12282         | 0.17  |
| 25671         | 0.38  |
| 44525         | 0.70  |
| 67494         | 1.14  |
| 99729         | 1.68  |
| 135708        | 2.39  |
| 174355        | 3.17  |
| 222324        | 4.02  |
| 277544        | 5.05  |
| 333041        | 6.13  |
| 401557        | 7.58  |
| 463075        | 8.78  |
| 535811        | 10.21 |

Tabla 2. Empuje en función del cuadrado de la velocidad angular de giro del motor MT4006-13

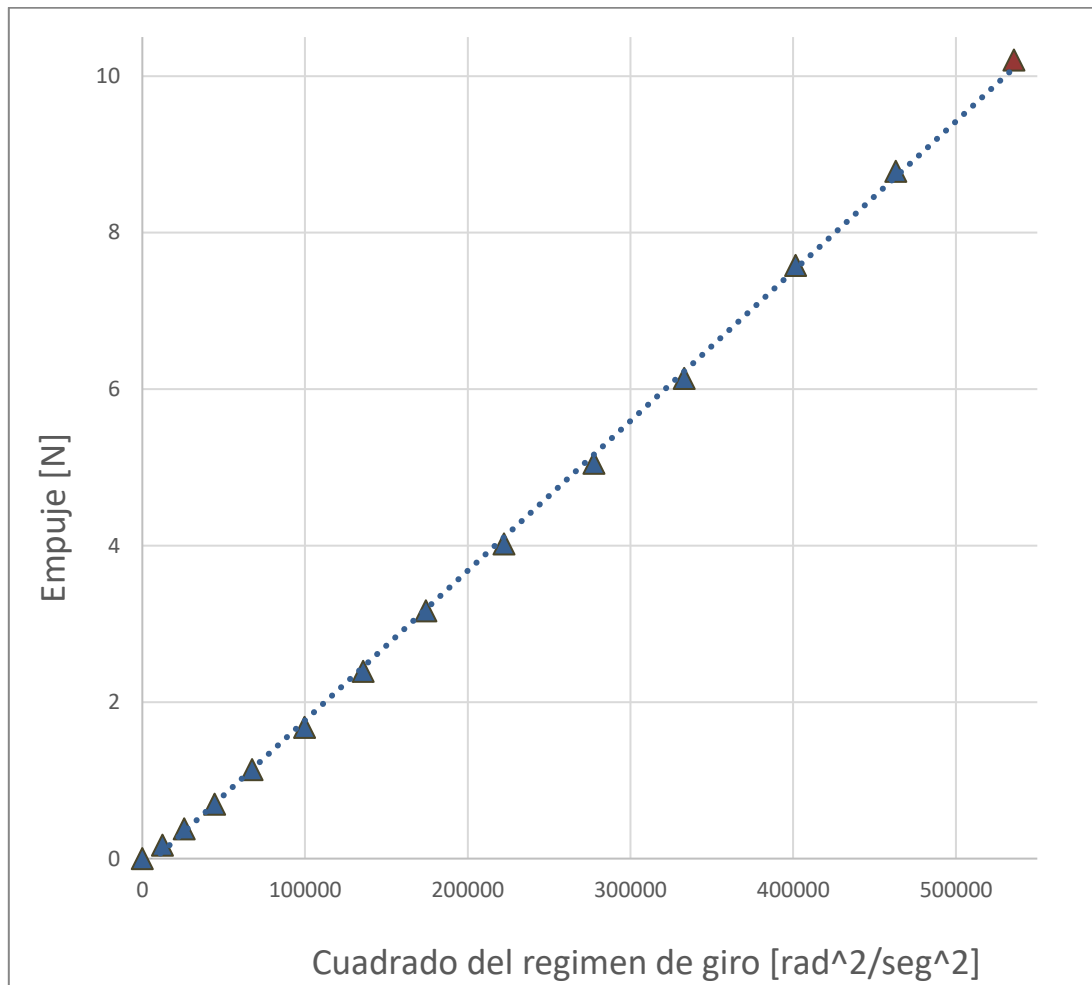


Figura 25. Ley de empuje en función del cuadrado de la velocidad angular de giro del motor MT4006-13

El primer intento de aproximación de la curva de empuje a los datos empíricos que se propone para hallar el coeficiente de empuje teórico, es a través de la media ponderada de todos los valores del coeficiente obtenido en el ensayo para cada velocidad de operación, el resultado de este valor medio de  $c_t$  es 0.00001713, en la Figura 26 en color verde podemos ver el resultado de esta aproximación. Esta aproximación es realmente representativa de la realidad a velocidades de giro bajas, pero a medida que aumenta el régimen de giro la aproximación empieza a subestimar el empuje real del motor por lo que este valor de  $c_t$  será descartado.

El siguiente enfoque es usando la pendiente del ajuste lineal clásico por mínimos cuadrados, la línea de ajuste se aproxima mucho a los datos obtenidos empíricamente y gracias a ella podemos hacer una aproximación del coeficiente de empuje de  $c_t = 0.00001914 \text{ kg} \cdot \text{m}$ .

En la Figura 26 podemos ver en naranja el resultado de aproximar los datos experimentales con  $T = 0.00001914 \text{ w}^2$ . Esta aproximación sobreestimaré un poco los resultados obtenidos experimentalmente, pero sigue perfectamente la tendencia de los datos experimentales sin penalizar en exceso ningún régimen de giro, por lo tanto, este valor de  $c_t$  será un serio candidato para modelar la planta propulsora del hexacóptero.

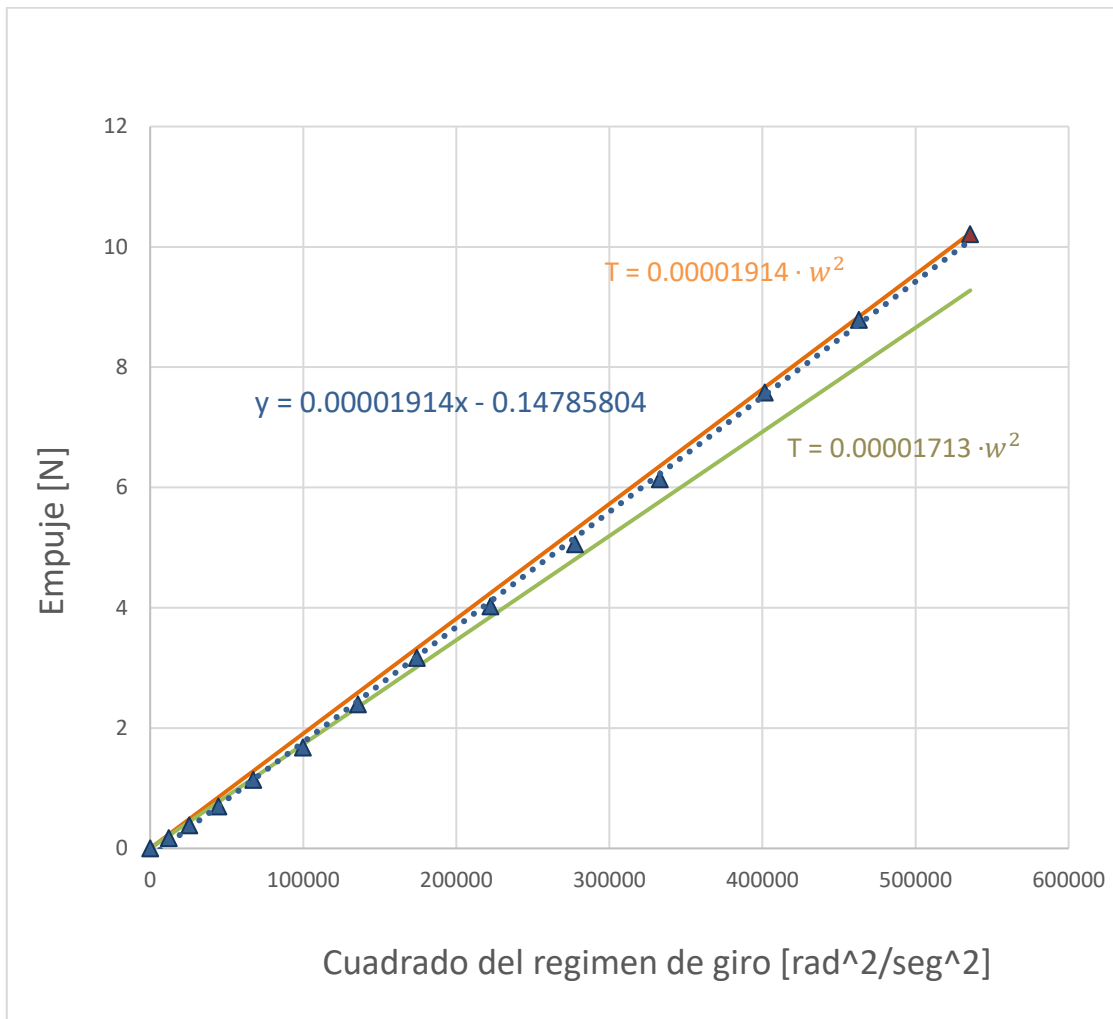


Figura 26. Ajuste lineal por mínimos cuadrado y primeras aproximaciones lineales a la ley de empuje

Otro enfoque que se propone es el ajuste polinómico por mínimos cuadrados utilizando un polinomio de segundo grado, el resultado viene reflejado en la Figura 27 y podemos que se ajusta mucho mejor que el ajuste lineal. El resultado de la aproximación lineal a través de este ajuste resulta en un  $c_t = 0.00001775$ .

En la Figura 27 podemos ver el en color granate la aproximación lineal del empuje según  $T = 0.00001775 w^2$ , se observa que para regímenes de giro bajos la concordancia con los resultados experimentales es absoluta, pero a medida que aumenta la velocidad angular del rotor, la aproximación empieza a subestimar el empuje del motor.

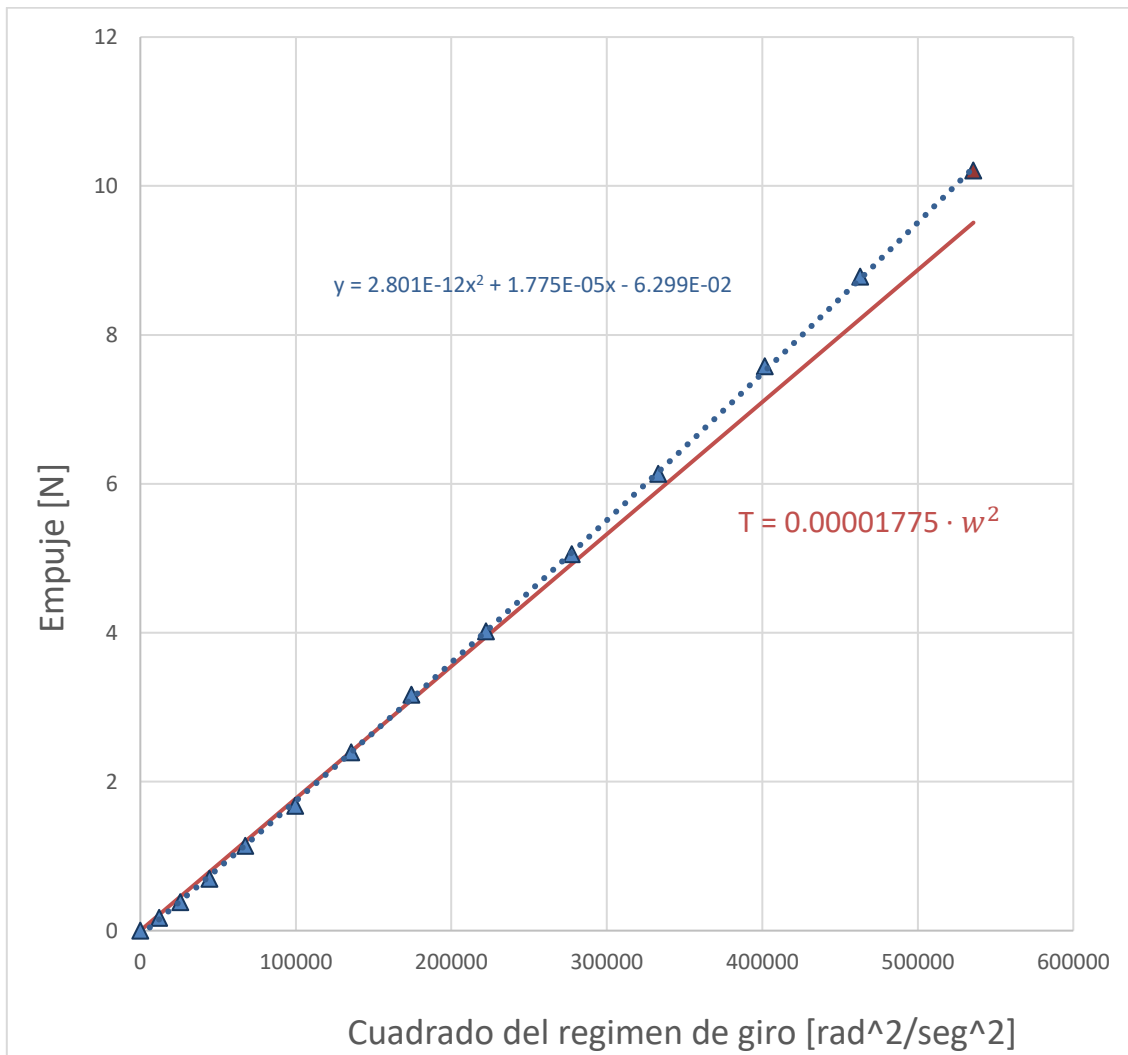


Figura 27. Ajuste polinómico de segundo grado por mínimos cuadrados y aproximación lineal de la ley de empuje

Por último, vamos a considerar el ajuste polinómico por mínimos cuadrados utilizando un polinomio de tercer grado representado en la Figura 28. Esta curva de ajuste se corresponde aún más que la anterior a los datos experimentales, pero al imponer la aproximación lineal del empuje asumiendo que  $c_t = 0.00001688$ , no podemos olvidar que los valores de abscisas de estas curvas son muy elevados y al suponer que esta curva de ajuste es equivalente a  $T = 0.000016666 w^2$  estamos descartando términos cúbicos y cuadrados de la línea de ajuste que harán que la aproximación no sea tan buena.

En la Figura 28 se puede como la aproximación, en color morado, realmente no es representativa de los datos experimentales para la mayor parte del régimen de giro del rotor.

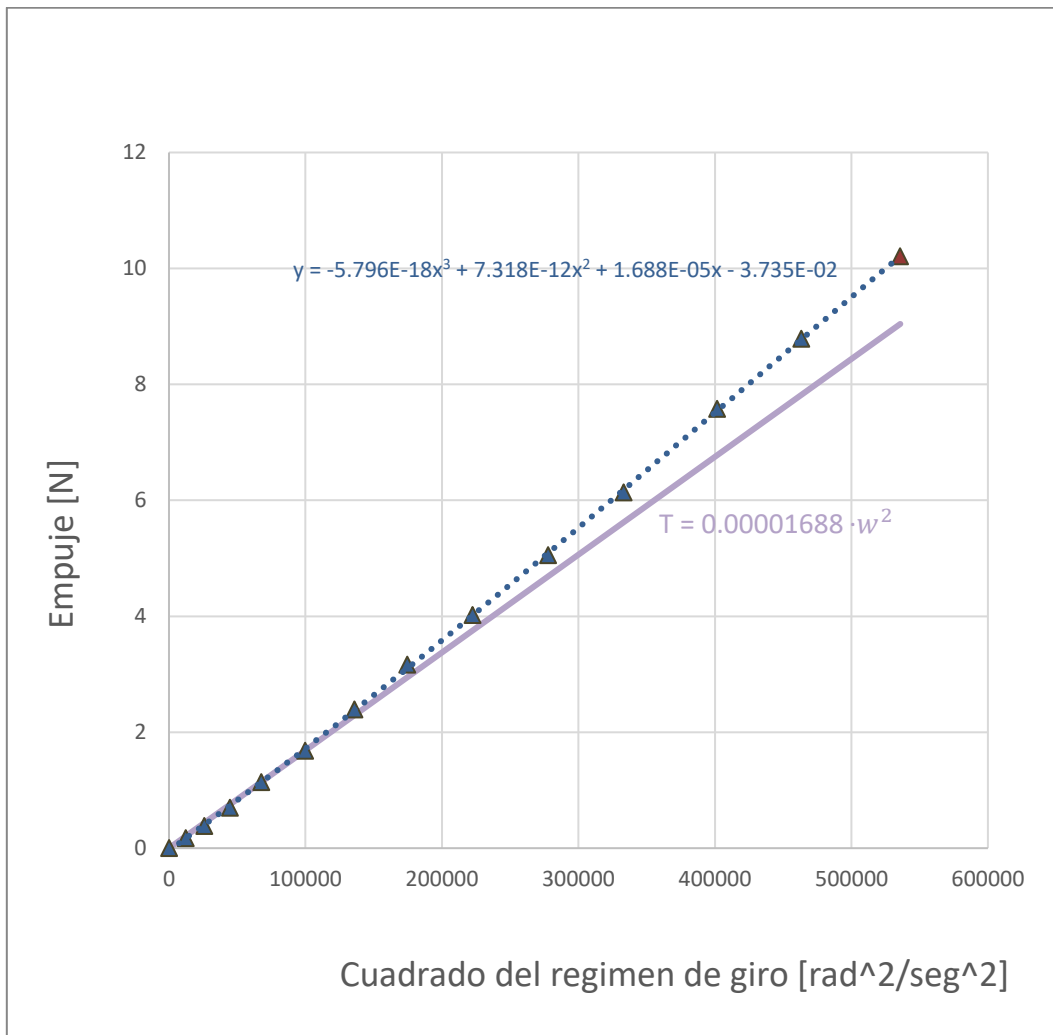


Figura 28. Ajuste polinómico de tercer grado por mínimos cuadrados y aproximación lineal de la ley de empuje

De los resultados obtenidos de los ensayos del motor MT4006-13 podemos concluir que el coeficiente  $c_t$  en la práctica no es constante y hay muchas maneras de determinarlo, por ello hay que elegir un tipo de ajuste y de él, una aproximación lineal para obtener el modelo más representativo posible.

Tras analizar cuatro tipos de ajuste, se ha considerado que para nuestro modelo el más realista para el hexacóptero es la aproximación obtenida tras el ajuste lineal por mínimos cuadrados, que obtiene un coeficiente de empuje a  $c_t = 0.00001914$  subestimando un poco el empuje, sobre todo a bajas rpm, pero sin penalizar en exceso ningún rango de giro del motor.

Por lo tanto, para la planta propulsora del hexacóptero el empuje en función de la velocidad angular vendrá determinado por la expresión:

$$T = 0.00001914 w^2 .$$

## 4.2 Octocóptero

El multirroto de ocho rotores u octocóptero tiene mayores dimensiones que el hexacóptero, por lo que, por lo general pesará más, experimentará momentos de cabeceo y alabeo más elevados y dispondrá de una planta propulsora con mayor capacidad de empuje. En la Figura 29 podemos ver el vehículo octocóptero real que se va a incorporar al modelo Matlab-Simulink.



Figura 29. Octocóptero real

### 4.2.1 Masa del vehículo.

La masa del octocóptero es de 5.75 kg, algo más pesado que el hexacóptero.

### 4.2.2 Dimensiones y características geométricas del dron.

Las dimensiones geométricas relevantes para el modelo en el caso del octocóptero son:

- $d = 0.581 \text{ m}$
- $h = 0.0571 \text{ m}$

El rotor de inercia de octocóptero en ejes móviles viene dado por:

$$I_1 = \begin{pmatrix} 0,136 & 0 & 0 \\ 0 & 0,154 & 0 \\ 0 & 0 & 0,217 \end{pmatrix}$$

Ecuación 48. Rotor de inercia del vehículo octocóptero

### 4.2.3 Dimensiones y características geométricas de la hélice.

Como se ha comentado la dimensión principal de la hélice es su radio y este en el caso del octocóptero mide:  $r = 0.1905 \text{ m}$

Y su rotor de inercia de la hélice en ejes móviles vendría determinado por:

$$I_H = \begin{pmatrix} 0,000436 & 0 & 0 \\ 0 & 0,000436 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

Ecuación 49. Rotor de inercia de la hélice del vehículo octocóptero

#### 4.2.4 Caracterización de la planta propulsora

En este apartado se modela la planta propulsora del octocóptero. El motor de este vehículo ha sido suministrado por el tutor del trabajo y se ha realizado el mismo ensayo de bancada explicado en apartado 4.1.4.

El motor de este vehículo es el MN4014-9 de la marca T-MOTOR de 400 KV. Comparado con los 740 KV del MT4006-13 que propulsa el hexacóptero, las hélices del octocóptero girarán a menos velocidad ante un mismo voltaje pero que un motor gire más despacio no quiere decir necesariamente que propulse menos, todo esto también viene influenciado por la hélice.



Figura 30. Motor octocóptero MN4014-9

A continuación, en la Figura 31 se pueden las especificaciones técnicas del MN4014-9, donde se recogen tanto los parámetros geométricos, eléctricos y de potencia del motor.

Specifications:

|                                    |              |
|------------------------------------|--------------|
| KV.....                            | 400          |
| Configuration.....                 | 18N24P       |
| Stator Diameter.....               | 40mm         |
| Stator Length.....                 | 14mm         |
| Shaft Diameter.....                | 4mm          |
| Motor Dimensions(Dia.*Len).....    | Φ44.7×34.5mm |
| Weight (g).....                    | 150g         |
| Idle current(10)@10v(A).....       | 1.3A         |
| No.of Cells(Lipo).....             | 4-8S         |
| Max Continuous current(A)180S..... | 30A          |
| Max Continuous Power(W)180S.....   | 900W         |
| Max. efficiency current.....       | (3-12A)>81%  |
| internal resistance.....           | 67mΩ         |

Figura 31. Especificaciones técnicas motor octocóptero MN4014-9 [Fuente: [12]]

El variador de velocidad que irá conectado al motor del vehículo octocóptero se trata de un T-Motor AIR Multirotor ESC 40 A. La velocidad de giro mínima de esta planta propulsora está en torno a 500 rpm mientras que el régimen de saturación no excede las 5000 rpm, muy alejado de las casi 7000 a las que era capaz de girar el motor del hexacóptero.





Figura 32. Controlador de velocidad motor octocóptero

Respecto a la hélice podemos destacar su mayor tamaño respecto a la del hexacóptero, lo que al final se va a traducir en una mayor relación empuje/régimen de giro, es decir en un mayor coeficiente,  $c_t$ .

De la hélice de este rotor no hemos podido averiguar ni su fabricante ni el valor de su paso, pero si hemos obtenido se diámetro de 381 mm a través del modelo geométrico en Solid Edge.

En la Figura 33 podemos ver el conjunto hélice-motor que se ha utilizado para el ensayo cuyos resultados se exponen a continuación.

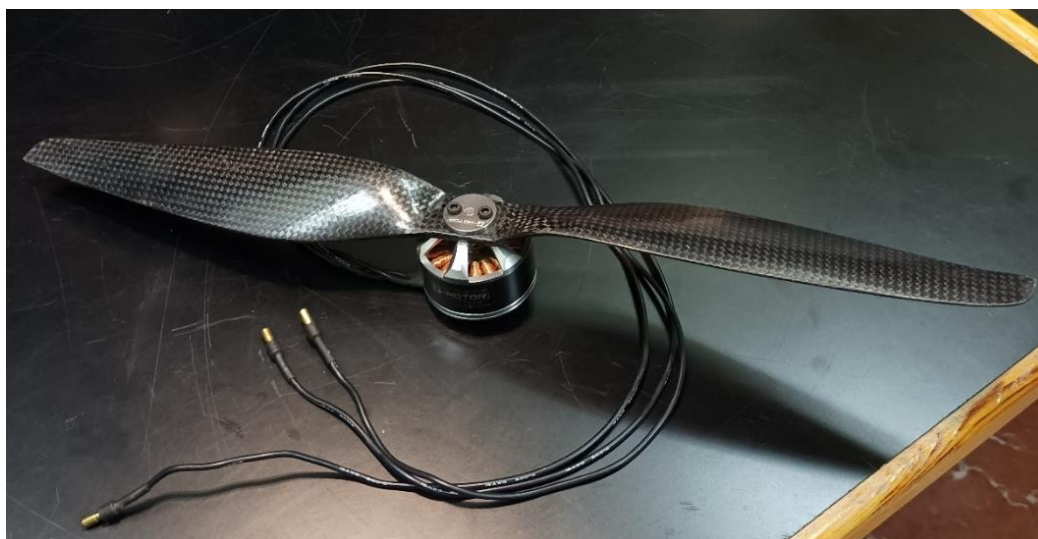


Figura 33. Conjunto motor octocóptero

Siguiendo el mismo procedimiento para el ensayo descrito en el apartado 4.1.4, y tras recoger más de 30 muestras por cada régimen de giro al término de los tres ensayos equivalentes, podemos construir la Tabla 3 y determinar el valor del coeficiente  $c_t$  para cada régimen de giro estudiado.

Como se ha comentado el valor de saturación de revoluciones por minuto de este motor estará en torno a 4900 rpm. No obstante, no se ha querido llegar a estas velocidades en el ensayo para no dañar el dispositivo por lo que, el valor del empuje a este régimen de giro se ha estimado mediante extrapolación de los datos experimentales obtenidos a menores velocidades de giro, representados en la Figura 35, los cuales mediante un ajuste polinómico de grado 3 por mínimos cuadrados sigue la Ecuación 50:

$$T = -5.5668E-18(w^2)^3 + 3.3421E-11(w^2)^2 + 4.1268E-05w^2 - 9.6963E-02$$

*Ecuación 50. Ajuste polinómico de tercer grado del empuje obtenido experimentalmente del motor MN4014-9*

Obteniéndose así un empuje de 13.16 N para una velocidad angular del rotor próxima a 4900 revoluciones por minuto. Se marca en distinto color en la Figura 35 para denotar que no ha sido obtenido experimentalmente como el resto.

En la Figura 34 observamos como el coeficiente dimensional de empuje no es del todo constante en el rango de operación y aunque si es cierto a partir de las 1500 revoluciones por minuto si tiene a estabilizarse en tornos a 4E-5 existe una gran diferencia con el obtenido a menores velocidades de giro.

| w [rad/s] | T [N] | Ct [kg · m] |
|-----------|-------|-------------|
| 57        | 0.04  | 1.31918E-05 |
| 125       | 0.41  | 2.67001E-05 |
| 156       | 0.93  | 3.79728E-05 |
| 206       | 1.69  | 3.98701E-05 |
| 262       | 2.95  | 4.29703E-05 |
| 317       | 4.44  | 4.42108E-05 |
| 365       | 5.93  | 4.45472E-05 |
| 416       | 8.00  | 4.61813E-05 |
| 467       | 10.44 | 4.79004E-05 |
| 516       | 13.16 | 4.93959E-05 |

*Tabla 3. Resultados experimentales del ensayo del motor MN4014-9*

Este valor tan cambiante del coeficiente  $c_t$  es un indicativo de que la hipótesis de que el empuje lleva una relación lineal con el cuadrado de la velocidad angular del rotor, no va a ser tan buena como en el caso del hexacóptero y los resultados de la aproximación hecha por la hipótesis diferirán más de los datos experimentales.

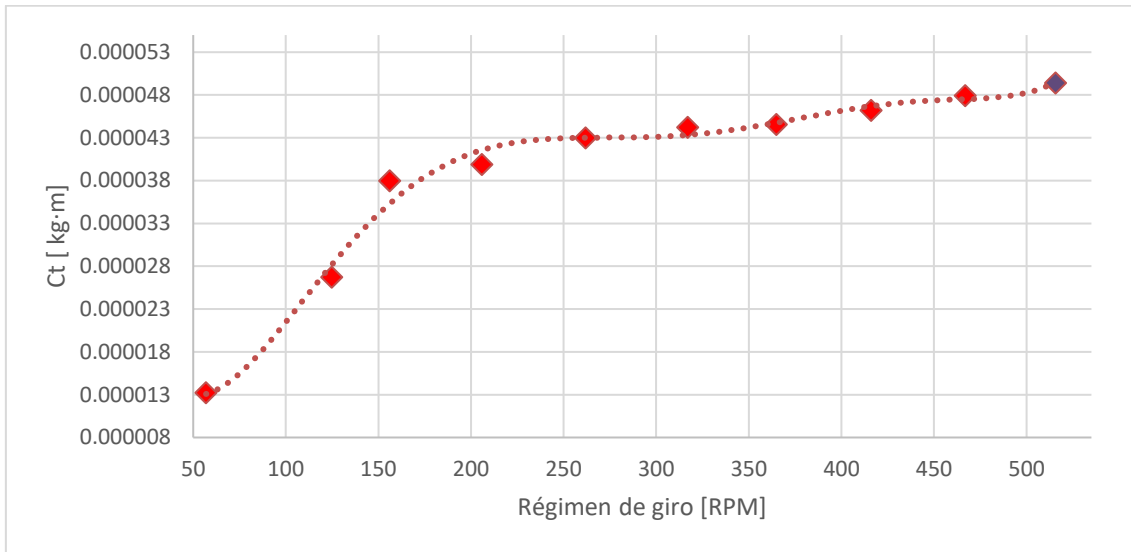


Figura 34. Coeficiente de empuje dimensional del octocóptero en función del régimen de giro

| $W^2$ [rad/s] | T [N] |
|---------------|-------|
| 3240          | 0.04  |
| 10228         | 0.10  |
| 24388         | 0.93  |
| 42364         | 1.69  |
| 68627         | 2.95  |
| 100487        | 4.44  |
| 133187        | 5.93  |
| 173099        | 8.00  |
| 217940        | 10.44 |
| 265885        | 13.13 |

Tabla 4. Empuje en función del cuadrado de la velocidad angular de giro del motor MN4014-9

Con los datos de la tabla podemos obtener en la Figura 35 la evolución cuasi lineal del empuje con el cuadrado de la velocidad angular, y así empezar a estimar el coeficiente dimensional de empuje,  $c_t$ . En la misma Figura 35 aparece superpuesta la línea de tendencia utilizada para la extrapolación del empuje a altas velocidades de giro.

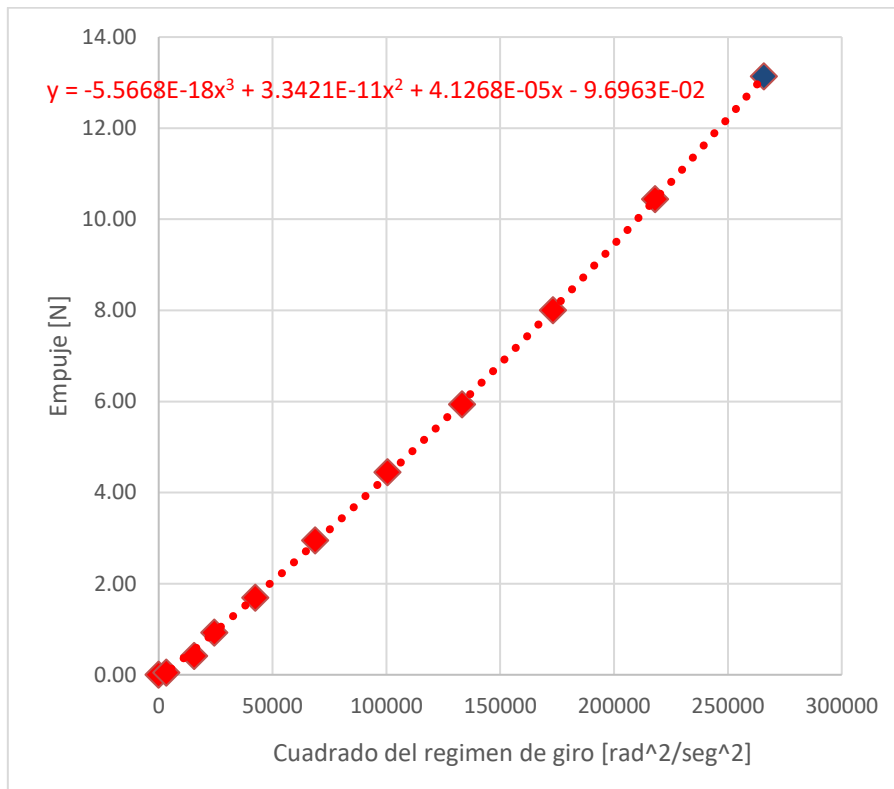


Figura 35. Ley de empuje en función del cuadrado de la velocidad angular de giro del motor MN4014-9

Ampliando la zona de la Figura 35 a bajas velocidades de giro, en torno a 500 -1200 rpm, obtenemos la Figura 36, dónde vemos como la tendencia lineal de la gráfica desaparece completamente, existiendo una clara tendencia exponencial en los valores de empuje obtenidos a velocidades de rotación inferiores a 1500 rpm.

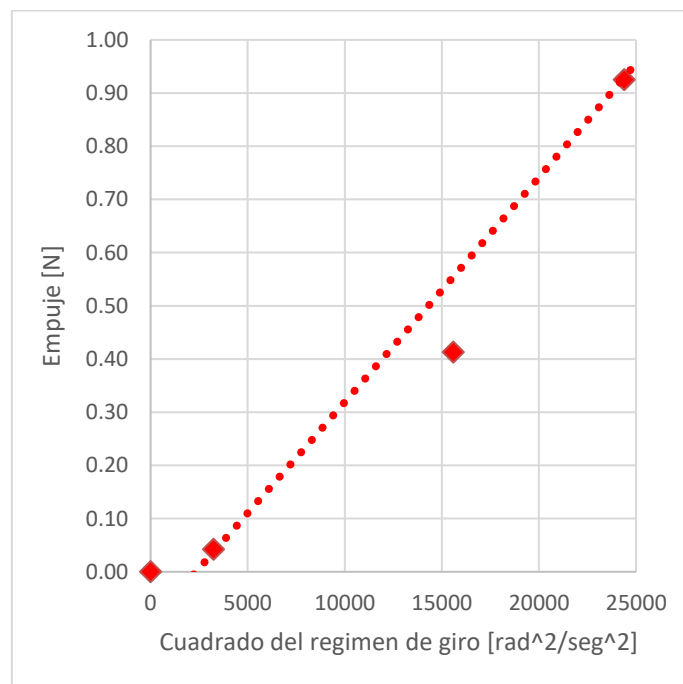


Figura 36. Zoom a bajas revoluciones de la ley de empuje del motor MN4014-9

Este comportamiento totalmente inesperado se puede deber principalmente a dos motivos:

1. Hipótesis no válida para este régimen del motor
2. Fallo en los aparatos de medida: Contador de revoluciones y báscula

Tras analizar toda la muestra y viendo el carácter lineal claro de los resultados experimentales en función del cuadrado de la velocidad angular se piensa que lo más probable es que sea un problema de resolución del contador de revoluciones a velocidades tan bajas, ya que, en regímenes inferiores a 1300 rpm, el valor obtenido de este contador oscilaba excesivamente entre 900 y 1300 rpm cuando el rotor no cambiaba la velocidad de giro.

A continuación, se pretende estimar de la forma más realista posible el valor de  $c_t$  con el principal objetivo de que los resultados empíricos se correspondan con la aproximación lineal del empuje en el mayor rango de velocidades angulares posibles.

Debido a que ya se ha demostrado como obtenemos a la aproximación lineal del empuje mediante ajustes por mínimos cuadrados en el apartado 4.1.4, para el análisis del octocóptero vamos a ir directamente a los resultados obtenidos en la Figura 37, dónde aparecen superpuestos a los resultados experimentales del ensayo las diferentes aproximaciones del coeficiente de empuje,  $c_t$ , dependiente del método de ajuste elegido.

A continuación, se describe el método de ajuste utilizado para obtener cada aproximación del coeficiente  $c_t$ :

1. En verde tenemos la aproximación obtenida mediante la media ponderada de todos los coeficientes obtenidos en función del régimen de giro donde  $c_t = 0.000038$  y por lo tanto la expresión del empuje sería:  $T = 0.0000393 w^2$
2. En naranja tenemos la aproximación obtenida mediante el ajuste lineal por mínimos cuadrados donde  $c_t = 0.0000492$  y por lo tanto la expresión del empuje sería:  $T = 0.000049 w^2$
3. En gris tenemos la aproximación obtenida mediante el ajuste polinómico de segundo grado por mínimos cuadrados donde  $c_t = 0.0000415$  y por lo tanto la expresión del empuje sería:  $T = 0.0000415 w^2$
4. En amarillo tenemos la aproximación obtenida mediante el ajuste polinómico de tercer grado por mínimos cuadrados donde  $c_t = 0.0000413$  y por lo tanto la expresión del empuje sería:  $T = 0.0000413 w^2$
5. En negro casi superpuesta con la verde tenemos la aproximación obtenida mediante el ajuste polinómico de cuarto grado por mínimos cuadrados donde  $c_t = 0.0000391$  y por lo tanto la expresión del empuje sería:  $T = 0.0000391 w^2$

Las conclusiones a las aproximaciones mediante los ajustes propuestos son claras, la aproximación mediante el valor medio de los coeficientes obtenidos experimentalmente dista mucho del comportamiento real del motor subestimando el empuje de este desde velocidades de giro relativamente bajas.

Respecto a la aproximación mediante los ajustes polinómicos son muy representativos de la realidad cuando el motor gira en regímenes medio-bajos, pero cuando el motor supera las

3500 rpm el empuje resultante de la aproximación es mucho inferior al que realmente proporciona el motor.

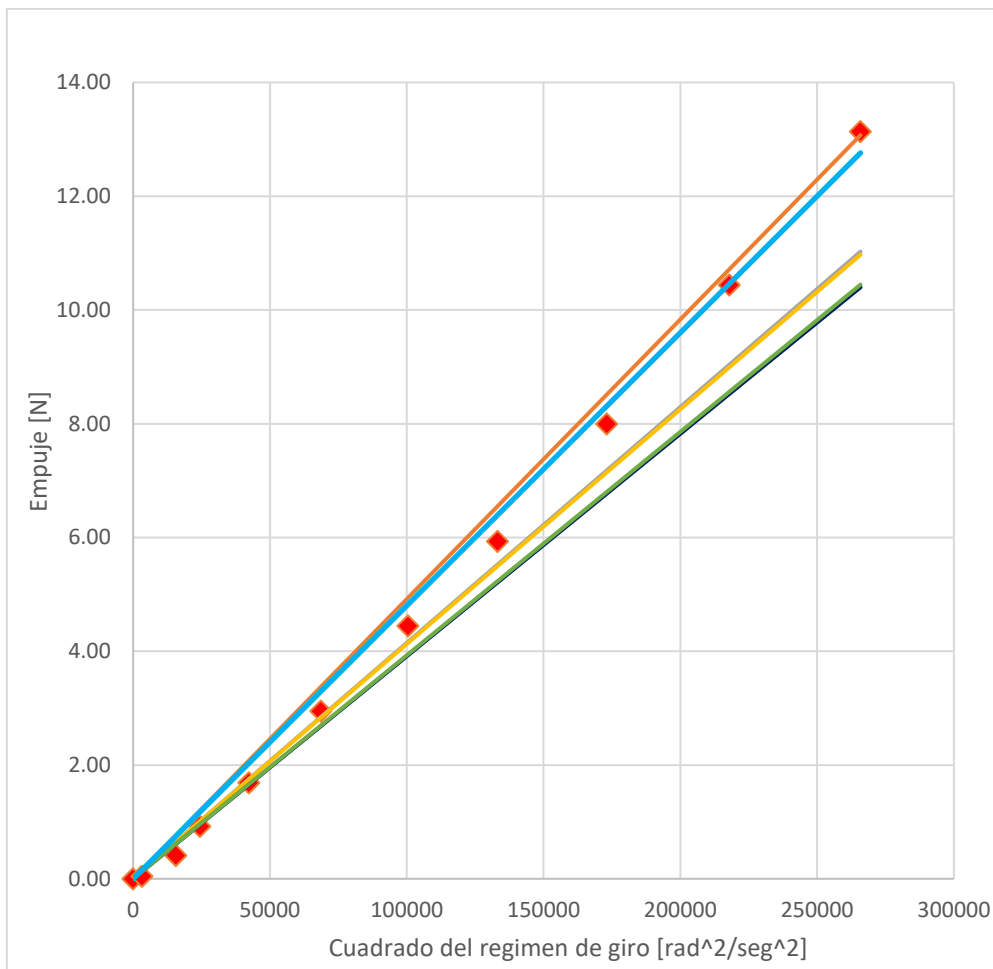


Figura 37. Conjunto de aproximaciones lineales de la ley de empuje obtenidas tras diferentes ajustes.

Por lo tanto, tal y como ha ocurrido con la planta propulsora del hexacóptero el enfoque más fidedigno a la realidad según la hipótesis de que el empuje tiene una relación lineal con el cuadrado de la velocidad angular, vuelve a ser la aproximación del coeficiente de empuje,  $c_t$ , derivada del ajuste lineal por mínimos cuadrados de los datos experimentales recogidos en el ensayo.

No obstante, aunque esta aproximación es la más adecuada de los ajustes propuestos, no es realmente buena, ya que, sobreestima el valor del empuje del conjunto motor penalizando prácticamente por igual a todo rango de velocidades angulares.

Se propone una solución intermedia entre el ajuste lineal y el ajuste de segundo grado, con el objetivo de que la curva obtenida se corresponda lo máximo posible con los datos experimentales, por lo tanto, el valor del coeficiente intermedio entre los dos ajustes sería  $c_t = 0.000048$ , pendiente de la recta azul de la Figura 38.

Por consiguiente, el empuje que proporciona este rotor en función de su velocidad angular de giro queda determinado por la Ecuación 51.

$$T = 0.000048 \omega^2$$

Ecuación 51. Ley de empuje asumida para el motor MT4006-13

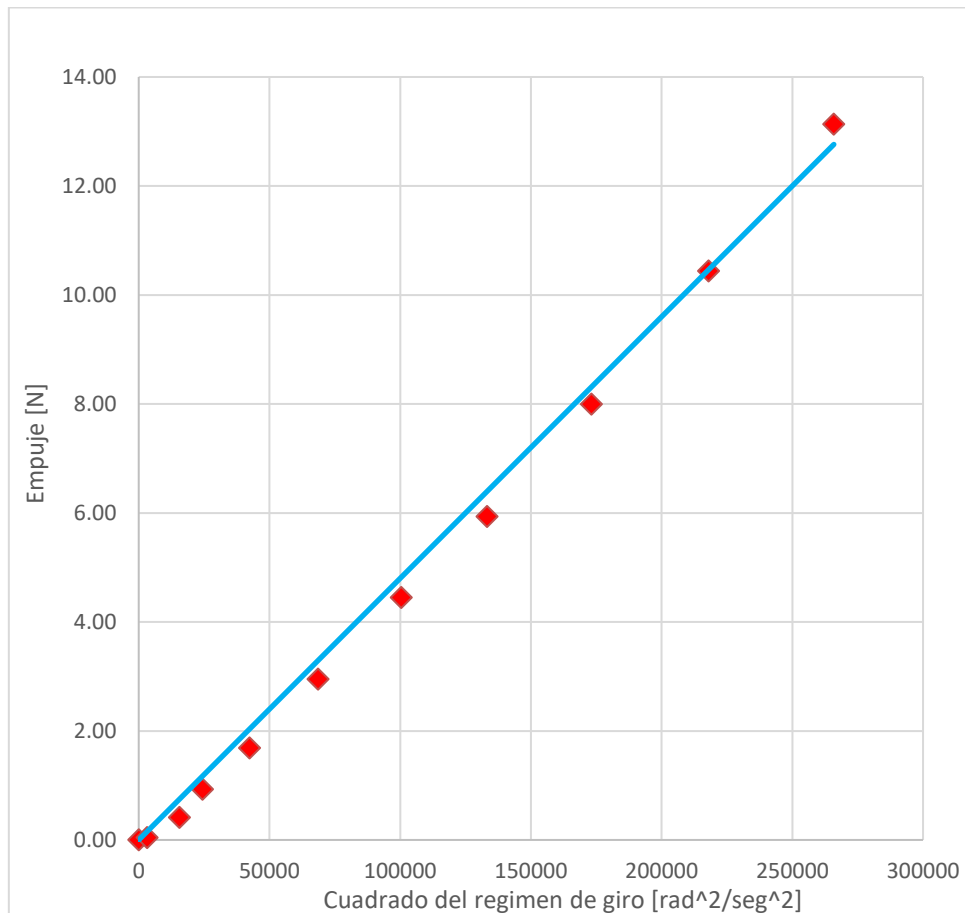


Figura 38. Representación de la ley de empuje asumida para el motor MT4006-13 junto a los datos experimentales.

## Chapter 5. Mejora del modelo

### 5.1 Modelo experimental de las baterías

El objetivo final de crear un modelo de la batería es simular de la manera más realista posible la autonomía del vehículo de la que dispone el piloto antes de aterrizar.

Del mismo ensayo comentado en el apartado anterior se ha recogido los parámetros eléctricos de consumo y potencia de la batería que alimenta a los motores con el fin de modelar en Simulink el consumo de esta de una manera realista.

Por lo tanto, a la vez que se recogían los distintos valores de empuje suministrados por el motor en función del régimen de giro se han recogido las siguientes variables eléctricas de la batería:

1. Intensidad de corriente en amperios, [A].
2. Potencial de la batería en voltios, [V].
3. Potencia de consumo en vatios, [W].
4. Cantidad de carga eléctrica suministrada por la batería, en miliamperios hora [mAh]

La cantidad de carga eléctrica se mide según el sistema internacional mediante el Culombio, el cual se define como la cantidad de carga que pasa por la sección transversal de un conductor eléctrico en un segundo, cuando la corriente eléctrica es de un amperio.

Por lo tanto, la cantidad de carga se puede definir según la Ecuación 52.

$$Q = I \cdot t$$

*Ecuación 52. Definición de la cantidad de carga*

Sin embargo, en las baterías es más habitual el uso del amperio-hora [Ah] o miliamperio-hora [mAh] para designar la cantidad de carga eléctrica de la que dispone una batería.

Este valor característico de la batería representa la cantidad de electricidad que atraviesa un conductor en una hora, momento en el cual la batería estará completamente agotada. Se utiliza para valorar la capacidad de la batería, es decir una medida de la cantidad de electricidad que puede almacenar durante la carga y liberar durante la descarga.

La capacidad de la batería entonces representa la corriente que puede fluir desde la batería durante un determinado tiempo, esto quiere decir que por ejemplo una batería de 6000mAh, como la del vehículo hexacóptero, será capaz de suministrar una corriente de 6 amperios durante una hora o de 600 miliamperios durante 10 horas, por lo tanto, la autonomía de la batería dependerá tanto de la capacidad de la batería como de la intensidad del circuito que alimenta.

En [11] se expone un método para calcular la autonomía de una misión típica de un dron donde se estima el valor medio de la corriente que alimenta el motor y haciendo uso de la Ecuación 53.

$$Autonomia [min] = \frac{C_T [mAh] \cdot 60}{I_m [A] \cdot 1000}$$

*Ecuación 53. Autonomía media de una misión típica de un multirrotor*

Donde  $C_T$  es la capacidad a plena carga de la batería expresada en [mAh] e  $I_m$  es el valor de corriente medio de la misión estudiada medida en [A].

Pero este cálculo resulta muy limitante y no es del todo preciso, por lo que en este trabajo se ha decidido ir un paso más adelante, ya que tenemos la posibilidad de integrar nuestro consumo de batería en cada iteración de la simulación.

El consumo de corriente de una batería expresada en forma integral viene definido por la Ecuación 54.

$$C = \int I(t) \cdot dt$$

*Ecuación 54. Consumo de corriente de una batería*



Conociendo el valor de la corriente que consume el conjunto de rotores del vehículo podemos estimar el consumo de la batería en función del tiempo. Sabiendo la capacidad total de la batería  $C_T$  se puede estimar un porcentaje de carga de esta a través de la Ecuación 55. Por lo tanto, cuando  $C = C_T$ , la batería estará completamente descargada.

$$\%carga = \left(1 - \frac{C}{C_T}\right) \cdot 100 = \left(1 - \frac{\int I(t) \cdot dt}{C_T}\right) \cdot 100$$

*Ecuación 55. Porcentaje de carga remanente de una batería*

### 5.1.1 Hexacóptero

En este apartado se presentan los datos necesarios para construir el modelo de descarga de la batería del vehículo hexacóptero, es decir, a través del mismo ensayo repetido tres veces dónde la velocidad angular del motor va aumentando en 500 rpm cada minuto aproximadamente hemos recogido todos los parámetros eléctricos de la batería comentados cuando introducimos la bancada de ensayos en el apartado 4.1.4.

La batería del vehículo se trata de una batería LiPo de 4 celdas de la marca HRB con una capacidad total de 6000mAh, un voltaje nominal de 14.8v y una capacidad de carga de 50C.



*Figura 39. Baterías HRB tipo LiPo a bordo del hexacóptero real.*

Tras haber recogido del ensayo más de 20 muestras para cada uno de los parámetros eléctricos de la batería a regímenes de giro en torno a las 1000, 1500, 2000, 2500, 3000, 3500, 4000, 4500, 5000, 5500, 6000 y 6500 rpm, obtenemos la siguiente tabla haciendo la media ponderada de estas muestras.

| <b>w</b><br><b>[rpm]</b> | <b>Intensidad</b><br><b>[A]</b> | <b>Potencia</b><br><b>[W]</b> | <b>Voltaje</b><br><b>[V]</b> | <b>Eficiencia</b><br><b>[g/W]</b> |
|--------------------------|---------------------------------|-------------------------------|------------------------------|-----------------------------------|
| 0                        | 0.00                            | 0                             | 0                            | 0                                 |
| 1058                     | 0.19                            | 2.65                          | 16.93                        | 6.52                              |
| 1530                     | 0.28                            | 4.36                          | 16.92                        | 8.86                              |
| 2015                     | 0.50                            | 8.00                          | 16.85                        | 8.87                              |
| 2481                     | 0.80                            | 13.00                         | 16.87                        | 8.93                              |
| 3016                     | 1.27                            | 20.65                         | 16.83                        | 8.28                              |
| 3518                     | 1.88                            | 31.09                         | 16.78                        | 7.85                              |
| 3987                     | 2.67                            | 44.26                         | 16.72                        | 7.30                              |
| 4503                     | 3.77                            | 61.87                         | 16.63                        | 6.63                              |
| 5031                     | 5.30                            | 87.20                         | 16.45                        | 5.91                              |
| 5511                     | 6.88                            | 111.65                        | 16.35                        | 5.61                              |
| 6051                     | 9.04                            | 145.63                        | 16.10                        | 5.31                              |
| 6498                     | 11.39                           | 179.70                        | 15.85                        | 4.99                              |
| 6990                     | 15.19                           | 235.92                        | 15.35                        | 4.41                              |

*Tabla 5. Datos experimentales de la batería HRB obtenidos del ensayo*

A medida que aumenta la velocidad de giro del rotor en torno al 100% del “throttle” se alcanzan condiciones en las que el motor sufre y puede peligrar su integridad. Por este motivo no se ha querido llegar a estas velocidades en el ensayo por lo que el último valor de la Tabla 5 ha sido obtenido extrapolando los resultados experimentales mostrados en la Figura 40. Para ello se hace un ajuste polinómico de grado 4 mediante mínimos cuadrados obteniéndose la ecuación que aproxima la curva obtenida experimentalmente.

Por lo tanto, teóricamente la intensidad de la corriente que alimenta el motor en función de la velocidad angular de esta vendrá determinada por la Ecuación 56:

$$I = 1.55519040E-14w^4 - 1.73626105E-10w^3 + 1.00017978E-06w^2 - 1.84265535E-03w + 1.25126981E+00$$

*Ecuación 56. Expresión del ajuste de cuarto grado de los datos experimentales del consumo de la batería HRB*

Siguiendo esta aproximación a una velocidad de angular de 6990 rpm, punto de saturación del motor del hexacóptero, este consumirá una corriente de 15.19 amperios.

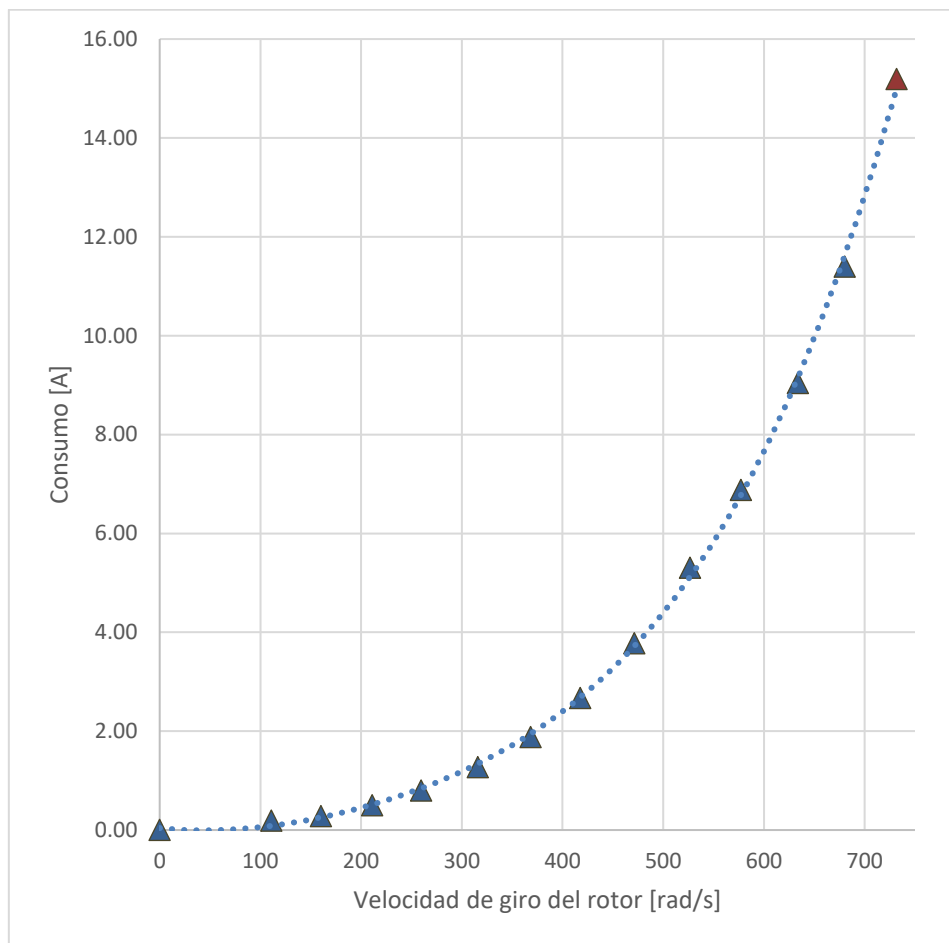


Figura 40. Ajuste polinómico de cuarto grado de los datos experimentales del consumo de la batería HRB

Como se ha comentado en la introducción de este apartado sabiendo la intensidad de corriente que pasa por el motor para un determinado régimen de giro queda determinado el consumo de la batería.

El valor de potencia al régimen de giro de saturación del motor también se obtiene también extrapolando los resultados experimentales, pero esta vez de la Figura 41, donde se muestra la evolución de la potencia de salida de la batería en función de la intensidad de corriente del circuito.

La línea de ajuste polinómico de segundo grado por mínimos cuadrados sigue la Ecuación 57:

$$W = -0.0939 I^2 + 16.956 I - 0.4648$$

Ecuación 57. Expresión del ajuste de segundo grado de los datos experimentales de la potencia de la batería HRB

Por lo tanto, sabiendo que, a 6990 rpm, la corriente mide 15.19 amperios, valor extrapolado de la Figura 41, la potencia que consume el motor será de aproximadamente 236 vatios.

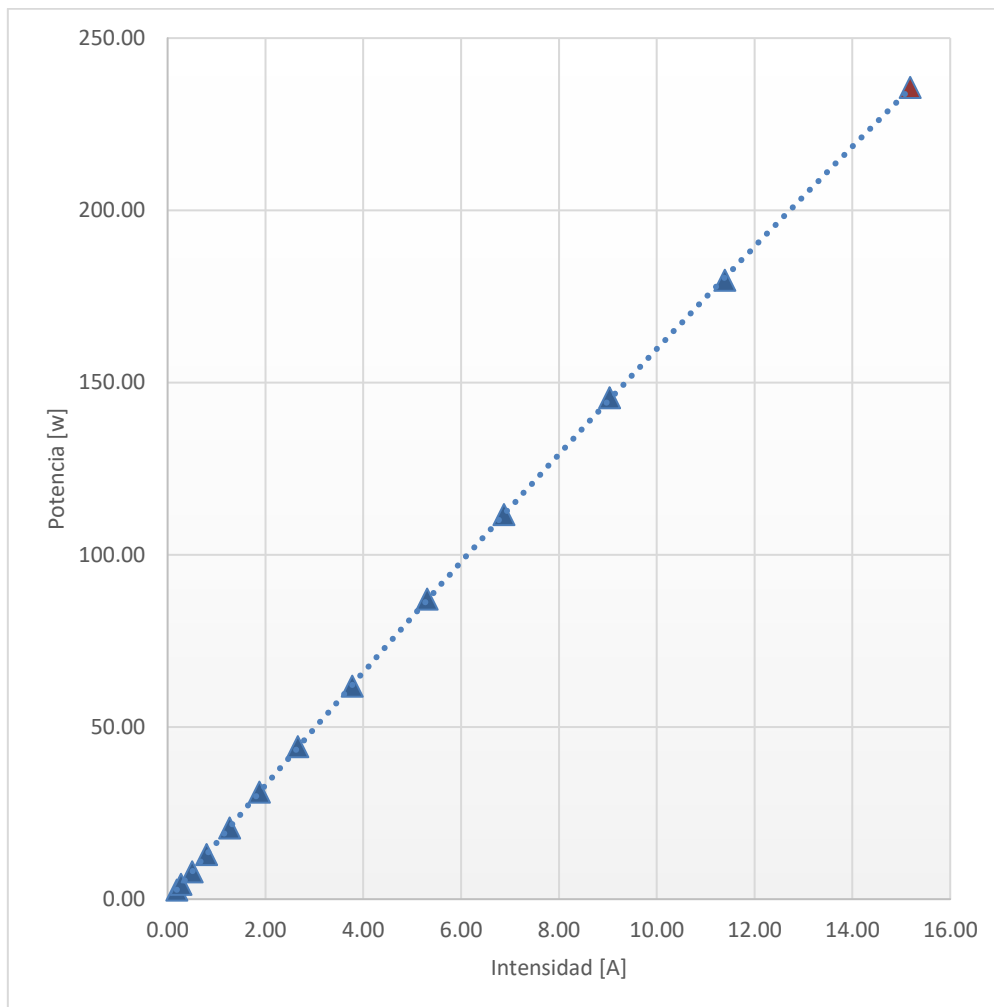


Figura 41. Ajuste polinómico de segundo grado de los datos experimentales de la potencia de la batería HRB

En la Figura 42 se muestra la evolución directamente proporcional del consumo en intensidad de corriente con el empuje suministrado por el motor. Esto tiene sentido ya que cuanto más empuje solicite el vehículo más rápido deberá girar el rotor, cuanto más rápido necesite girar más potencia necesitará consumir de la batería, y sabemos de la Figura 41 que una mayor potencia lleva asociada una mayor intensidad de corriente lo cual hace aumentar el ratio de consumo de la batería y bajar su autonomía.

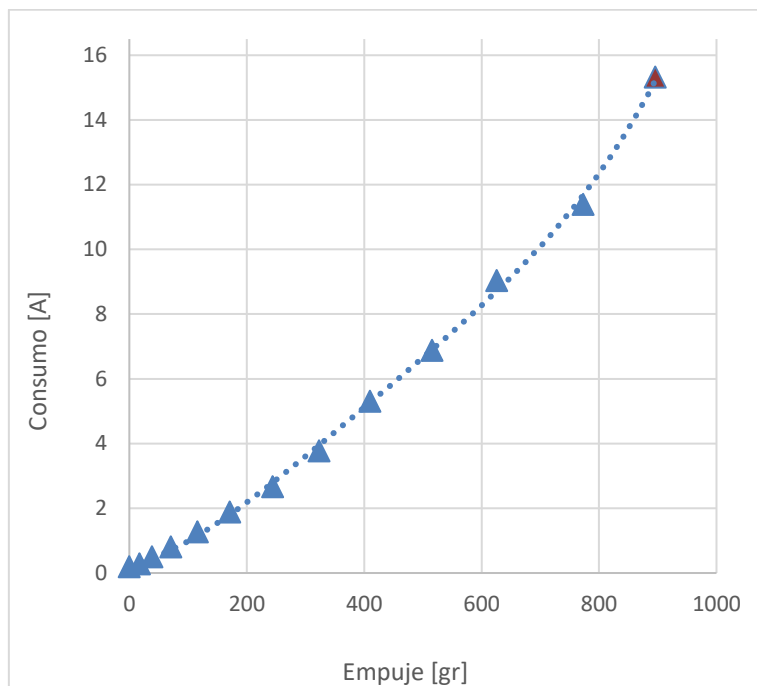


Figura 42. Curva de consumo de la batería HRB en función del empuje que suministra el motor MT4006-13

Respecto a la eficiencia energética del motor se hace un análisis del parámetro empuje/potencia para cada régimen de giro estudiado, lo que realmente indica cuantos gramos de empuje es capaz de proporcionar ese rotor por cada vatio de potencia que le entra. En la Figura 43 podemos ver el carácter exponencial de esta gráfica donde la potencia necesaria solicitada por el motor aumenta más rápido cuanto más rápido gire el motor.

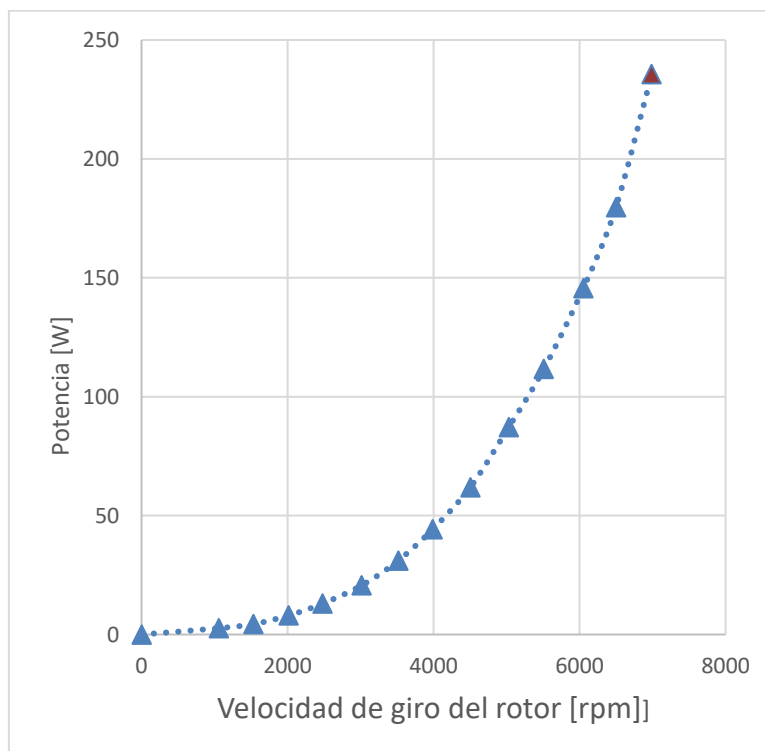


Figura 43. Potencia consumida por el motor MT4006-13 en función del régimen de giro

El valor de eficiencia depende en gran medida del régimen de giro del motor. En la figura podemos ver que los máximos niveles de empuje por unidad de potencia se van a obtener a bajas revoluciones.

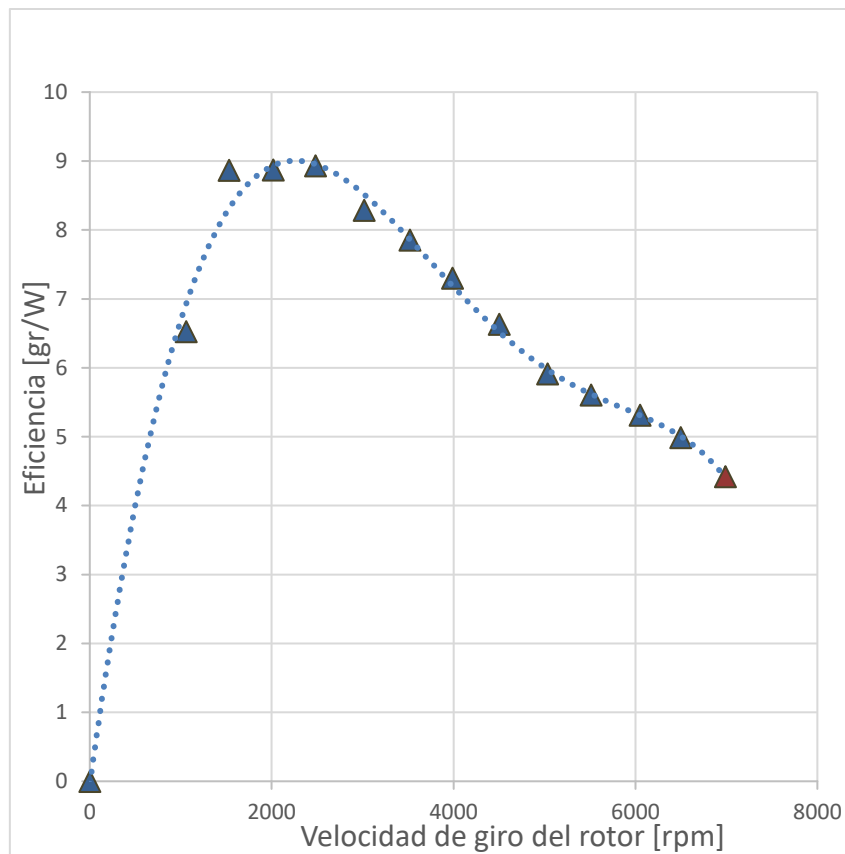


Figura 44. Eficiencia empuje/potencia en función del régimen de giro del motor MT4006-13

A bajos regímenes de giro el motor es capaz de aumentar su empuje consumiendo poca potencia, a medida que aumenta el régimen de giro en torno a 4000 rpm, en la Figura 44 podemos ver como este valor de eficiencia decrece hasta ser mínimo en el régimen de giro máximo.

Este comportamiento se debe a que a velocidades de giro muy elevadas el consumo de potencia y por lo tanto la corriente del circuito aumentan de manera exponencial haciendo al sistema motor de este vehículo hexacóptero comportarse de una manera mucho menos eficiente.

### 5.1.2 Octocóptero

La batería proporcionada por el tutor del trabajo y que se pretende introducir al modelo Matlab-Simulink del octocóptero se trata de un batería tipo LiPo de 4 celdas de la marca RCInnovations con 8000 mAh de capacidad máxima, un voltaje nominal de 14.8 voltios y 30C de capacidad de carga.



Figura 45. Baterías RC tipo LiPo a bordo del octocóptero real.

Tras el mismo ensayo explicado en apartados anteriores se obtienen los valores de los parámetros eléctricos de la batería para regímenes de giro próximos a 500, 1000, 1500, 2000, 2500, 3000, 3500, 4000 y 4500 rpm.

| w [rpm] | I[A]  | Potencia [W] | Voltaje | Eficiencia [g/W] |
|---------|-------|--------------|---------|------------------|
| 0       | 0.00  | 0.0          | 0.0     | 0.0              |
| 544     | 0.11  | 1.2          | 16.9    | 3.65             |
| 966     | 0.20  | 3.0          | 16.8    | 3.55             |
| 1491    | 0.80  | 13.0         | 16.8    | 7.26             |
| 1965    | 1.46  | 24.5         | 16.7    | 7.04             |
| 2502    | 2.61  | 43.1         | 16.6    | 6.98             |
| 3027    | 4.21  | 68.8         | 16.4    | 6.59             |
| 3485    | 6.27  | 101.6        | 16.3    | 5.96             |
| 3973    | 9.44  | 149.8        | 15.9    | 5.45             |
| 4458    | 13.78 | 214.2        | 15.5    | 4.97             |
| 4924    | 19.48 | 298.6        | 15.4    | 4.49             |

Tabla 6. Datos experimentales de la batería RC obtenidos del ensayo

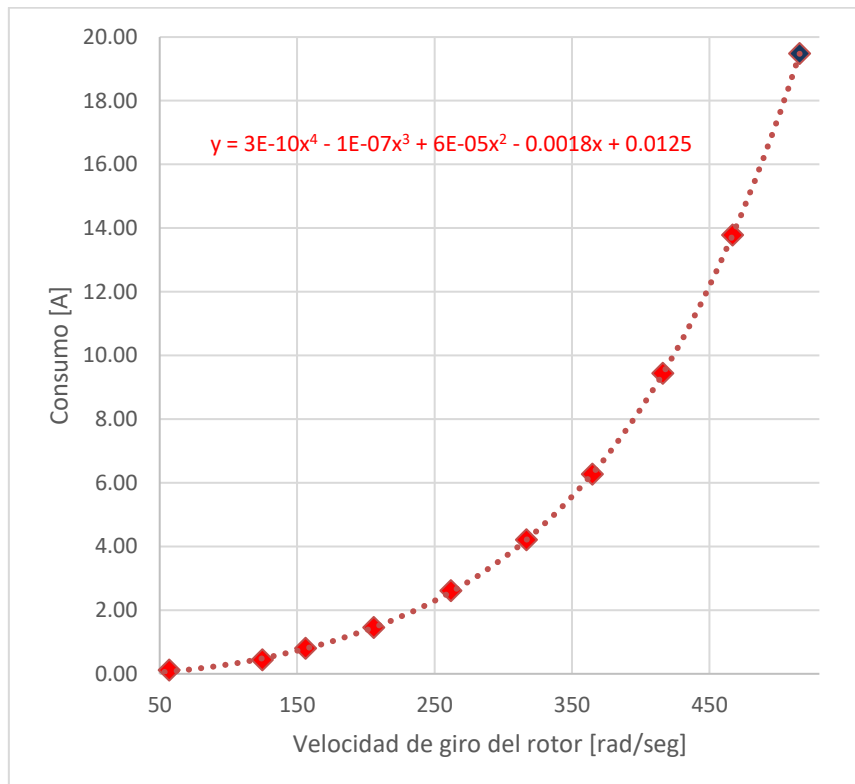
Al ser peligroso llegar al régimen de giro máximo los valores de esto parámetros en torno a las 4900 rpm se han obtenido extrapolándolos resultados experimentales extraídos del ensayo.

Para el valor de la intensidad se ha aproximado mediante un ajuste polinómico de cuarto grado por mínimos cuadrados la Figura 46, obteniéndose la expresión:

$$I = 3.408E-10 w^4 - 1.369E-07 w^3 + 5.671E-05 w^2 - 1.815E-03w + 1.249E-02$$

*Ecuación 58. Expresión del ajuste de cuarto grado de los datos experimentales del consumo de la batería RC*

Por lo que para  $w = 4924$  rpm, valor de saturación del motor MN4014-9 obtenemos una intensidad de corriente aproximada de 19.48 amperios.



*Figura 46. Ajuste polinómico de cuarto grado de los datos experimentales del consumo de la batería*

Para obtener la potencia a estas elevadas revoluciones se ha aproximado su valor mediante la expresión obtenida también tras un ajuste polinómico de tercer grado por mínimos cuadrados la figura. Esta expresión es:

$$W = 0.0131 I^3 - 0.3451 I^2 + 17.955I - 0.9985$$

*Ecuación 59. Expresión del ajuste de tercer grado de los datos experimentales de la potencia de la batería RC*



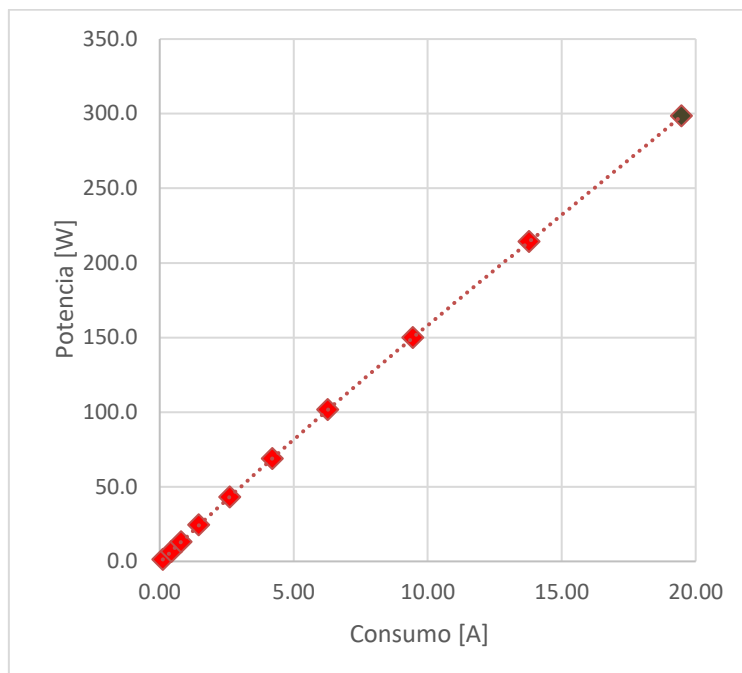


Figura 47. Ajuste polinómico de tercer grado de los datos experimentales de la potencia de la batería RC

Por lo tanto, para una  $w=4924$  rpm, el consumo es de aproximadamente 19.48 amperios, por lo que, según la Ecuación 59, la potencia consumida por el motor será de aproximadamente 300 vatios.

Tanto el consumo en función del empuje como la evolución de la potencia con el régimen de giro siguen una tendencia lineal y exponencial, respectivamente, tal y como ocurría con la batería del vehículo hexacóptero. Ver Figura 48 y Figura 49.

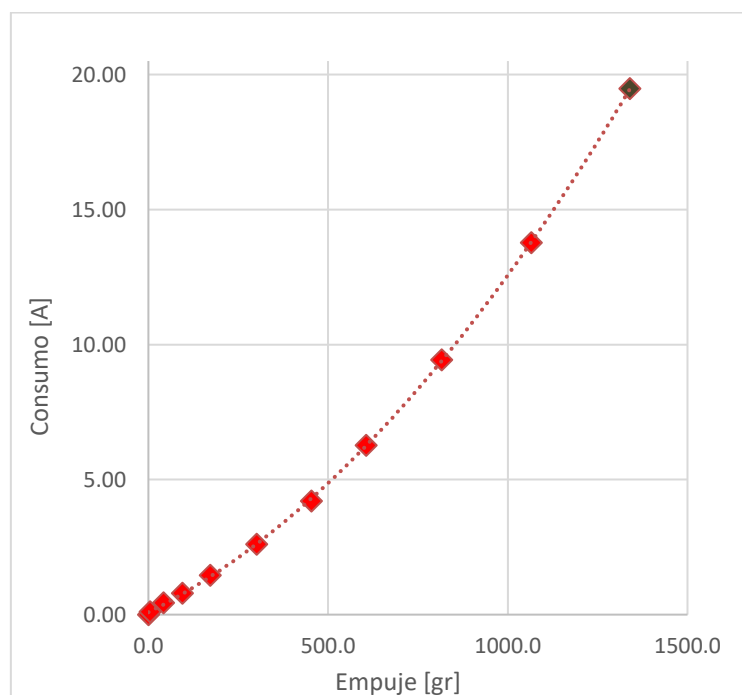


Figura 48. Curva de consumo de la batería RC en función del empuje que suministra el motor MN4014-9

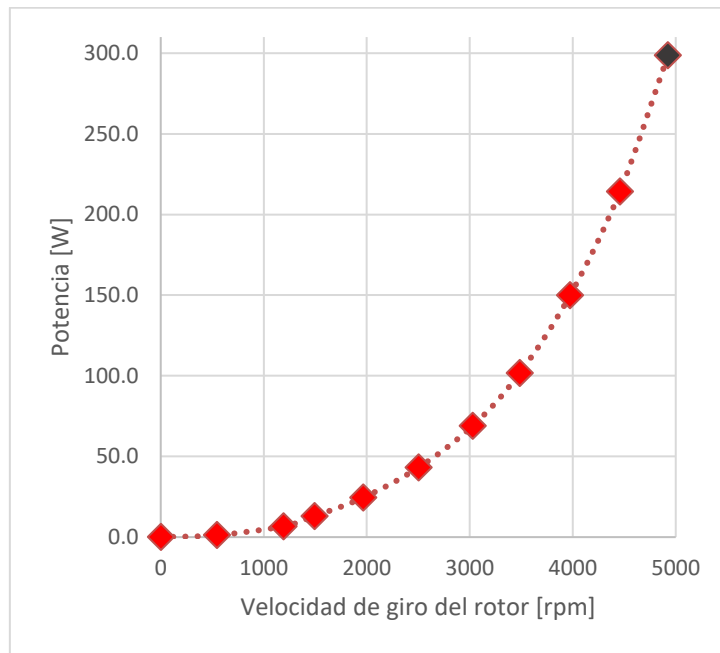


Figura 49. Potencia consumida por el motor MN4014-9 en función del régimen de giro

Respecto a la eficiencia de este motor podemos ver en la Figura 50 que la máxima eficiencia energética se conseguirá a regímenes de giro medio bajos en torno a las 2000 rpm. de giro, mientras que desciende a medida que nos acercamos al punto de saturación del motor.

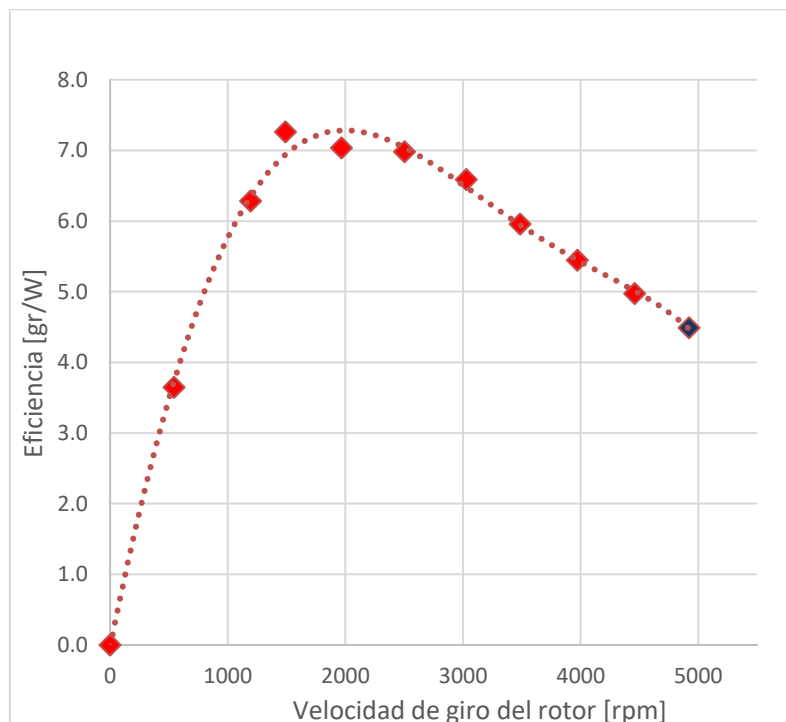


Figura 50. Eficiencia empuje/potencia en función del régimen de giro del motor MN4014-9

Una vez caracterizada experimentalmente ambas baterías es importante remarcar que cuando implementemos el modelo en Simulink en vez de interpolar directamente sobre los datos experimentales, se utilizará una línea de ajuste lo más representativa de la realidad posible.

## 5.2 Efecto suelo basado en altura no en altitud

El modelo de multirroto heredado en este proyecto estimaba el incremento de empuje según la Ecuación 26 utilizando la altitud  $Z$ , elevación del vehículo respecto a el plano  $XY$  de nuestro sistema de referencia fijo  $\{0\}$ .

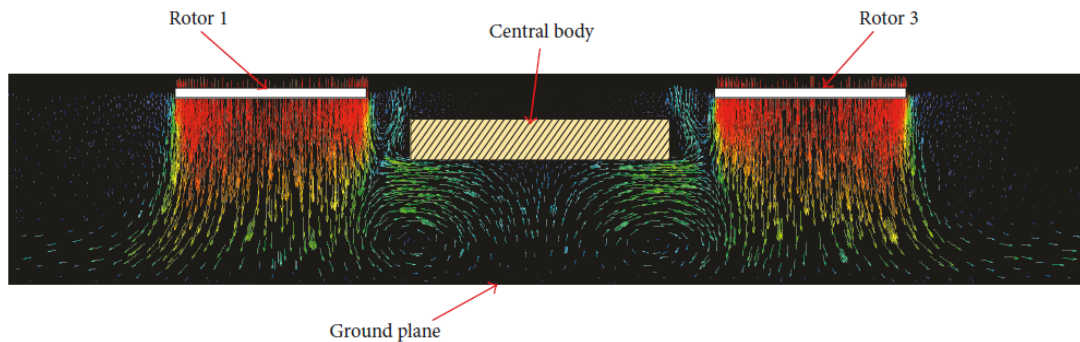


Figura 51. Flujo de aire alrededor de un cuadricóptero volando cerca del suelo

Si bien esto es correcto cuando el dron vuela cerca del suelo, limita bastante el carácter realista de la simulación cuando el dron vuela por encima de una superficie situada en un plano paralelo al  $XY \{0\}$  pero a una altura  $Z \{0\} > 0$ .

Físicamente el efecto suelo que genera esta superficie es el mismo, ya que se genera una “almohadilla” de aire de mayor presión por debajo del dron que hace que el empuje de los rotores se vea incrementado, pero con la expresión heredada del anterior modelo, basada en  $Z$ , este efecto no se hacía efectivo ya que esta altura no era inferior a 10 veces el radio de la hélice, por lo tanto, esta información no pasaba al modelo dinámico y este efecto pasaba desapercibido en estas situaciones.

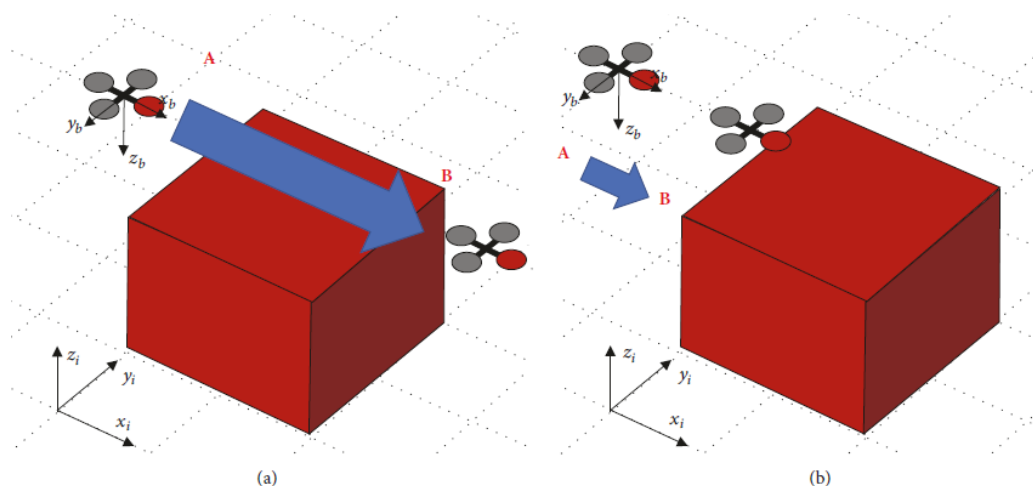


Figura 52. Esquema de efecto suelo basado en elevación

Tras analizar este fallo del modelo, decidimos colaborar con otro alumno de la universidad y gracias a la implementación del modelo en Unity, detallada en [13] conseguimos basar la evolución de la Ecuación 26 en una variable mucho más realista para este efecto físico, “Above Ground Level”.

Unity permite calcular la distancia de un punto del espacio a la superficie sólida más cercana en dirección vertical, este punto del espacio será el centro de masa del vehículo y la variable obtenida hemos decidido llamar AGL (Above Ground Level). Con esta variable y gracias al módulo UDP\_receiver implementado en Simulink, para más información sobre este módulo el lector puede consultar [13], recibimos en cada iteración la distancia vertical del dron a la superficie sólida más cercana.

Simplemente sustituyendo en la Ecuación 26 el valor de Z por el de AGL, conseguimos tener un efecto suelo mucho más realista que actúa sobre el modelo dinámico para cualquier valor de altitud del vehículo.

### 5.3 Condición suelo

La condición suelo desarrollada en este trabajo principalmente lo que pretende es que el multirrotor se mantenga estático cuando esté reposando sobre una superficie sólida, como puede ser el suelo o cualquier superficie creada en el entorno gráfico Unity.

La simulación determinará que el vehículo está apoyado sobre una superficie a través de dos parámetros principalmente, los cuales se envían desde el motor gráfico Unity desarrollado en [13].

Estos parámetros son específicamente dos: WoW, “weight on wheels” y AGL, “Above ground level”

Cuando estas dos variables alcanzan ciertas condiciones, explicadas con detalle en el apartado 6.2.2.4.1.4, la condición suelo se activa en el modelo.

Las principales consecuencias sobre el modelo cuando se alcanza la condición suelo son:

1. Momentos de reacción nulos  $\vec{\tau} = (0 \ 0 \ 0)$
2. Del vector fuerzas totales desaparece la fuerza del viento y la fuerza de gravedad
3. Vector velocidad nulo:  $\vec{v} = (0 \ 0 \ 0)$
4. Coordenada Z {0} del vehículo constante  $Z_0 = cte$
5. Velocidad angular ejes móviles nula:  $\vec{\omega} = \vec{0}$
6. Ángulos de “roll” y “pitch” nulos:  $\psi = 0 ; \theta = 0$

En el 6.2.2.4.1.4 se continúa con el análisis de esta condición suelo, desde un punto de vista más relacionado con su implementación en Simulink.

### 5.4 Fallo rotor

A raíz de añadir el modelo del hexacóptero y el octocóptero en este proyecto se ha considerado la posibilidad de hacer que uno o más rotores del vehículo fallen y dejen de girar.

Esta condición de fallo en un cuadricóptero haría perder completamente el control del vehículo, mientras que en drones con mayor número de rotores aún se puede mantener un control básico pero limitado de la actitud del vehículo. Otra funcionalidad que se ha implementado y que se explicará en detalle en el 6.2.2.3 es la posibilidad de realizar un aterrizaje de emergencia cuando se considere que el control del dron es inestable.

La forma de simular que un rotor falle es sencilla si se conoce la matriz de control del vehículo. Como se ha comentado en el 3.2 el vector velocidades angulares  $W^2$  se obtiene según la Ecuación 43 donde cada columna de la matriz de control A representa la contribución de cada rotor al empuje, par de arrastre, par de alabeo y par de cabeceo totales del vehículo. La manera de simular que una de los rotores se ha parado, condición de fallo rotor, es hacer que estas contribuciones al sistema de fuerzas que afecta al vehiculó sean cero, por ejemplo, si en el caso del hexacóptero queremos simular que el rotor 5 se ha parado, el sistema a resolver en el modelo dinámico simplificado vendría dado por la Ecuación 60.

$$\begin{pmatrix} T \\ \tau_x \\ \tau_y \\ \tau_z \end{pmatrix} = \begin{pmatrix} c_t & c_t & c_t & c_t & 0 & c_t \\ c_t d \sin \alpha_1 & c_t d \sin \alpha_2 & c_t d \sin \alpha_3 & c_t d \sin \alpha_4 & 0 & c_t d \sin \alpha_6 \\ -c_t d \cos \alpha_1 & -c_t d \cos \alpha_2 & -c_t d \cos \alpha_3 & -c_t d \cos \alpha_4 & 0 & -c_t d \cos \alpha_6 \\ -c_q & c_q & -c_q & c_q & 0 & c_q \end{pmatrix} \begin{pmatrix} w_1^2 \\ w_2^2 \\ w_3^2 \\ w_4^2 \\ w_5^2 \\ w_6^2 \end{pmatrix}$$

*Ecuación 60. Sistema de ecuaciones del modelo simplificado para un hexacóptero con el quinto rotor averiado*

Así conseguimos que la contribución del rotor que ha fallado a las fuerzas que actúan sobre el dron sea nula, pero no solo eso, aunque la velocidad angular del rotor averiado podría ser cualquier valor debido a que la solución a este sistema según la Ecuación 43 se obtiene según el método de mínimos cuadrados, esta velocidad angular va a ser la mínima posible y por lo tanto se consigue el objetivo de simular la parada del rotor.

En el caso del hexacóptero al simular esta condición de fallo en situación de “hover”, régimen estacionario del dron en vuelo, para compensar el par de arrastre o momento guiñada que supone esta perturbación, el sistema encuentra automáticamente la solución de apagar el motor opuesto para cumplir la orden del “hover”:  $\vec{H} = (Mg \ 0 \ 0 \ 0)$ .

En el caso del “hover”, esta condición de fallo vendrá precedida del incremento de las velocidades angulares de los rotores operativos, ya siendo este número de rotores inferior la orden sigue siendo la misma: contrarrestar el peso del vehículo para mantenerlo estacionario.

Los motores tienen una velocidad máxima de rotación disponible, en el caso del hexacóptero este valor limite está cercano a 7000 revoluciones por minuto, con este valor de saturación junto con la orden establecida por el piloto en modo “hover” de ángulo de guiñada nulo, los cuatro rotores restantes no son capaces de sustentar el peso del dron, 4.178 kg, por lo que bajo esta situación el dron aterrizará automáticamente en un descenso equilibrado.

En el caso del octocóptero al tener una configuración mayor de motores, puede cumplir la orden “hover” cuando un solo rotor a fallado, sin embargo, cuando se simula el fallo consecutivo del segundo rotor pasa algo similar a lo que ocurre con el hexacóptero. La configuración de motores no dispone de la fuerza necesaria para compensar los 5.75 kg que pesa al vehículo a la vez que cumple la orden de ángulo de guiñada nulo.

Se podría desacoplar esta ecuación en la resolución del sistema para comprobar si 6 motores serían capaces de sustentar el octocóptero, pero esto pondría en peligro la integridad del

vehículo ya que seguramente induciría en el vehículo oscilaciones no amortiguadas que finalmente acabarían por desestabilizar el vuelo.

## 5.5 Modo de operación básico y funciones automatizadas

A continuación, se recogen las formas que tenemos de controlar la actitud y posición del vehículo en la simulación, desde el modo manual utilizando un control clásico hasta una serie de funciones automatizadas elaboradas en este trabajo que le conceden al vehículo un tipo de control en régimen estacionario.

Tanto el control manual clásico como el control estacionario se consigue modificando el bloque PID, al fin y al cabo, en este bloque podemos modificar tanto la variable a controlar como el valor que queremos que alcance el dron, por lo tanto, jugando con las entradas a este bloque podemos conseguir actitudes concretas del dron que pueden resultar interesantes y que se comentan en este apartado.

Cabe resaltar que con la ayuda de una interfaz de usuario elaborada en [13] conseguimos cambiar entre los diferentes modos de operación sin necesidad de ejecutar un código Simulink diferente.

### 5.5.1 Modo manual

Si bien el modo manual fue heredado del código Matlab-Simulink de [1] hay que destacar todo el proceso iterativo para ajustar todo el bloque PID de los dos nuevos vehículos implementados en este proyecto: Hexacóptero y octocóptero.

El bloque PID tiene asociadas unas ganancias y unas constantes que afectan a la salida de la variable de control  $H = (T \quad \tau_x \quad \tau_y \quad \tau_z)$ , que en resumen controla toda la actitud y posición del dron, estas variables van directamente relacionadas con los motores que las generan e indirectamente con el peso y las características geométricas del vehículo, por lo tanto las ganancias y constantes que valen para el control manual de un vehículo no valdrán para otro al cambiar la planta propulsora, dimensiones y peso del este.

Aunque existen algoritmos dentro de Simulink que ajustan automáticamente las constantes del control PID, estos solo resultan eficaces en sistemas donde intervienen pocas variables, no obstante en nuestro caso, se ha comprobado que el modelo del multirroto no responde adecuadamente a este “autotuning” de constantes PID por lo que se ha seguido un proceso iterativo a base de prueba de error hasta conseguir un control adecuado, aunque no perfecto, de los nuevos vehículos añadidos al modelo Matlab-Simulink.

En el apartado 6.2.2.1.1 se explicará concretamente cómo se ha implementado este control manual en Simulink mediante los diferentes tipos de controles PID en función de la variable a controlar.

### 5.5.2 Aterrizaje y aterrizaje de emergencia

Se ha añadido la funcionalidad de aterrizar el dron cuando quiera el piloto mediante un botón en la interfaz gráfica desarrollada en [13]. Al accionar este botón el código de Simulink limita las velocidades angulares de los rotores a un valor característico para cada vehículo, lo que

hace que el dron baje de forma lenta y equilibrada sin suponer un peligro para la integridad del vehículo.

También se ha añadido a el modelo la posibilidad de aterrizar bajo condiciones críticas de funcionamiento sin necesidad de que el piloto accione ningún botón. Estas condiciones críticas de funcionamiento hemos decidido que sean:

1. Cuando el nivel de batería es inferior al 15%.
2. Cuando se ha producido un fallo rotor y el dron solo tiene 4 rotores operativos, ya que perder uno más supondría la pérdida de control del dron.
3. Cuando los rotores no son capaces de sustentar el dron.

Todas estas funcionalidades se explicarán en detalle describiendo el código elaborado en el apartado 6.2.2.3.

### **5.5.3 Hover con retroalimentación de altura**

Hasta ahora se ha considerado el “hover” como el estado estacionario en vuelo del dron independientemente de la altura, esto es porque hasta ahora la retroalimentación del bloque PID se hace con la velocidad vertical del dron y con carácter proporcional donde el empuje en posición neutra del joystick busca contrarrestar el peso del dron.

Pero existe otra variable que retroalimentar el PID para conseguir otro tipo de “hover”, esta es retroalimentando la altura, en este tipo de control también necesitas un PID de componente proporcional y te permite desde la interfaz gráfica de Unity indicarle un valor de altura que quieres que alcance el dron.

Este tipo de control se explicará en detalle describiendo sus bloques de Simulink en el apartado 6.2.2.1.2.

### **5.5.4 Avance frontal en vuelo equilibrado**

Esta funcionalidad se ha desarrollado reciclando el bloque PID del control manual y permite al dron avanzar estacionariamente hacia delante o hacia atrás sin perder el control.

La idea de este bloque es muy sencilla, inclinar el dron un cierto ángulo, concretamente 20º en el sentido positivo del “pitch” o ángulo de cabeceo. Esto hará que el empuje tenga cierta componente longitudinal lo que hace que avance hacia adelante el dron.

Si queremos que el dron retroceda basta simplemente con inclinar el dron 20º en el sentido negativo del pitch, obteniéndose el mismo efecto, pero en el sentido contrario.

Cabe destacar que la inclinación del dron hace que el empuje tenga componente horizontal en ejes tierra, pero a cambio de esto se pierde parte de la componente horizontal, por lo que si la constantes PID están configuradas para mantenerse en “hover” esta funcionalidad hará que baje el dron si no se hace nada con la ganancia del empuje. Por este motivo se ha generado un bloque que modifique esta ganancia cuando esta funcionalidad está activada permitiendo al dron avanzar en sentido positivo de X {1} sin perder altitud.

### 5.5.5 Avance lateral en vuelo equilibrado

Esta funcionalidad sigue la misma lógica, pero inclinando el dron en el sentido positivo y negativo de “roll” o ángulo de alabeo.

Por comodidad se ha elegido el mismo ángulo de inclinación que para el avance frontal en ambos sentidos, es decir 20º, ya que así el mismo efecto de pérdida de empuje vertical será resuelto con el mismo bloque que incrementa la ganancia del empuje del control manual.

En el apartado 6.2.2.1.1 se mostrará los bloques Simulink necesarios para implementar estos tipos de control de movimiento horizontal estacionario.

## Chapter 6. Desarrollo en Matlab/Simulink

Este trabajo se centra principalmente en el desarrollo de un simulador de vuelo para tres tipos de multirrotores:

1. Cuadricóptero
2. Hexacóptero
3. Octocóptero

Con ayuda de sus diferentes constructores y del desarrollo de un modelo parametrizado en función del número de rotores que tiene el dron, el mismo modelo desarrollado en simulink servirá para los tres tipos de drones.

A continuación, se van a presentar las diferentes partes del código Matlab y simulink que componen el simulador de vuelo:

### 6.1 MATLAB

#### 6.1.1 Constructores

En los archivos Matlab (.m) que nombramos como “constructor\_(tipo\_de\_dron)”, van a definirse principalmente:

1. Características geométricas del dron: número de rotores, masa, matriz de inercia, distancia y altura de los rotores respecto al centro de masas (CdM), otras características geométricas.
2. Hélice: Cuerda, radio, número de palas, otras características geométricas.
3. Motores: Tipo de motor, tiempo de respuesta, velocidad angular máxima y mínima de giro.
4. Coeficientes aerodinámicos: De empuje, de par de arrastre y de rozamiento con el viento, obtenidos experimentalmente.
5. Parámetros de la batería: Número de baterías, capacidad máxima, gráficas de consumo obtenidas experimentalmente.
6. Matriz de control
7. Parámetros del entorno: Gravedad, densidad y viscosidad del aire, etc.

Como se ha comentado se han considerado 4 configuraciones de vehículos, dos heredadas de [1] y [2]: Cuadricóptero con gimbal y cuadricóptero sin gimbal) y otros dos modelos integrados



durante el desarrollo de este trabajo: Hexacóptero y octocóptero), por lo tanto, tendremos 4 constructores distintos, los cuales se denominan:

1. Constructor\_WOG
2. Constructor\_WG
3. Constructor\_Hexa
4. Constructor\_Octo

### 6.1.2 Archivo de configuración

En este archivo matlab llamado "RunCase\_Multirrotor.m", se declara la variable global "dron" y acto seguido se determinan las constantes de control ( $K_{Thrust}/D_{Thrust}/P_{Thrust}/K_{Yaw}/P_{Yaw}/K_{Roll}$ ...) y se ejecuta el constructor correspondiente al dron elegido desde el motor gráfico Unity.

Además de las constantes necesarias y características de cada multirrotor, se establece la velocidad de muestreo de la simulación,  $dt$ , la cual se fija en 0.0025 segundos, independientemente del dron elegido.

Cabe destacar que la conexión entre las plataformas Simulink-Unity se ha conseguido gracias a la colaboración con el proyecto [13], con la que se ha logrado fusionar el modelo desarrollado en este trabajo con un entorno de simulación atractivo.

En esta plataforma utilizada principalmente como motor gráfico de videojuegos se han desactivado las físicas del movimiento y sirve exclusivamente para plasmar la solución del sistema dinámico que proporciona Simulink, además de tener, como se ha comentado anteriormente, una interfaz de usuario atractiva tanto para visualizar el movimiento como para añadir la selección de funcionalidades comentadas en el apartado 5.5.

La comunicación Simulink-Unity se realiza mediante protocolo UDP y ha sido desarrollada en paralelo al proyecto. Este trabajo se va a centrar principalmente en describir el modelo dinámico y de control, mostrar las nuevas funcionalidades y sofisticaciones realizadas al modelo, pero si el lector desea obtener más información de los módulos: "Enviar datos" y "Recibir datos" puede consultar [13], dónde se explica con detalle esta transferencia UDP de datos entre Simulink y Unity, tanto en una dirección como en otra.

## 6.2 SIMULINK

La simulación se divide en tres módulos claramente diferenciados:

1. Modulo 1: Recibir datos
2. Modulo 2: Modelos dinámico y control
3. Modulo 3: Enviar datos

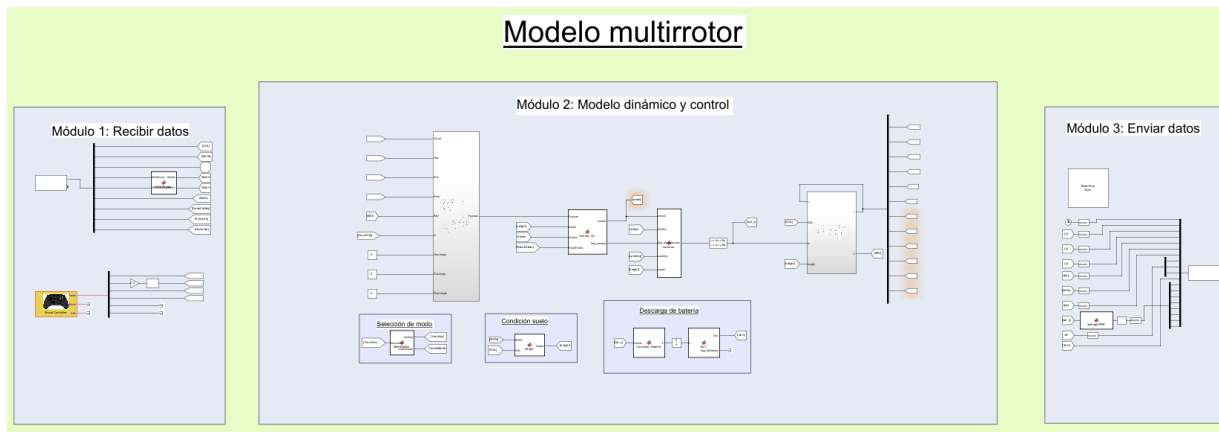


Figura 53. Modelo Simulink del multirroto

### 6.2.1 Módulo 1: Recibir datos

No se va a profundizar en cómo está desarrollado este módulo ya que no ha sido desarrollado en este proyecto, aunque si el lector está interesado puede consultar [13], para más información.

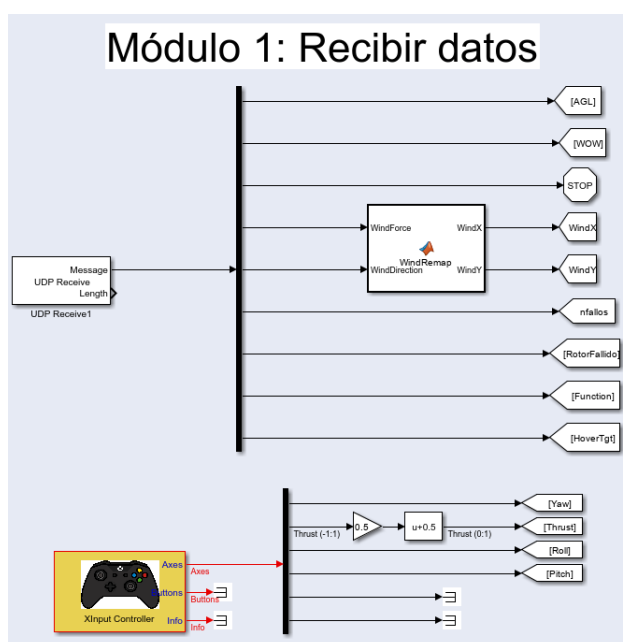


Figura 54. Módulo recibir datos

Lo que si se va a exponer son los parámetros de salida de este módulo de los cuales se nutre el módulo 2, objeto principal del proyecto. Estos parámetros los podemos dividir en dos grupos:

#### 6.2.1.1 Recibidos desde Unity

Entre los parámetros recibidos mediante protocolo UDP desde el motor gráfico Unity, tenemos:

- AGL – Variable tipo “float” que indica la distancia vertical mínima del CdM del dron a el cuerpo a superficie sólida más cercana.
- WoW (Weight on Wheels) – Variable tipo “boolean”, que indica el contacto de las patas del dron con otra superficie,



- c. STOP – Variable tipo “boolean” que para la simulación desde Unity.
- d. Wind X / Wind Y – Variables tipo float que indican la fuerza del viento en la dirección X e Y, respectivamente.
- e. nFallos – Variable tipo “integer” que indica el número de rotores rotos, condición de fallo que se comenta en el apartado 5.4.
- f. RotorFallido - Variable tipo “integer” que indica el primer rotor que se rompe, esta variable se obtiene de forma aleatoria en Unity.
- g. Function –
- h. HoverTgt – Variable tipo “integer” que indica la altura objetivo en la funcionalidad “Automatic hover” que se explica en el apartado 5.5.3.

### **6.2.1.2 Recibidos desde el mando del piloto**

Lo parámetros recibidos desde el mando del piloto van a ser útiles para el modo de control manual, haciendo uso de controladores PID con constantes de control específicas a cada dron, se pretende conseguir un manejo del dron cómodo y versátil.

Estos parámetros son 4, y vienen normalizados antes de entrar en el módulo de control PID.

- i. Thrust: Variable tipo float que va de cero a uno. El empuje generado por los motores será proporcional a esta variable.
- j. Yaw: Variable tipo float que va de menos uno a uno. El ángulo de guiñada llevado a cabo por el dron será proporcional a esta variable. Si este valor es positivo el dron girara en un sentido y si es negativo en el opuesto.
- k. Pitch: Variable tipo float que va de menos uno a uno. El ángulo de pitch llevado a cabo por el dron será proporcional a esta variable. Si este valor es positivo el dron se inclinará en un sentido y si es negativo en el opuesto.
- l. Roll: Variable tipo float que va de menos uno a uno. El ángulo de roll llevado a cabo por el dron será proporcional a esta variable. Al igual que ocurre con la variable pitch, se inclinará en un sentido si es positiva y en el opuesto si es negativa.

Cabe destacar que el mando utilizado para controlar la simulación es un mando de la videoconsola XBOX, por lo que la normalización de las variables va a venir condicionadas por este hecho, es decir a diferencia de un controlador típico de drones donde la posición de reposo del Thrust puede ser cero en nuestro caso siempre será 0.5, esto se debe a que el Joystick de nuestro mando vuelve al centro siempre que se deja libre.

Se podrían modificar las ganancias de la normalización para obtener otro comportamiento, pero se ha considerado conveniente esta característica para conseguir la condición de “hover” en vuelo manual sin tener que tocar el controlador.

## 6.2.2 Modulo 2: Modelo dinámico y control

Como se ha comentado anteriormente, el principal objetivo de este trabajo ha sido ampliar el módulo 2 de modelo de multirroto. Como principales mejoras al modelo heredado de [1] y [2], podemos destacar los siguientes hitos:

1. Parametrización de la simulación para que acepte los modelos del hexacóptero y el octocóptero.
2. Modelo de descarga de batería con datos obtenidos experimentalmente.
3. Efecto suelo.
4. Condición suelo.
5. Fallo rotor
6. Aterrizaje automático
7. Control automático – Funcionalidades básicas de movimiento desactivando el modo manual.

A lo largo de este apartado, según se vayan desglosando los subsistemas se explicará en detalle la forma de implementar cada mejora al modelo.

### 6.2.2.1 Control manual y automático

En este bloque se reciben las variables de control que vienen del mando del piloto (Thrust, Yaw, Roll y Pitch) o vienen de la interfaz gráfica desarrollada en Unity (HoverTgt), así como todas las variables que recoge el bloque IMU, posición, velocidad, orientación del dron. La información proporcionada por el bloque IMU será utilizada por los controladores PID para tener un control estable del dron en la simulación.

La Figura 55 es una representación del bloque “Control” a alto nivel, donde solo aparecen las entradas y salidas del bloque. La salida es un vector de 4 elementos que hemos denominado fuerzas, pero en realidad es el equivalente al vector H de la teoría desarrollada en el apartado 3.2, donde H ha sido denominado como orden de control y viene definido por  $\vec{H} = (T \ \tau_x \ \tau_y \ \tau_z)$ , este vector será consecuencia del bloque PID y se define como valor de las fuerzas y momentos necesarios para alcanzar la orden impuesta por el piloto o por el control por automatismos.

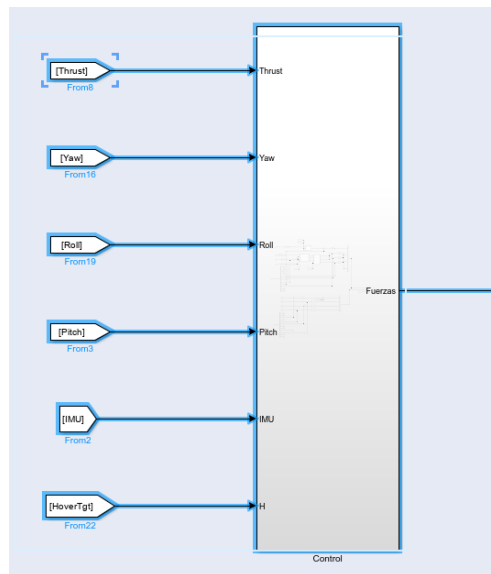


Figura 55. Bloque control

Si analizamos el bloque más a bajo nivel tenemos que empezar a hablar del control manual y de las funcionalidades de control automatizadas que se han añadido al modelo.

#### 6.2.2.1.1 Control manual

El control manual sigue siendo muy similar al heredado de proyectos anteriores:

1. Control del “Thrust” con retroalimentación del modelo de la velocidad vertical en  $Z\{0\}$  del dron.
2. Control del “Yaw” o guiñada con retroalimentación del modelo de la velocidad angular del dron en el eje  $Z\{1\}$ .
3. Control del “Roll” o alabeo con doble retroalimentación del modelo de la velocidad en  $Y\{0\}$  y del propio ángulo de alabeo del dron.
4. Control del “Pitch” o cabeceo con doble retroalimentación del modelo de la velocidad en  $X\{0\}$  y del propio ángulo de cabeceo del dron.

Cabe destacar que las ganancias han tenido que ser ajustadas a los nuevos modelos de vehículo y como novedad se han añadido tres bloques Simulink intermedios antes de la determinación del error de los PIDs.

Estos bloques intermedios son tres:

1. ForwardBackwardMode: Su función principal es desplazar el dron en la dirección  $X\{1\}$  hacia delante y hacia detrás. Esto lo consigue interponiéndose entre los valores de (yaw, roll, pitch) objetivo que comanda el mando y modificándolos a  $(0,0, \pm 0.349)$ , este valor de pitch induce en el dron una inclinación de  $\pm 20^\circ$  girando respecto al eje  $Y\{1\}$ , lo que hace que el dron se desplace en la dirección  $\pm X\{1\}$ .

Esta funcionalidad se comanda desde la interfaz gráfica Unity desarrollada en [13] mediante la variable Function, esta variable entra en el bloque Mode Select y activa la señal discreta FBMode a 1 (sentido positivo de  $X\{1\}$ ) o a  $-1$  (sentido negativo de  $X\{1\}$ ).

Si cualquier otro automatismo o modo de control está activado la señal discreta FBMode estará fija en 0 y este bloque no tendrá efecto ninguno sobre las variables (yaw, roll, pitch).

2. RightLeftMode: Sigue el mismo razonamiento que la función anterior con la salvedad de que ahora el objetivo es desplazar el dron en la dirección  $Y\{1}$  en un sentido y en otro. Por lo tanto, modificando las variables (yaw, roll, pitch) que comanda el mando y a  $(0, \pm 0.349, 0)$ , conseguimos que el dron se incline  $\pm 20^\circ$  respecto al eje  $X\{1}$ , lo que hace que el dron se desplace en la dirección  $\pm Y\{1}$ .

La lógica con la señal discreta RLMMode sigue la misma lógica que FBMode, +1 un sentido, -1 el opuesto y con 0 este bloque no afecta a las señales provenientes del mando del piloto: (yaw, roll, pitch).

3. GainIncrease: Si se analiza la Figura 56 se puede observar que los automatismos ForwardBackwardMode y RightLeftMode no afectan en absoluto al empuje de los motores o Thrust. Esto significa que cuando el dron se incline para avanzar en la dirección solicitada este empuje ganará componente horizontal que induza el desplazamiento requerido, pero consecuentemente perderá componente vertical lo que hará que el dron pierda sustentación y la ganancia establecida para mantener a el dron equilibrado en altura dejará de ser válida.

Por este motivo nace este bloque, el cual induce un incremento de ganancia cuando se activa cualquiera de los dos automatismos, al inclinarse el mismo número de grados el mismo incremento de ganancia será válido para ambos modos, permitiendo así que el dron avance en ambos sentidos en vuelo equilibrado.

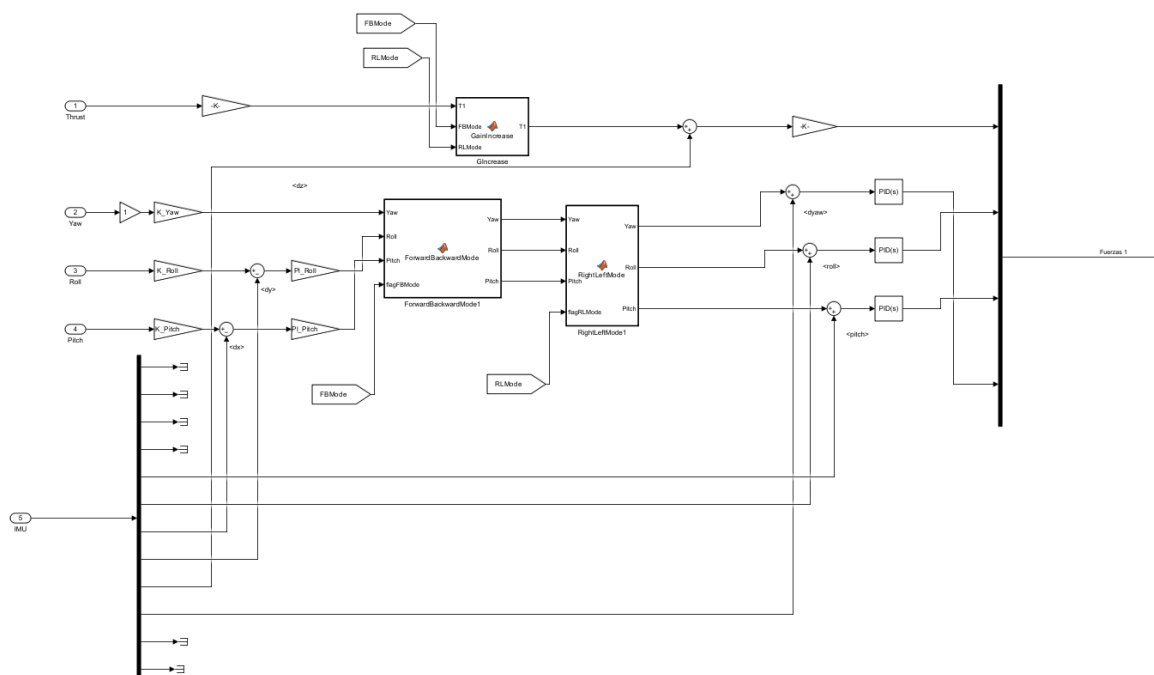


Figura 56. Bloque PID del control manual

### 6.2.2.1.2 Hover con retroalimentación de altura

Otro automatismo que se ha implementado en este modelo es el de hover con retroalimentación de altura. Se ha modificado completamente el bloque PID heredado de anteriores proyectos a:

1. Control del “Thrust” con retroalimentación del modelo de la coordenada vertical  $Z\{0\}$  del dron.
2. Control del “Yaw” o guiñada con retroalimentación del modelo de la velocidad angular del dron en el eje  $Z\{1\}$ .
3. Control del “Roll” o alabeo con retroalimentación del ángulo de alabeo del dron.
4. Control del “Pitch” o cabeceo con retroalimentación del del propio ángulo de cabeceo del dron

Esta funcionalidad permite al piloto desde la interfaz de Unity establecer un valor de altura en metros que se envía a Matlab a través de la variable HoverTgt, esta entra al controlador PID retroalimentado con la altura, mediante un tipo de control proporcional modificado con el peso del vehículo y tras otro control proporcional, conseguimos el objetivo propuesto por este automatismo.

Respecto a las otras variables a controlar (yaw, roll, pitch), sería muy difícil ajustar las ganancias de este el controlador si estos valores pudieran ser variables, por lo que se consideran nulos cuando la función de Hover está activada, lo cual tiene mucho sentido ya que conseguimos tener el dron sin ningún tipo de velocidad angular,  $\vec{\Omega} = \vec{0}$ , lo que permite la estabilidad del vehículo.

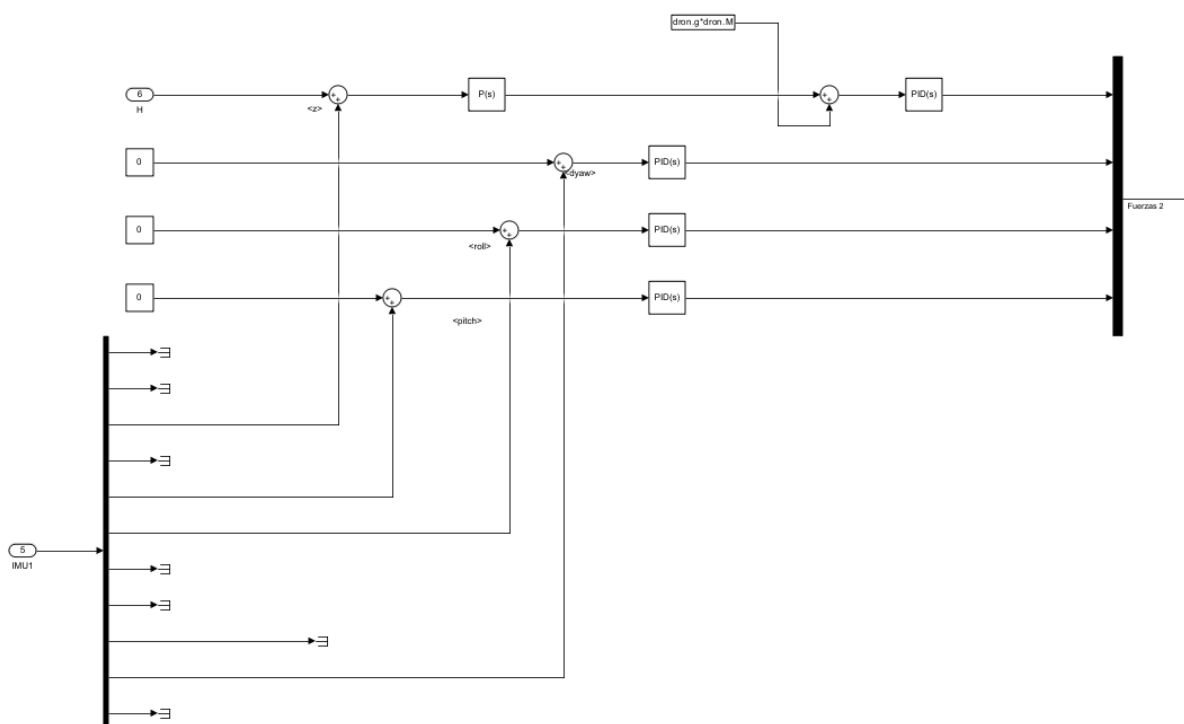


Figura 57. Bloque PID del modo hover con realimentación de la altura

### 6.2.2.1.3 Selección de modos

La selección de modos se consigue mediante la variable discreta ControlMode que sale del bloque Mode Select en función del valor de la variable "Function" proveniente de la interfaz gráfica Unity.

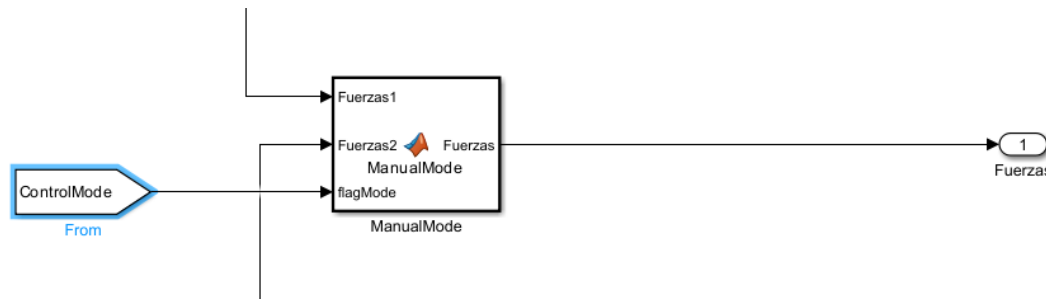


Figura 58. Bloque selección de modo de control

Cuando ControlMode es igual a 0, estaríamos en el modo manual con o sin funciones de avance frontal o lateral, mientras que cuando vale 1 tendríamos al dron bajo condiciones de "hover" a la altura que se haya impuesto en la interfaz gráfica Unity.

### 6.2.2.2 Determinación de las velocidades angulares de los rotores

En este bloque control está la parte del código dónde se aplica la teoría de la matriz de control y del modelo simplificado explicado en el apartado 3.2, por lo que se aplicará la Ecuación 43.

De donde obtendremos el vector velocidades angulares de dimensiones  $1 \times N$  donde  $N$  es el número de rotores del vehículo en cuestión. También en este bloque se implementa la condición de fallo rotor, así como la posibilidad de comandar el aterrizaje automático el cual se lleva a cabo en el bloque "motores" explicado en el apartado 6.2.2.3.

Analizando a alto nivel el bloque de la Figura 59 tenemos cuatro entradas y dos salidas.

Las entradas son:

1. Vector fuerzas o  $\vec{H} = (T \ \tau_x \ \tau_y \ \tau_z)$  que sale del bloque control y que ya se ha explicado su origen.
2. Variable height: Variable discreta que indica si el dron está en contacto directo con una superficie de reposo. Esta variable es útil en este bloque ya que, si se alcanza la condición suelo, height = 0, los momentos que recibe el control son prácticamente nulos ya que la fuerza del peso tras el contacto los anularía, por lo tanto, si esto ocurre la orden  $\vec{H}$  cambia a  $\vec{H} = (T \ 0 \ 0 \ 0)$ .
3. Variable nFallos: Variable discreta que indica el número de rotores fallidos.
4. Variable RotorFallido: Variable discreta que indica el primer rotor que se rompe en la condición fallo rotor.

Las salidas de este bloque son:

1.  $w_{mots}$ : Velocidad angular de los motores para cumplir la orden  $\vec{H}$ .



2. `flag_landing`: Variable discreta que induce a aterrizar el vehículo si pelagra su integridad debida a falta de control por la condición de fallo rotor.

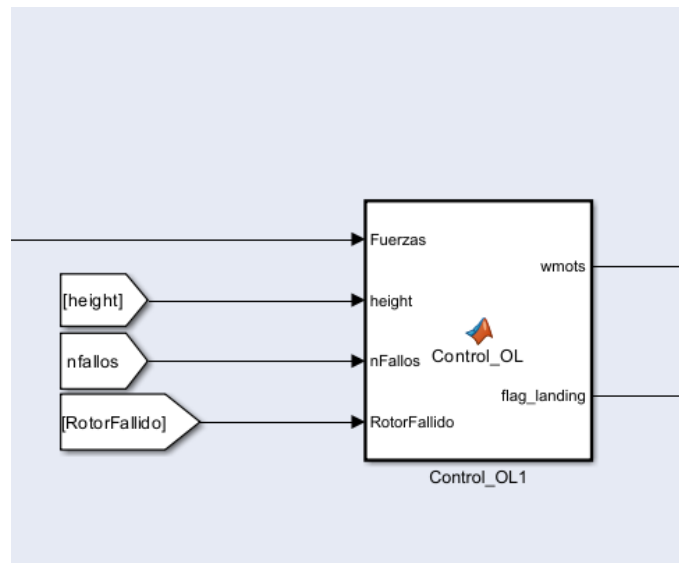


Figura 59. Bloque control de la velocidad angular de los rotores

Analizando el bloque un poco más a bajo nivel, el vector velocidades angulares se ha conseguido aplicando la inversa o pseudoinversa, según el vehículo, a la matriz de control y multiplicándola por el vector  $\vec{H}$ .

Mientras que la condición de fallo rotor aplica la teoría desarrollada en el apartado 5.4 a cada matriz de control dependiente del tipo de vehículo considerado, parando un rotor cualquiera del vehículo mediante el uso de la variable `RotorFallido` generada aleatoriamente en el motor gráfico Unity.

### 6.2.2.3 Saturación, “autolanding” y transitorio de los motores

De la Figura 60 podemos extraer que el bloque “motores” tiene 5 entradas y una única salida.

Las 5 entradas, 4 de ellas ya conocidas, al bloque motores son:

1. `wmots`
2. `nFallos`
3. `flag_landing`
4. `height`
5. `Landing`: Variable discreta que se envía desde la interfaz gráfica Unity para indicar al dron que aterrice.

La única salida de este bloque es `wmots`, que también es entrada. En este hecho radica la utilidad de este bloque, permite la modificación de `wmots` con el fin de modificar el comportamiento del dron por ejemplo haciendo que aterrice el vehículo o bien para dotar a la simulación de más realismo, por ejemplo, modelando la saturación y la velocidad mínima de los motores.

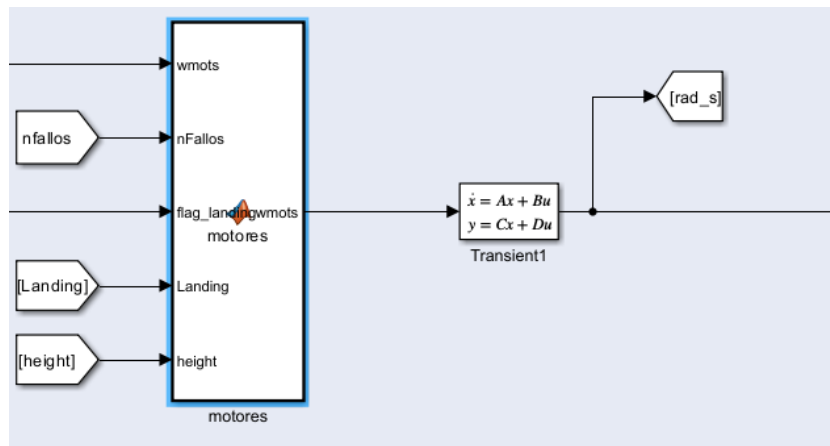


Figura 60. Bloque motores

Respecto al aterrizaje y el aterrizaje automático este bloque reducirá la velocidad angular de los motores, en función del vehículo, cuando alguna de las señales “Landing” (aterrizaje accionado por el piloto en la interfaz) o “flag\_landing” (aterrizaje automático por condición de fallo) estén activadas.

Respecto a la saturación y velocidad mínima de un motor, este bloque impedirá que la velocidad angular obtenida al resolver el bloque Control\_OL sea superior a su punto de saturación ni sea inferior al valor mínimo al que puede girar el rotor. Estos valores son característicos de cada motor y vienen definidos en el constructor de cada vehículo.

En cuanto al bloque “Transient” hace referencia al comportamiento transitorio de los motores, que consigue un comportamiento más realista de la simulación ya que la velocidad solicitada por el control no se alcanza de manera instantánea si no que existe un periodo de transición entre regímenes.

#### 6.2.2.4 Modelo Dinámico

El objetivo del modelo dinámico es determinar continuamente la posición y orientación del dron en función de las fuerzas y momentos generados por el modelo en cuestión.

Para lograr esto, el modelo debe proporcionar seis valores independientes que representen las tres posiciones lineales y las tres rotaciones en el espacio. El vector que contiene esta información se denomina vector de estado o simplemente estado.

En el modelo dinámico tendremos las siguientes variables de entrada:

1. Retroalimentación del vector de estado ampliado, aquí denominado como  $x$ , que contiene la siguiente información de la posición del dron:

$$\vec{p} = (X \ Y \ Z); \vec{n} = (\varphi \ \theta \ \psi); \vec{v} = (u \ v \ w)_1; \vec{\alpha} = (p \ q \ r)$$

2. AGL (Above ground level) indica la distancia vertical a la superficie sólida más cercana, en este bloque se empleará únicamente para calcular el incremento de empuje por efecto suelo.
3.  $w$ , velocidad angular de los rotores, equivalente a  $w_{mots}$ , pero tras pasar por el transitorio.

#### 4. Height – Utilizado para aplicar la condición suelo

Las salidas de este bloque son la solución de la simulación, es decir dónde y con que orientación se encuentra el dron. Tendremos dos salidas.

1. Vector de estado ampliado que retroalimenta el modelo dinámico con el fin que las condiciones finales de una iteración sean las condiciones iniciales de la siguiente y así sea posible la integración del movimiento. Este vector de estado se envía al módulo 3: Envío de datos dónde mediante UDP se envía a Unity para su representación gráfica.
2. Vector de estado ampliado que recoge el bloque IMU, este contiene una información ligeramente diferente, y se envía al bloque PID para controlar el vehículo de manera más precisa, la información que se envía en el segundo vector de estado es:

$$\vec{p} = (X \ Y \ Z); \vec{n} = (\varphi \ \theta \ \psi); \vec{V} = (U \ V \ W)_0; \vec{\dot{n}} = (\dot{\varphi} \ \dot{\theta} \ \dot{\psi})$$

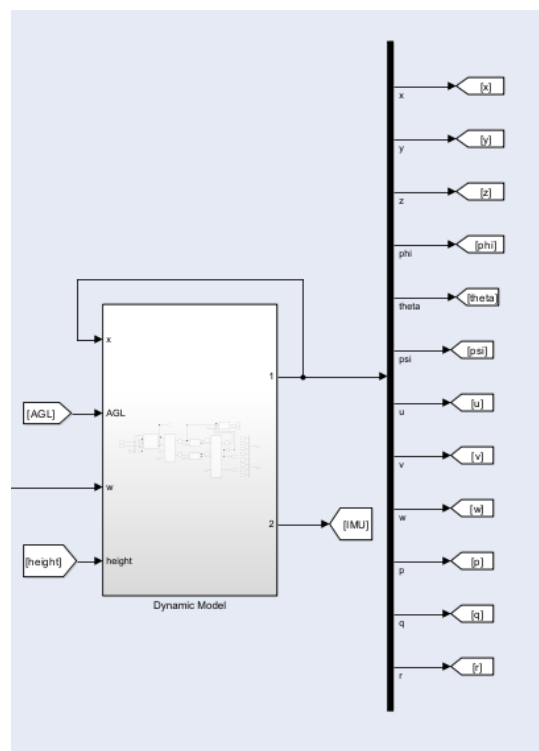


Figura 61. Bloque modelo dinámico

A más bajo nivel podemos ver los distintos bloques que componen el modelo dinámico, en este apartado no se va a analizar el código integrado en cada bloque ya que este es una representación literal de la teoría del modelo del sólido rígido explicado en el apartado 2.4,

pero si vamos a hacer un análisis de que entra y que sale de cada bloque.

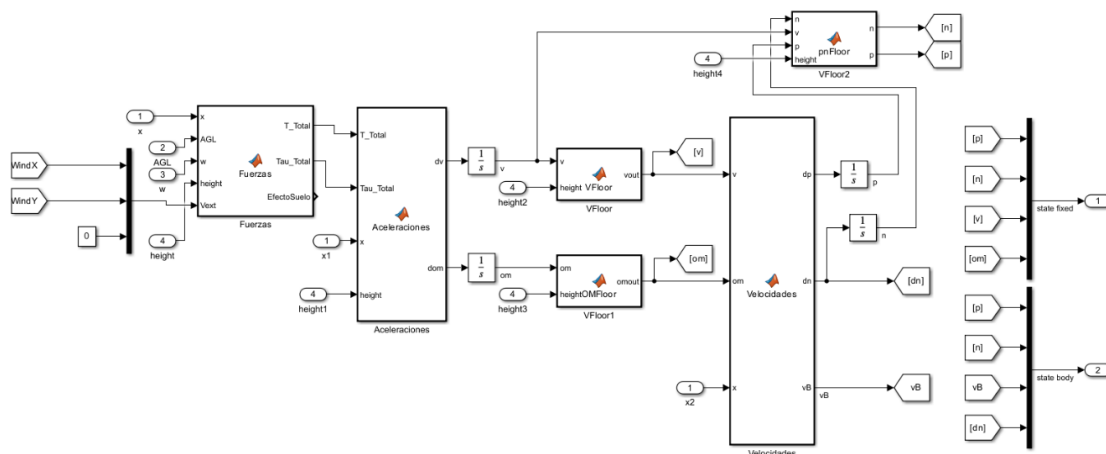


Figura 62. Interior del bloque modelo dinámico

#### 6.2.2.4.1.1 Fuerzas

Al bloque Fuerzas principalmente entran las velocidades angulares de los motores, el vector de estado retroalimentado para permitir la integración, parámetros para mejorar el realismo de la simulación: AGL, height, WindX y WindY.

Estos dos últimos son valores de la velocidad del viento que pueden ser modificados desde la interfaz gráfica de Unity y que lo que modifican es la velocidad relativa entre el vehículo y el viento, lo cual genera una fuerza de rozamiento con este mayor o menos según sean las direcciones de estas.

De este bloque salen dos vectores de tres elementos  $T\_Total$ , que hace referencia al conjunto de fuerzas totales en ejes fijos en las tres direcciones del espacio y  $Tau\_Total$ , que se refiere a los tres momentos en ejes fijo a los que se ve sometido el vehículo según su situación en la simulación, calculo que realiza tras analizar todos las fuerzas y momentos que se han considerado relevantes explicadas en el apartado 2.3.

#### 6.2.2.4.1.2 Aceleraciones

Una vez se obtienen las fuerzas y momentos a los que está sometido el dron en un instante determinado, podemos obtener las aceleraciones que ese sistema de fuerzas y momento inducen sobre el dron.

Esto precisamente es lo que hace este bloque de Simulink, cuya salida serán la aceleración del vehículo,  $\vec{v}$ , en las tres direcciones de los ejes fijos  $\{0\}$ , y las tres aceleraciones angulares  $\vec{\Omega}$  del vehículo en ejes móviles  $\{1\}$ .

Integrando estos valores tras el bloque de aceleraciones obtenemos los vectores:  $\vec{v}$  y  $\vec{\Omega}$ .

#### 6.2.2.4.1.3 Velocidades

Una vez obtenidas la velocidad en ejes fijos y la velocidad angular en ejes móviles, prácticamente el modelo dinámico queda resuelto, con una salvedad. El vector estado que nosotros definimos en la teoría no iba en función de  $\vec{\Omega}$ , si no que iba en función de  $\vec{\omega}$ .

Por este motivo es necesario aplicar el Jacobiano al vector  $\vec{d}$  para así tener esa misma información en los ejes intermedios, obteniéndose así  $\vec{n}$ .

Entonces llegados a este punto el modelo dinámico ha calculado  $\vec{n}$  y  $\vec{v} = \vec{p}$ . Integrando estas dos variables mediante un integrador discreto obtenemos la posición y la orientación del vehículo para cada una de las iteraciones del modelo, es decir el vector estado:

$$\overrightarrow{\text{estado}} = (\vec{p}_0 \ \vec{n}_1) = ((X \ Y \ Z)_0 \ (\varphi \ \theta \ \psi)_1)$$

#### 6.2.2.4.1.4 Condición suelo

A lo largo de la explicación del modelo se ha hecho mención en repetidas ocasiones a la variable height. Esta tiene su origen un bloque externo al flujo principal del modelo donde se evalúa si el vehículo está apoyado en alguna superficie sólida, si esto ocurre la variable height tomará el valor de cero, mientras que si el vehículo está en el aire esta variable valdrá 1.

Es un código muy sencillo que utiliza dos variables que proceden del motor gráfico Unity, estas dos variables son WOW (Weight On Wheels) será uno cuando la parte inferior del dron entre en contacto con una superficie y AGL (Above Ground Level).

También entra a este bloque la altura del vehículo alimentada desde la solución del modelo dinámico, variable a la que se ha tenido que acoplar un “delay” inicial para evitar la falta de información en la primera iteración de la simulación.

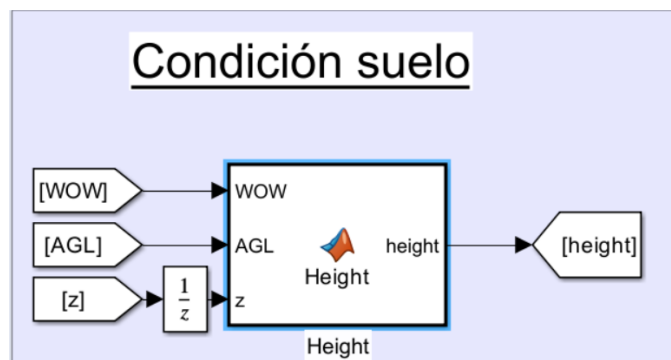


Figura 63. Bloque condición suelo

Por lo tanto, con estas tres variables se va a determinar si el vehículo está tocando el suelo o no, simplemente supondremos que el vehículo toca el suelo cuando pase al menos una de estas tres situaciones:

1. WoW sea igual a 1, lo que significa que hay contacto con otra superficie.
2. AGL o elevación del dron respecto a la superficie más cercana inferior a 10 cm.
3. Z inferior a cero, lo que significa que estamos por debajo del suelo.

Como ha podido observar el lector en la Figura 62 que presenta el modelo dinámico a bajo nivel, existen tres bloques de los que no se ha hablado y que todos contienen la palabra “Floor” en su nombre, estos bloques van asociados a la condición suelo y solo surtirán efecto cuando height sea igual a cero. Vamos a describir que hace cada uno de ellos:

1. VFloor: Anula las velocidades horizontales del dron antes de integrarlas y solo anula la vertical si el dron tiene componente vertical en sentido descendente.
2. OMFloor: Anula las tres componentes del vector:  $\vec{\Omega}$ : velocidades angulares de los ejes móviles antes de ser transformado a ejes intermedios e integrado.
3. pnFloor: Anula la segunda y tercera componente del vector  $\vec{n}$ , es decir el “roll” y el “pitch” del vehículo y establece como coordenada Z {0} la última coordenada Z {0} que el dron tenía antes de alcanzarse la condición suelo.

Existen otras consecuencias derivadas de la condición suelo que quedan reflejadas en el código, por ejemplo, ya se ha comentado en el apartado 6.2.2.2 que el vector  $\vec{H}$  se modifica al tocar el suelo, anulándose los momentos a los que se ve sometido pasando a ser:  $\vec{H} = (T \ 0 \ 0 \ 0)$ .

### 6.2.2.5 Selección de funcionalidades

En este bloque se configuran las variables discretas o “flags” que permiten a el modelo aplicar los diferentes automatismos. En la Figura 64 podemos observar que su única entrada será la variable “Function” que procede de la interfaz gráfica Unity desarrollada en [RAUL], donde en función del botón que apretemos enviará un valor u otro.

Este valor es interpretado por el bloque “Mode Select” dónde se configuran las diferentes variables discretas o “flags” que accionan los automatismos o modos de control.

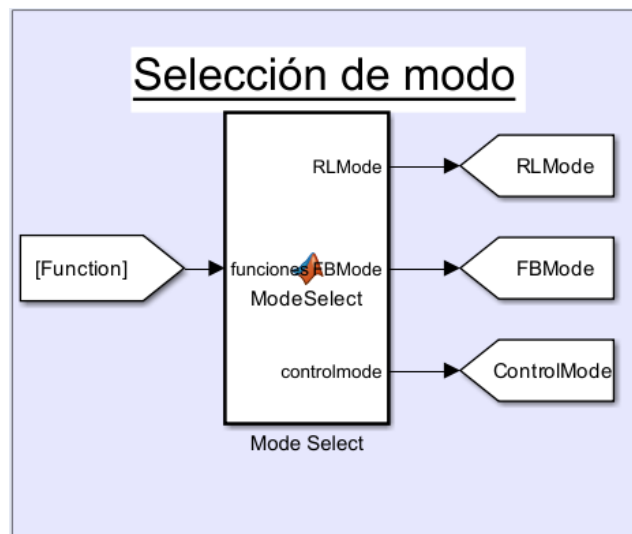


Figura 64. Bloque selección de modo

### 6.2.2.6 Descarga de batería

Respecto al bloque Simulink que simula la descarga de la batería solo es válido para los vehículos hexacoptero y octocoptero, ya que de los cuadricópteros no dispones de la curva de consumo en función de la velocidad angular del motor.

El funcionamiento de este bloque es sencillo, disponemos de una curva de consumo obtenida experimentalmente en el apartado 5.1 y disponemos también de la velocidad angular a la que giran los motores en nuestro entorno simulado.

En el primer bloque de este módulo Consumo\_Bateria se calcula mediante interpolación lineal sobre la curva experimental la corriente que está consumiendo cada uno de los motores en función de su velocidad angular.

Una vez calculada para cada uno de ellos se hace la suma del consumo de los N motores del dron. Después mediante un integrador discreto se va integrando este valor a medida que avanza la simulación.

Sabiendo la capacidad total de la batería, parámetro recogido en el constructor del vehículo y sabiendo que la batería estará completamente agotada cuando el consumo integrado del bloque Consumo\_Bateria designado con la letra k, sea igual a la capacidad total de la batería, podemos estimar el porcentaje de batería remanente.

Normalmente los drones disponen de dos baterías idénticas a bordo, este parámetro también se ha considerado en el constructor.

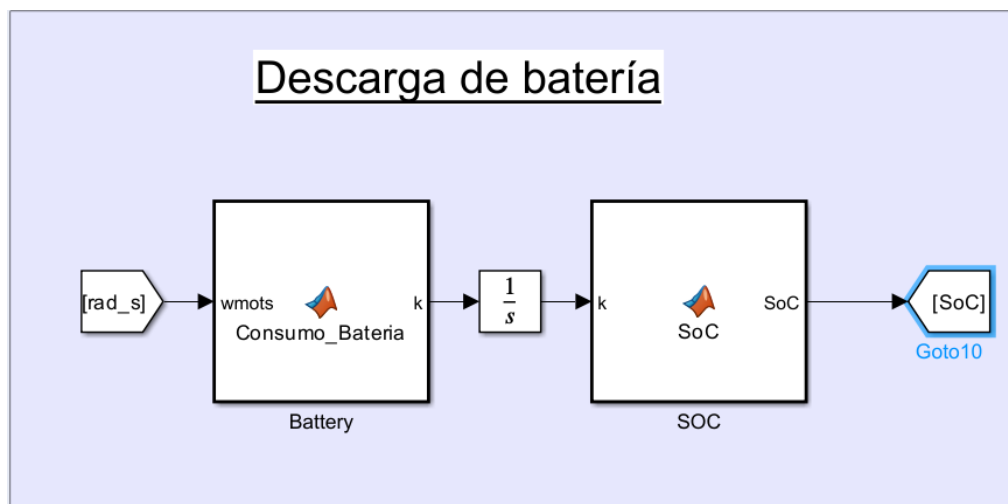


Figura 65. Bloque descarga de la batería

Por lo tanto, del bloque SoC, sale el valor porcentual del estado de la batería en el instante actual de la simulación. Este dato se envía a la interfaz gráfica Unity dónde se representará de una manera visual.

Se ha considerado oportuno que el estado inicial de la batería no sea siempre el 100% para hacer la simulación más realista, por lo que en colaboración con [13] se ha implementado en la formula del bloque SoC una constante inicializada desde la interfaz gráfica que inicializa al valor que le indiques el estado de carga de la batería.

Para más información de como se ha inicializado esta constante desde el motor gráfico Unity el lector puede consultar [13].

### 6.2.3 Módulo 3: Envío de datos

No se va a profundizar en cómo está desarrollado este módulo ya que no ha sido desarrollado en este proyecto, aunque si el lector está interesado puede consultar [13], para más información.

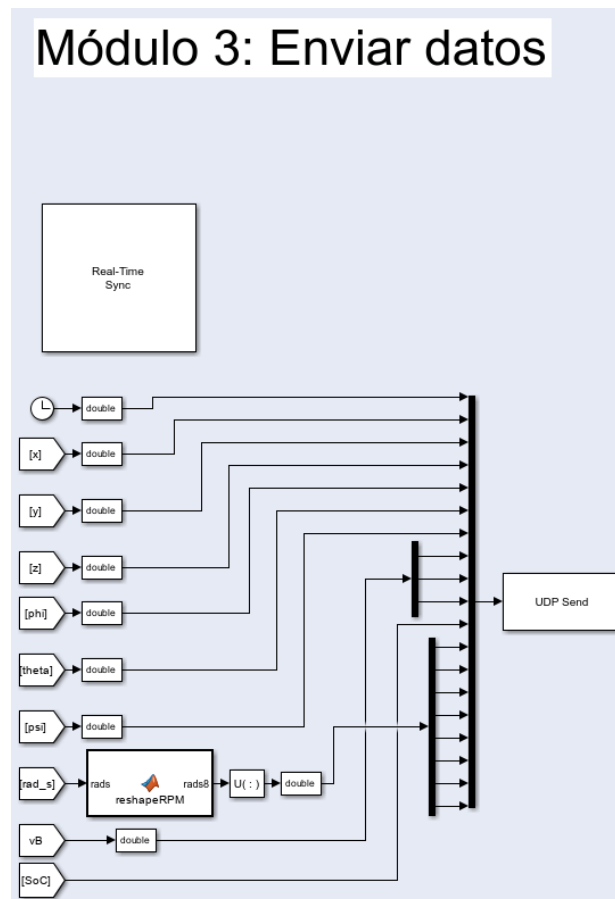


Figura 66. Modulo enviar datos

Lo que si se va a exponer son los parámetros de salida de este módulo de los cuales se nutre Simulink principalmente para determinar la posición del dron durante toda la simulación, pero también para crear una interfaz informativa del estado del vehículo.

Estos parámetros de salida son:

- Reloj – Tiempo que se lleva ejecutando la simulación
- x – Coordenada cartesiana X en el sistema de referencia fijo.
- y - Coordenada cartesiana Y en el sistema de referencia fijo.
- z - Coordenada cartesiana Z en el sistema de referencia fijo.
- phi - Angulo de “yaw” en el sistema de referencia móvil.
- theta – Angulo de “pitch” en el sistema de referencia móvil.
- psi - Angulo de “roll” en el sistema de referencia móvil.
- rad\_s – Velocidad angular de los n rotores del dron simulado, en rad/s.
- vB – Velocidad relativa del dron en ejes móviles
- SoC (State of charge) – Estado de carga de la batería



## Chapter 7. Interfaz gráfica

A continuación, se hace un pequeño comentario relacionado con el correcto uso de la interfaz gráfica elaborada en [13] la cual, se alimenta del modelo desarrollado en este documento. Dentro de la interfaz tenemos dos escenas claramente diferenciadas el menú de configuración y la interfaz gráfica.

### 7.1 Menú de configuración

Este menú de configuración accesible desde el botón menú visible en la esquina inferior izquierda de la interfaz gráfica permite elegir el multirroto que se desea volar, el estado inicial de carga de la batería, la velocidad horizontal del viento, así como el valor de FPS máximos que queremos para nuestra simulación.

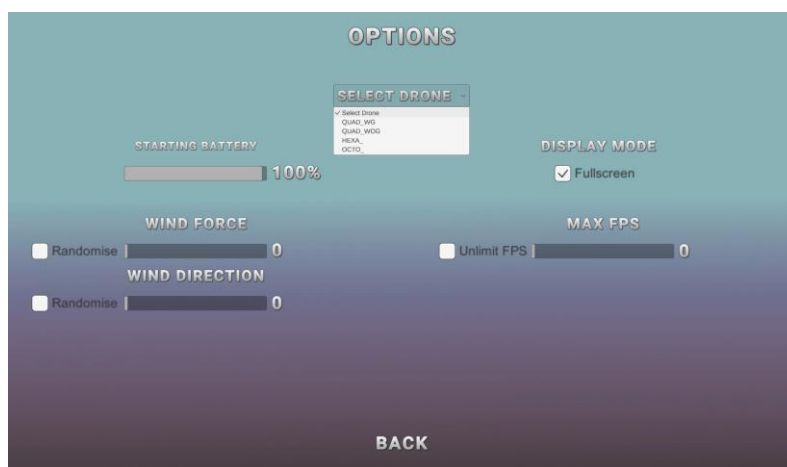


Figura 67. Menu de configuración

### 7.2 Interfaz gráfica

En las Figura 68 y Figura 69 podemos ver la interfaz gráfica cargada para el hexacóptero y el octocóptero, respectivamente. A continuación, se hace una descripción de lo que podemos encontrar en cada esquina de la interfaz.



Figura 68. Interfaz gráfica con vehículo hexacóptero



Figura 69. Interfaz gráfica con vehículo octocóptero

### 7.2.1 Menú de opciones

En la esquina inferior izquierda tenemos el menú de opciones donde cada uno de sus botones nos permitirá configurar de cierta manera la simulación.



Figura 70. Menú de opciones

#### 7.2.1.1 Botón "Menu"

Pulsando este botón accedemos al menú de configuración del apartado 7.1.

#### 7.2.1.2 Botón "Stop"

Con este botón paramos la simulación.

#### 7.2.1.3 Botón "Start"

Con este botón iniciamos la simulación.

#### 7.2.1.4 Botón "Camera"

Este botón permite cambiar de cámara.

#### 7.2.1.5 Botón "Functions"

En este botón encontraremos el acceso a todas las funciones automatizadas y condiciones de fallo desarrolladas en este trabajo. Al pulsar este botón se abre el desplegable que muestra la Figura 71.



Figura 71. Funciones del menu de opciones

En este desplegable aparecen las siguientes posibilidades:

1. Con el botón “Manual” se controla el vehículo mediante el mando.
2. Con el botón “Autolanding” se comanda la función aterrizaje automático.
3. Utilizando el cuadro de texto “Hover Target” se le puede indicar un valor constante de altura que al apretar el botón “Hover” el vehículo buscará alcanzar de forma autónoma.
4. Con el botón “Faulty Rotor +1” se incrementa en 1 el número de rotores rotos, con el botón “Reset Rotors” todos los rotores vuelven a su funcionamiento nominal.
5. Con el desplegable “Flight Assist Modes” se accionan los modos de vuelo automatizados:
  - a. “Forward” – El dron avanzara en vuelo estacionario hacia adelante.
  - b. “Backward” – El dron avanzara en vuelo estacionario hacia detrás.
  - c. “Right” – El dron avanzara lateralmente en vuelo estacionario hacia la derecha.
  - d. “Left” – El dron avanzara lateralmente en vuelo estacionario hacia la izquierda.

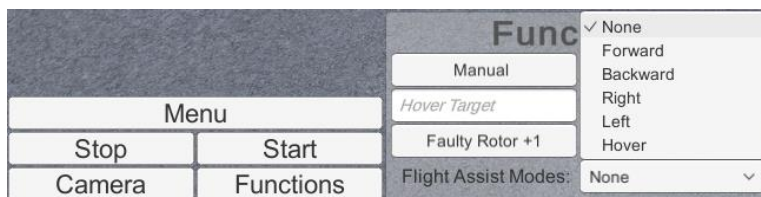


Figura 72. Modos de asistencia de vuelo

### 7.2.2 Cuadro informativo

En la esquina superior izquierda tenemos un cuadro informativo de datos de la simulación que nos permite conocer de un solo vistazo los siguientes datos de la simulación:

1. El tiempo transcurrido de simulación
2. Las coordenadas cartesianas en ejes fijos o tierra del vehículo en tiempo real.
3. Mensajes informativos del estado de la simulación, tales como “joystick desconectado” o “altura de vuelo máxima excedida”.



Figura 73. Cuadro informativo

### 7.2.3 Relojes y estado de la batería

En la esquina superior derecha tenemos lo que se han denominado relojes con la información de las revoluciones por minuto medias que llevan los motores, el “Above Ground Level” del vehículo y el módulo de la velocidad absoluta del vehículo, todo ello en tiempo real.

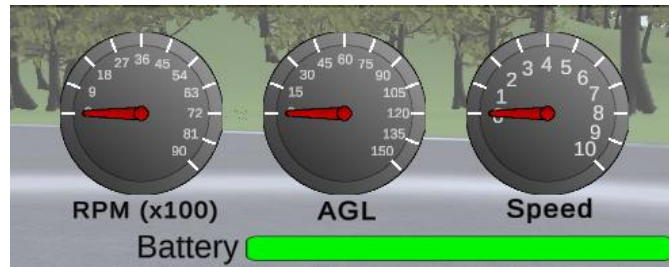


Figura 74. Relojes informativo y estado de la batería

Debajo de los relojes tenemos la barra del estado de carga de la batería, la cual aparecerá en verde cuando esté cerca del 100% transicionando a rojo cuando esté cerca de su punto de descarga.

### 7.2.4 Estado del mando del piloto

Por último, en la esquina inferior derecha tenemos la representación digital de los joysticks controlados por el piloto, estos también se moverán en tiempo real registrando en la interfaz gráfica toda orden suministrada por el mando real del piloto.

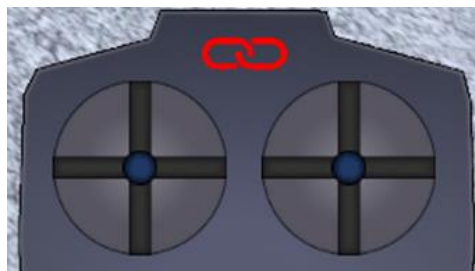


Figura 75. Estado del mando del piloto

## Chapter 8. Resultados

A continuación, se presentan las figuras para hacer los estudios comparativos, dónde siguiendo el criterio del proyecto, las gráficas azuladas corresponden al vehículo hexacóptero y las gráficas rojizas al vehículo octocóptero.

### 8.1 Estudio comparativo del modelo propulsivo

Tras aproximar los coeficientes dimensionales  $c_t$ , de ambos motores hemos obtenidos las siguientes aproximaciones a las expresiones del empuje en función de la velocidad angular de los motores:

$$T_{Hexacoptero} = 0.00001914 w^2$$

$$T_{Octocoptero} = 0.000048 w^2$$

Ecuación 61. Leyes de empuje asumidas en el Chapter 4

Como se ha comentado anteriormente estas expresiones son de carácter lineal respecto a la variable  $w^2$  de pendiente  $c_t$ .

En la Figura 76 se representan ambas aproximaciones, de donde podemos extraer las siguientes conclusiones:

1. El motor MT4006-13, hexacóptero, alcanza mayores velocidades de giro, en torno a 7000 rpm, que el MN4014-9, octocóptero, en torno a 5000 rpm.
2. Ante una misma velocidad de giro del rotor el motor MN4014-9 es capaz de proporcionar más empuje que el MT4006-13, esto se debe a que  $c_t$  en el octocóptero es mucho mayor que en el hexacóptero, lo que hace a la recta de empuje del motor MN4014-9 tener una pendiente mucho más elevada.

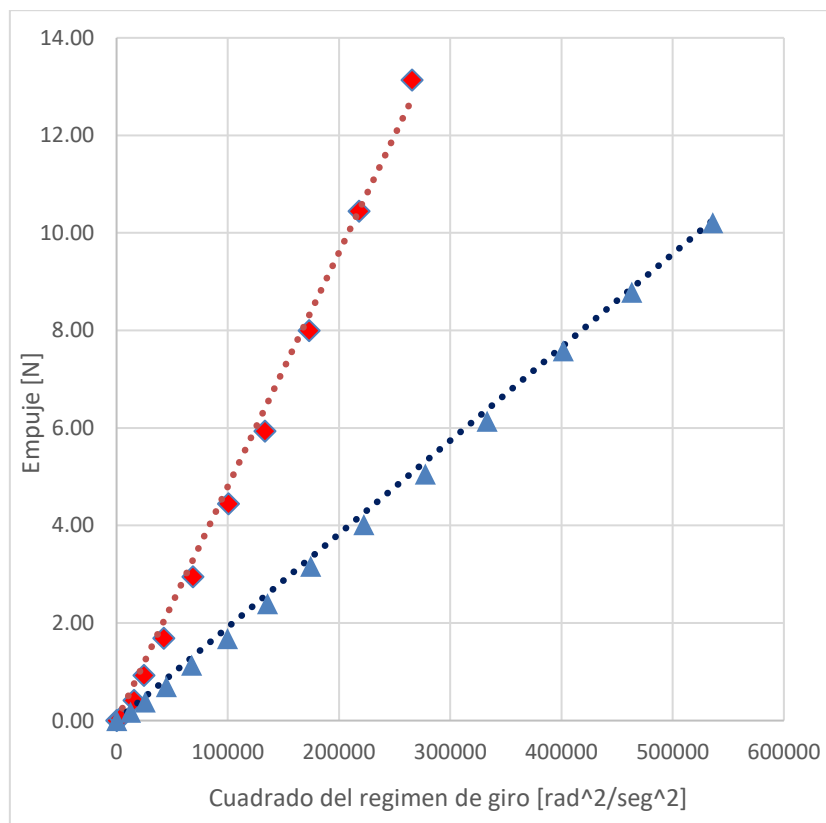


Figura 76. Representación de las leyes de empuje y los datos empíricos obtenidos en el ensayo

## 8.2 Estudio comparativo del modelo de batería

En este apartado se va a hacer un análisis comparativo entre las dos fuentes de energía estudiadas, la HRB batería del vehículo hexacóptero y la RC del octocóptero.

Las dos fuentes son baterías de polímero de litio (LiPo) recargables con el mismo número de celdas, cuatro en particular cada una de ellas con un voltaje nominal de 3.7 lo que hace un total de 14.8V y un voltaje máximo de 4.2V, total de la batería a carga máxima: 16.8V.

Respecto a sus diferencias la HRB tiene una capacidad de carga/descargar de 50C mientras que la RC la tiene de 30C, esto junto a la capacidad total influirá en lo rápido que se descarga la batería.

Respecto a las capacidades totales de ambas, la HRB dispone de 6000 mAh, mientras que la RC dispone de 8000 mAh, esto indica que ante el mismo consumo de corriente la batería RC del vehículo octocóptero proporcionaría una mayor autonomía al dron ya que su capacidad total es mayor.

A continuación, se hace un análisis comparativo de las gráficas más representativas del comportamiento de la batería en relación con el régimen de operación del motor, para todas las figuras se ha seguido el mismo criterio: Las curvas azules corresponden a la batería HRB del vehículo hexacóptero, mientras que las curvas naranjas describen la RC del vehículo octocóptero.

En la Figura 77 se analiza el consumo frente al régimen de giro del motor, en ella podemos ver que el motor del hexacóptero es capaz de girar a mayores velocidades consumiendo menos intensidad de corriente, por otro lado, el octocóptero a pesar de girar más lento consume mayor corriente, esto hace que consuma más rápido la batería reduciendo su autonomía. Un mayor régimen de giro no siempre implica mayor capacidad de empuje, el motor MN4014-9 del octocóptero es capaz de proporcionar mayor empuje girando a casi la mitad de revoluciones que el MT4006-13 del hexacóptero.

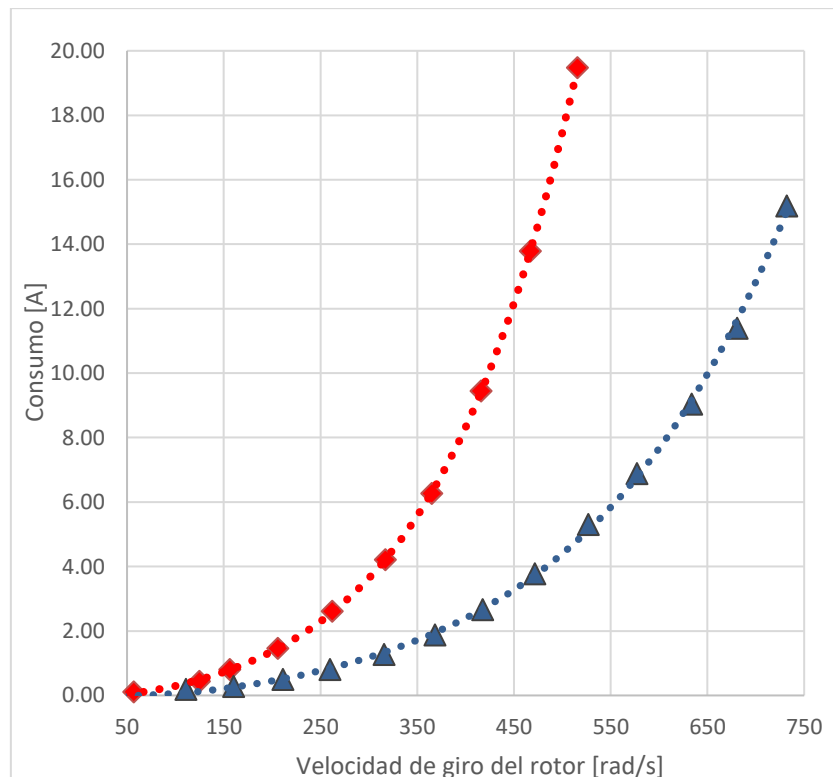


Figura 77. Representación de las curvas de consumo



Este hecho se puede ver perfectamente en la Figura 78 donde se compara el consumo de corriente del motor en función del empuje que este puede aportar. Lo primero que vemos es que el motor MT4006-13 a casi 7000 rpm aporta poco más de un kilo de empuje, mientras que el MN4014-9 rotando a casi 2000 rpm menos aporta aproximadamente 300 gramos de empuje más.

Respecto al consumo en función del empuje ambos motores prácticamente tienen la misma tendencia, siendo a partir de los 900 gramos de empuje cuando el motor MT4006-13, hexacóptero, consume más corriente que el MN4014-9.

Este hecho se hace notable cuando el MT4006-13 gira a velocidad máxima próxima a las 7000 rpm consumiendo más de 15 amperios, mientras que el MN4014-9 para conseguir el mismo empuje, aproximadamente 1 kg, debe girar a unas 4300 rpm consumiendo menos de 13 amperios.

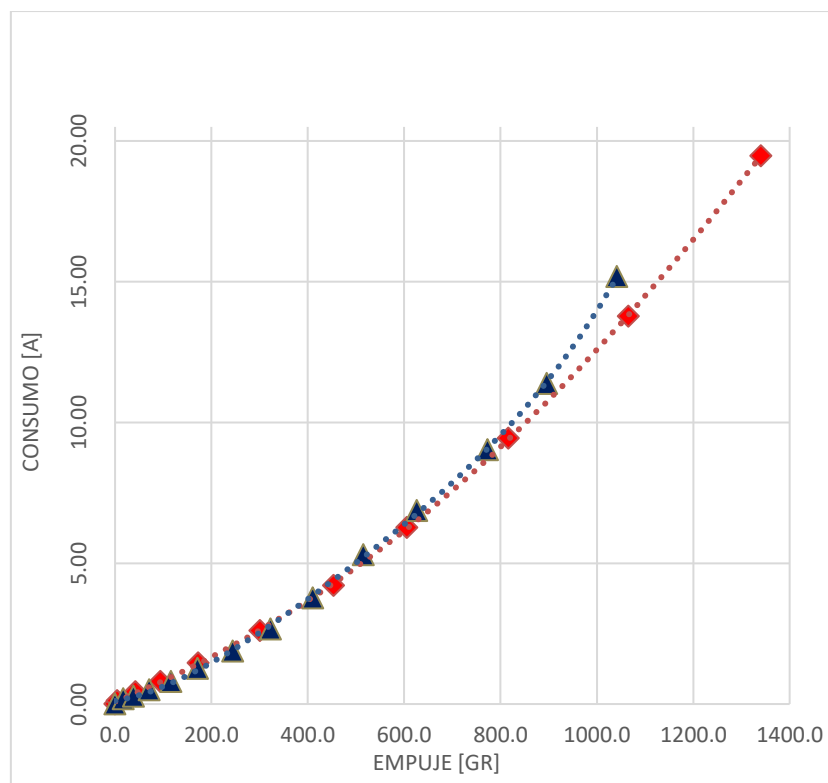


Figura 78. Relación consumo empuje

En relación a la potencia suministrada por la batería y consumida por el motor, de la Figura 79 se puede extraer la conclusión de que el MN4014-9 consume mucha más potencia que el MT4006-13 en todo su rango de régimen de giro. Esto se debe a que el motor MN4014-9 solicita a la batería, por lo general, mayores niveles de corriente para girar aun a menos revoluciones que el MT4006-13, proporcionando también por lo general mayor empuje, tal y como se ha comentado anteriormente.

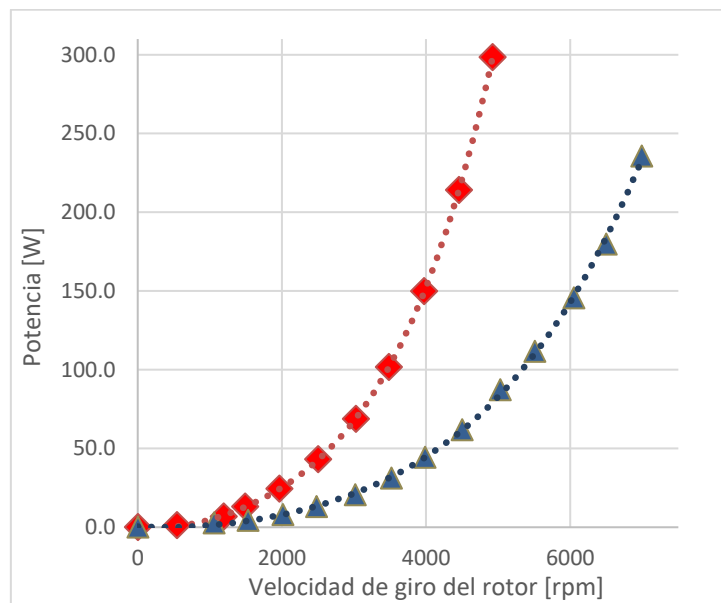


Figura 79. Potencia consumida en función del régimen de giro

Sabiendo los niveles de potencia que consume el motor y el empuje que este suministra en función del régimen de giro podemos valorar comparativamente la eficiencia empuje/potencia de ambos motores, es decir, cuantos gramos es capaz de suministrar el motor por unidad de potencia.

De la Figura 80 podemos extraer la conclusión de que el conjunto motor del hexacóptero es más eficiente en cuanto al empuje que puede suministrar en función de la potencia que consume prácticamente en todo el rango de giro estudiado, excepto a velocidades de giro muy elevadas donde prácticamente estas eficiencias se igualan llegando a ser incluso inferior la del hexacóptero en los puntos de saturación de los motores.

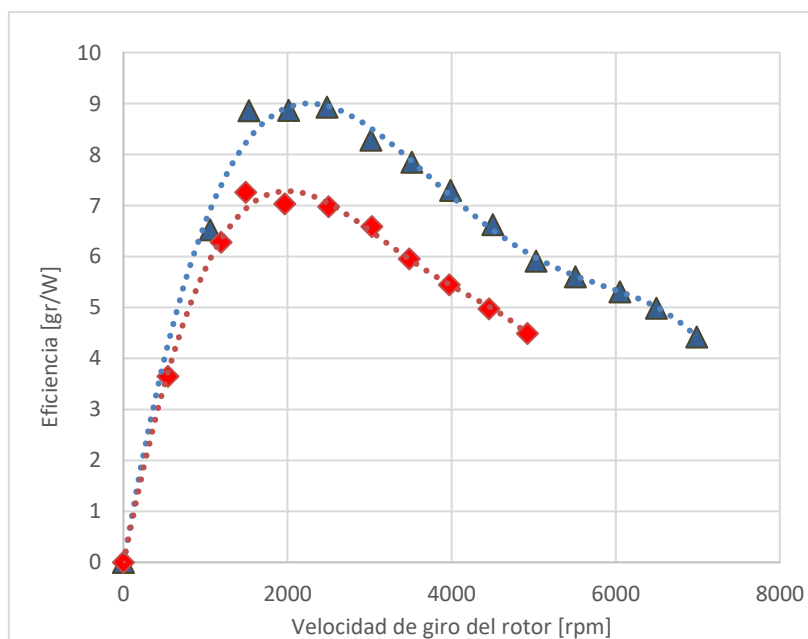


Figura 80. Eficiencia empuje/potencia en función del régimen de giro



### 8.3 Evidencias: Monitoreo de la simulación

Todas las figuras que se muestran a continuación han sido obtenidas en tiempo real mediante monitores “scope” en Simulink, han sido guardadas en el “workspace” como estructuras temporales, por lo que, el eje horizontal de cada una de las figuras de este apartado representa el tiempo que ha durado la grabación de la simulación.

#### 8.3.1 Consumo de batería

En este apartado se aporta la curva de descarga obtenida de monitorear el estado de carga de las distintas baterías modeladas de los dos vehículos multirrotores desarrollados en este proyecto.

Los datos se han recogido a través de un monitor Simulink dónde se ha ido modificando la velocidad de giro de los rotores intentando mantenerla constante duran unos segundos para resaltar la dependencia del consumo de la batería con este parámetro.

##### 8.3.1.1 Hexacóptero

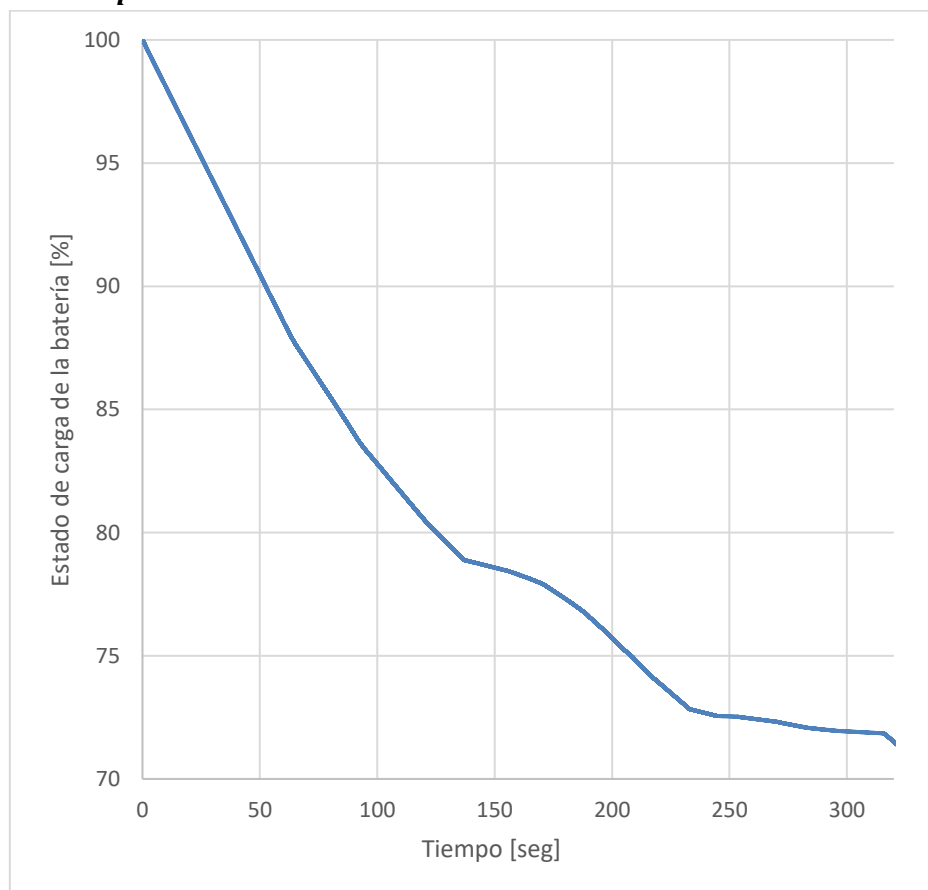


Figura 81. Descarga de la batería recogida durante la simulación – Vuelo hexacóptero

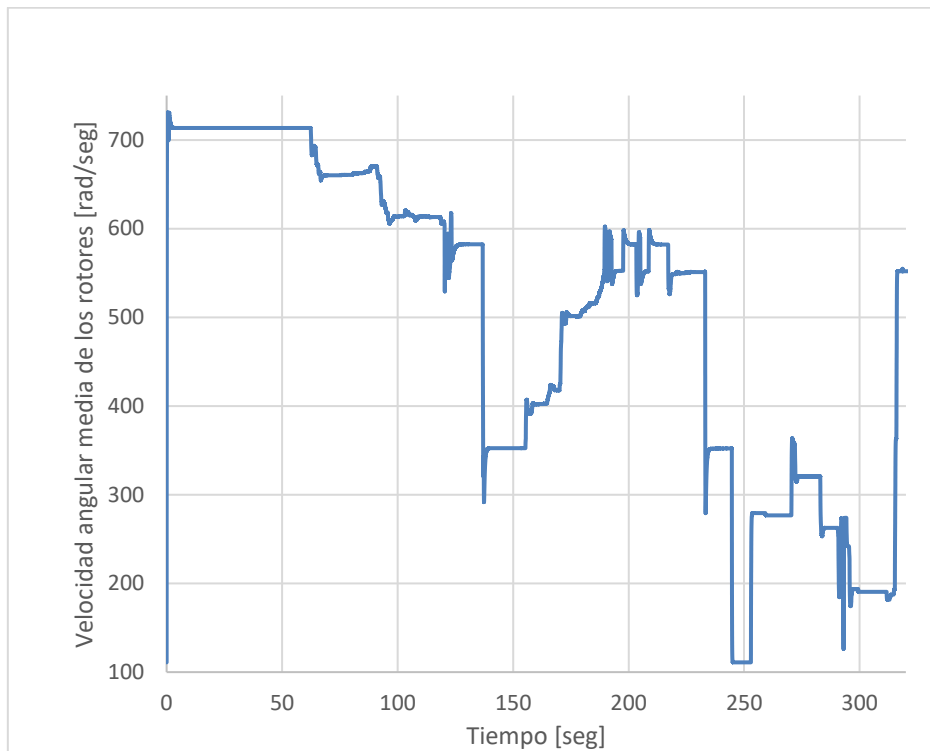


Figura 82. Velocidad angular media de los rotores durante la simulación – Vuelo hexacóptero

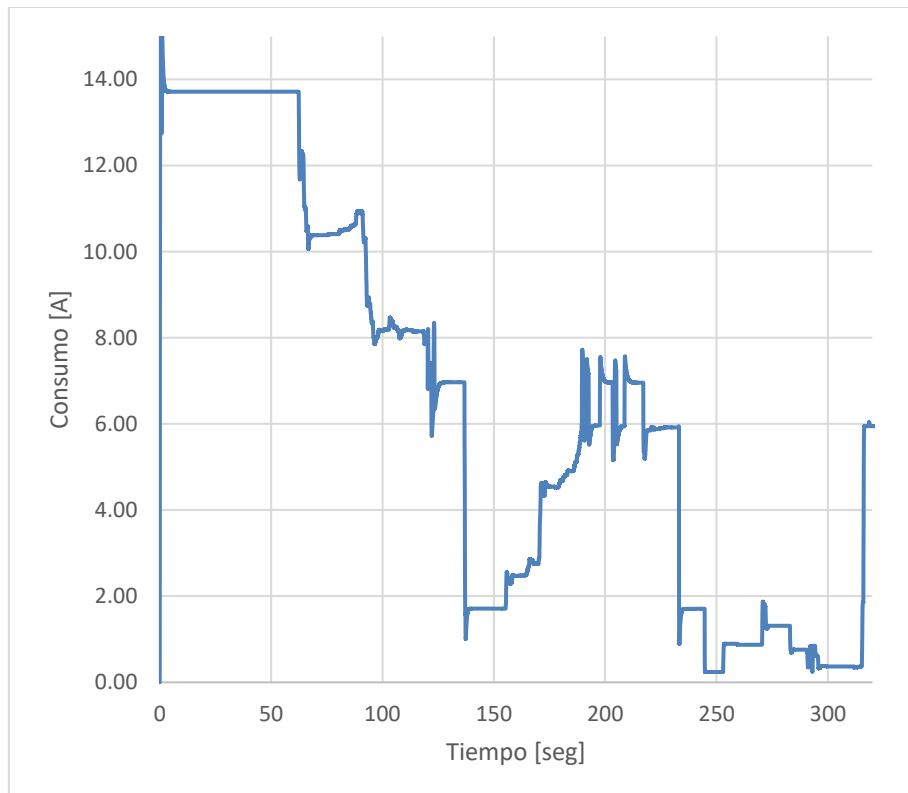


Figura 83. Corriente media consumida por los rotores durante la simulación – Vuelo hexacóptero

### 8.3.1.2 Octocóptero

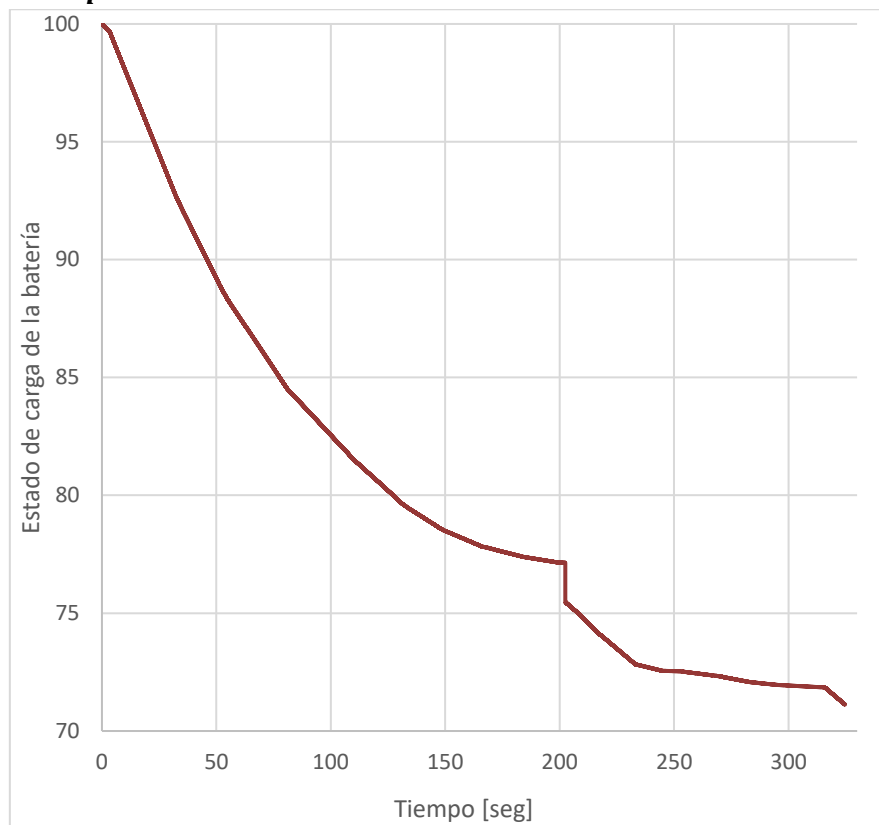


Figura 84. Descarga de la batería recogida durante la simulación – Vuelo octocóptero

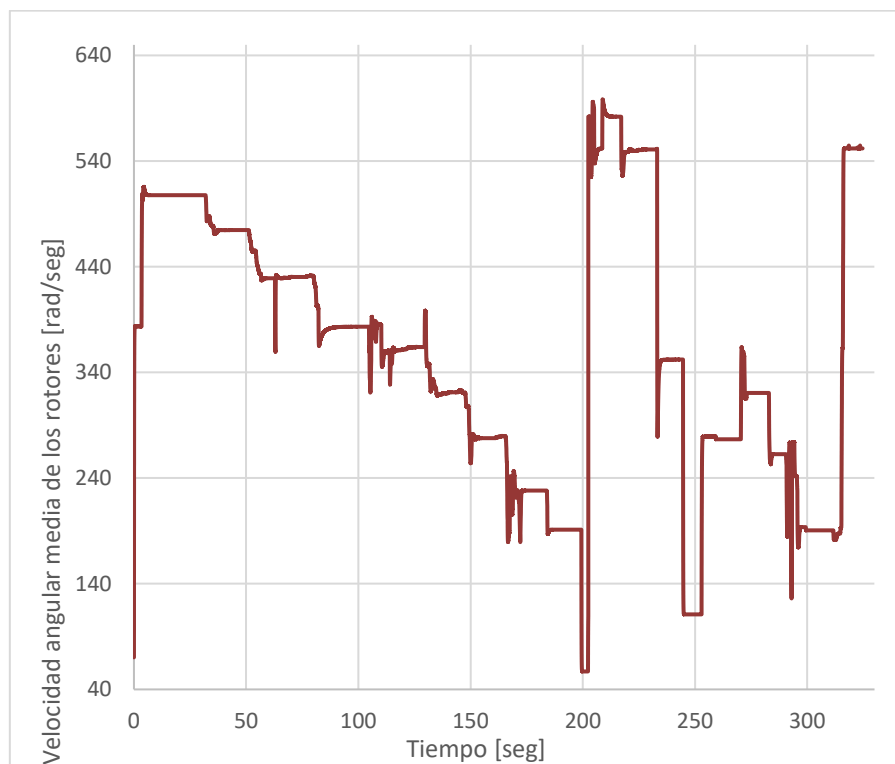


Figura 85. Velocidad angular media de los rotores durante la simulación – Vuelo octocóptero

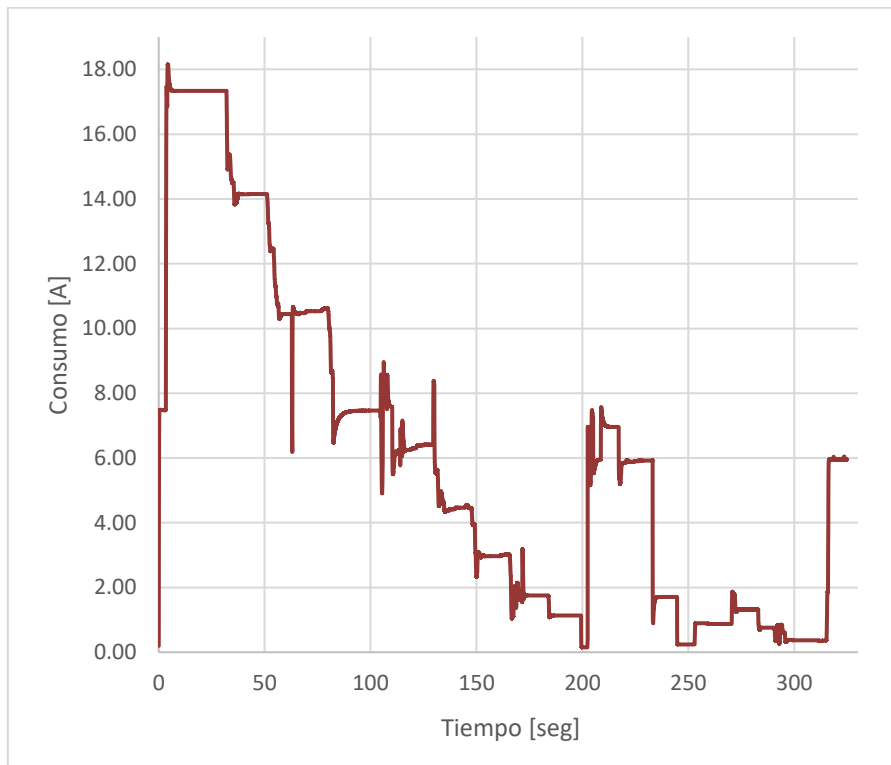


Figura 86. Corriente media consumida por los rotores durante la simulación – Vuelo octocóptero

Las figuras de este apartado y del apartado 8.3.1.1 inducen a las mismas conclusiones a las cuales son coherentes respecto a la teoría utilizada para modelizar la descarga de la batería y con los datos obtenidos experimentalmente del ensayo en el apartado 5.1.

Las conclusiones son las siguiente:

1. Un aumento del régimen de giro lleva asociado un aumento en el consumo del motor de carácter exponencial.
2. Un aumento en el consumo de corriente por parte del motor lleva asociada una mayor pendiente en la curva de descarga de la batería, representada en las Figura 81 y Figura 84.

### 8.3.2 Efecto suelo

A continuación, se aportan evidencias del incremento de empuje producido por el efecto suelo, para ello se ha creado en Simulink un monitor para ver la diferencia entre el  $T_{libre}$  cuando el dron vuela sin influencia del suelo y el  $T_{suelo}$ , cuando por el contrario el dron vuela cerca del suelo.

Haciendo uso de la variable  $ES = T_{con\ efecto\ suelo} - T_{libre\ de\ efecto\ suelo}$ , positiva cuando el efecto suelo está presente y cero cuando no.

Lo primero que queremos destacar es la diferencia entre la variable altitud o  $Z\{0\}$  y la variable altura o AGL, a menudo confundidas como sinónimos. En la Figura 87 y Figura 88 podemos ver que al principio ambas variables coinciden, esto es porque al inicio del vuelo la superficie que

estaba debajo del dron era el propio suelo ( $Z \{0\} = 0$ ). Alrededor del segundo 125, el vehículo vuela sobre una plataforma de unos 11,5 metros, lo que hace que el parámetro altitud difiera ahora si del parámetro altura.



Figura 87. Valor de altitud recogido durante la simulación.

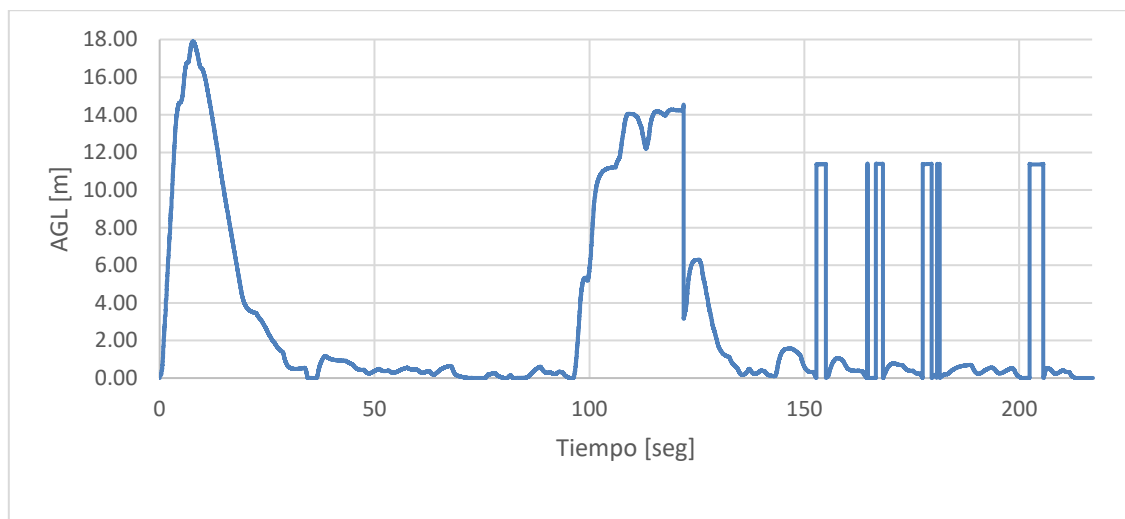


Figura 88. Valor de AGL recogido durante la simulación

Basando nuestro incremento de empuje en función del parámetro AGL, en vez de en Z, se ha conseguido hacer más realista la simulación, cuantificando este efecto en plataformas del entorno diferentes al propio suelo. Tras una simulación de vuelo a alturas bajas sobre el suelo y sobre la plataforma obtenemos los siguientes resultados del incremento de empuje mostrados en la Figura 90.

Los valores elevados que se aparecen en la figura para AGL, en torno a la altura de la superficie donde reposa el dron, se deben a que este parámetro de Unity falla cuando estás apoyado en una superficie, por eso la condición suelo se ajustó con un “collider” en Unity que activa el WoW. Para el efecto suelo esto no es un problema ya que en condiciones de reposo queremos que no actúe este efecto.

Como el efecto suelo empieza a ser significativo para este vehículo en valores inferiores a medio metro se ha hecho zoom a la Figura 88, para obtener la Figura 89.

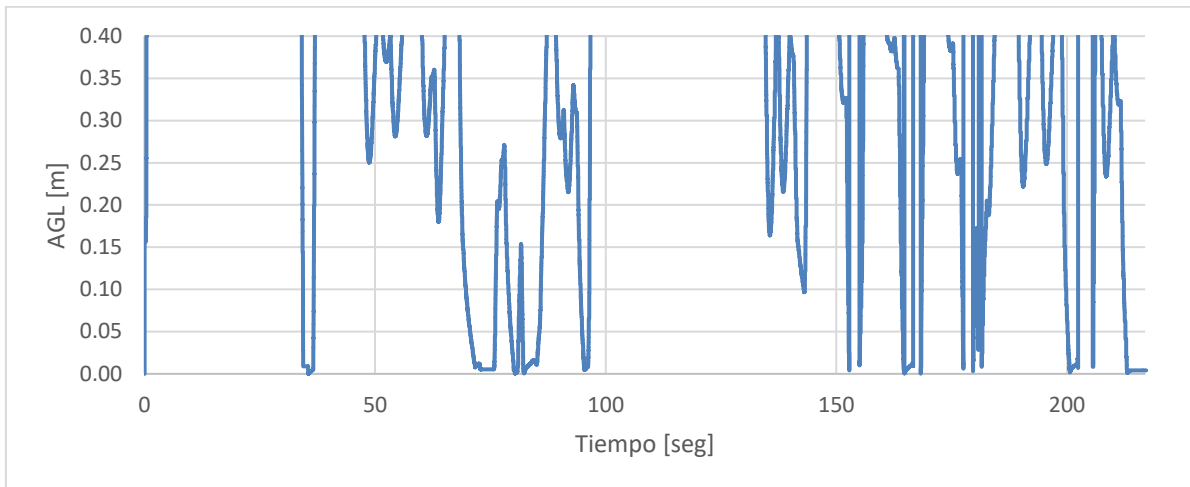


Figura 89. Valores de AGL inferiores a 40 cm durante la simulación

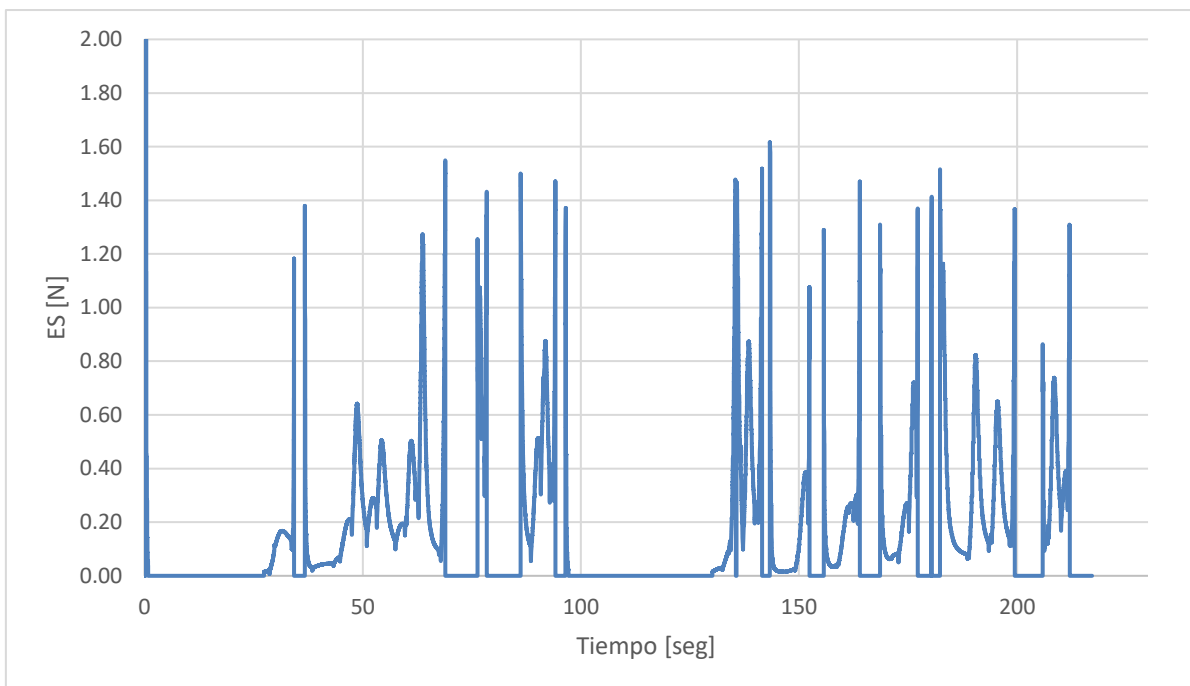


Figura 90. Incremento de empuje debido al efecto suelo

La conclusión a la Figura 90 es que el efecto suelo será más grande cuanto más cerca del suelo esté y será nulo cuando el vehículo esté en reposo o a más de 50cm de elevación respecto a la superficie de reposo.

### 8.3.3 Condición suelo

En este apartado se aportan evidencias de la evolución del vector de estado cuando se alcanza la condición suelo, es decir, cuando el vehículo se apoya en cualquier superficie sólida. Para ello se han simulado dos aterrizajes uno en el suelo y otro en una plataforma sólida situada a 11.5 metros de altura respecto al suelo.

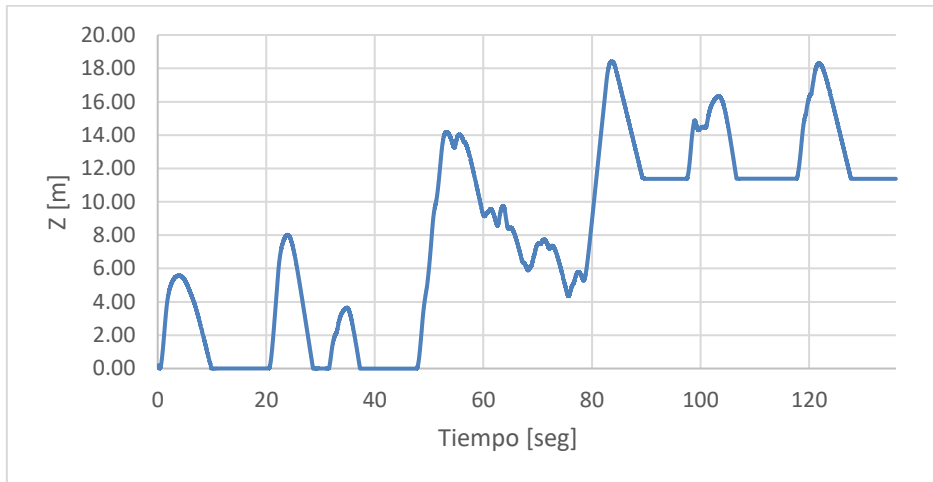


Figura 91. Variación de altitud durante la simulación

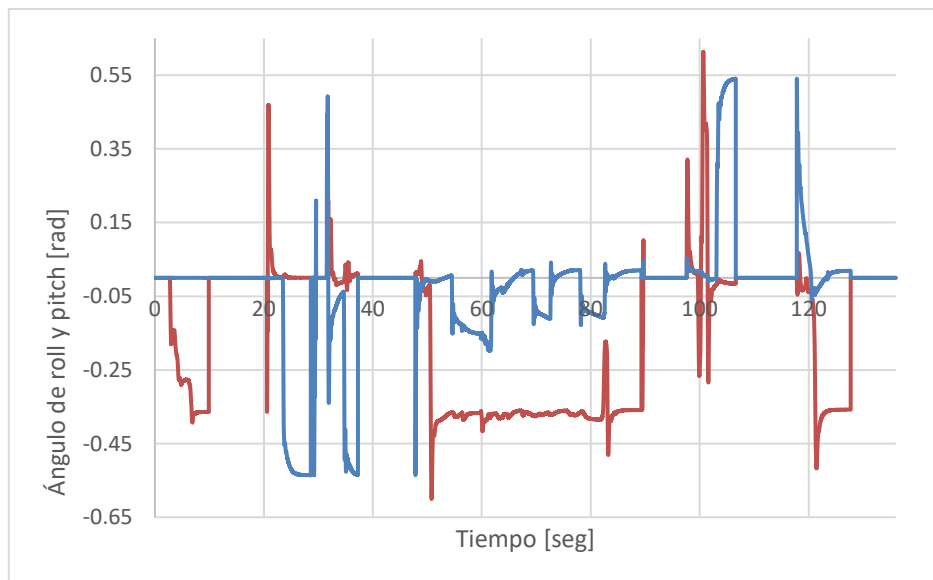


Figura 92. Variación de roll y pitch durante la simulación

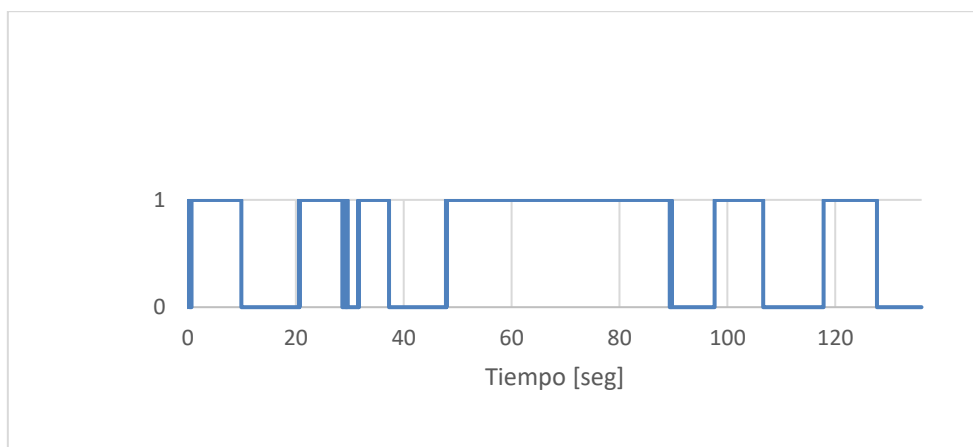


Figura 93. Evolución de la flag de la condicion suelo durante la simulación

La principal conclusión a las figuras de este apartado es que cuando se alcanza la condición suelo, valor de la “flag” = 0 en la Figura 93, la variable altitud se mantiene constante y los ángulos de cabeceo y alabeo se anulan independientemente del valor que llevasen cuando esta condición suelo se activó o de dónde aterrice el multirroto.

### 8.3.4 Fallo rotor

Para recoger las evidencias de la condición de fallo rotor se ha decidido monitorear la variable número de fallos y las velocidades angulares del vehículo.

Debido a que el cuadricóptero no admite esta condición de fallo, ya que el control del vehículo se hace imposible con solo tres rotores, se ha decidido registrar las evidencias con el hexacóptero y octocóptero.

Cabe resaltar que toda condición de fallo presentada en este apartado se ha producido mientras el vehículo estaba en situación “hover”, es decir, con un empuje total igual a su peso y con los ángulos que definen la orientación del vehículo configurados nulo. Esta situación del control cuando se produce la condición de fallo es muy importante, ya que el control siempre va a buscar cumplir la orden  $\vec{H}$ .

En este caso particular las velocidades de los motores siempre compensaran el momento de arrastre de cada motor para que el ángulo de guiñada sea cero.

#### 8.3.4.1 Hexacóptero

En las Figura 94 y Figura 95 de este apartado se puede observar cómo cambian las velocidades cuando se induce la condición de fallo motor.

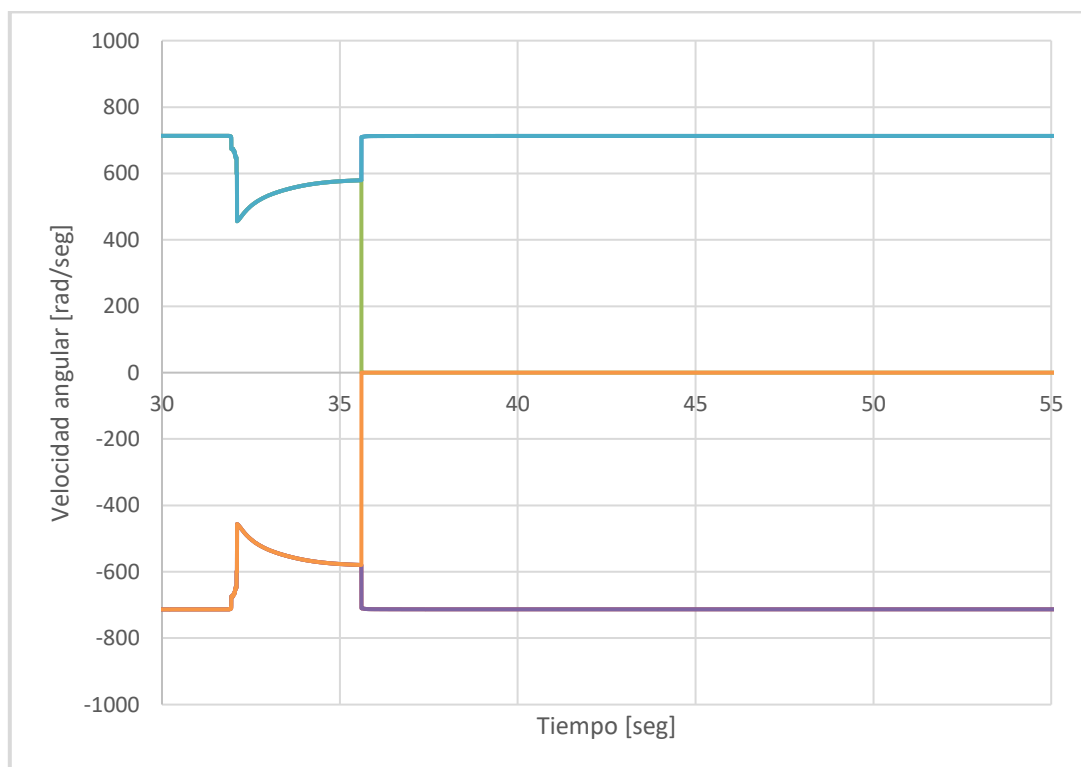


Figura 94. Velocidades angulares de los rotores del hexacóptero durante la simulación



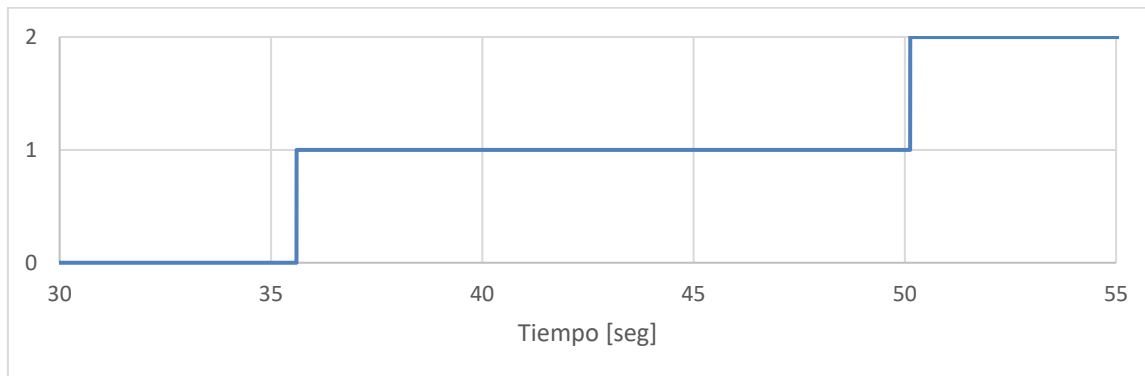


Figura 95. Número de rotores del hexacóptero comandados para parar durante la simulación

En el caso particular del hexacóptero, el control actúa consecuentemente respecto a la orden  $\vec{H}$  apagando el motor opuesto a la vez que el primer motor falla, esto ocurre para conseguir la condición de guiñada cero ya que es la única manera de compensarlo.

Por otro lado, y como consecuencia de que dos motores ya no están operativos, para seguir cumpliendo la orden en la que el empuje total contrarreste el peso de vehículo, el resto de motores tienen que acelerar, llegando a su punto de saturación.

Lamentablemente ni siquiera en este estado de potencia máxima 4 rotores son capaces de mantener en vuelo estacionario al multirroto hexacóptero por lo que el dron correría peligro y se decide ordenar el aterrizaje automático del vehículo simulado.

#### 8.3.4.2 Octocóptero

A continuación, se muestran el caso de la condición de fallo del vehículo octocóptero, muy similar al del apartado anterior, pero obteniéndose un comportamiento del control diferente.

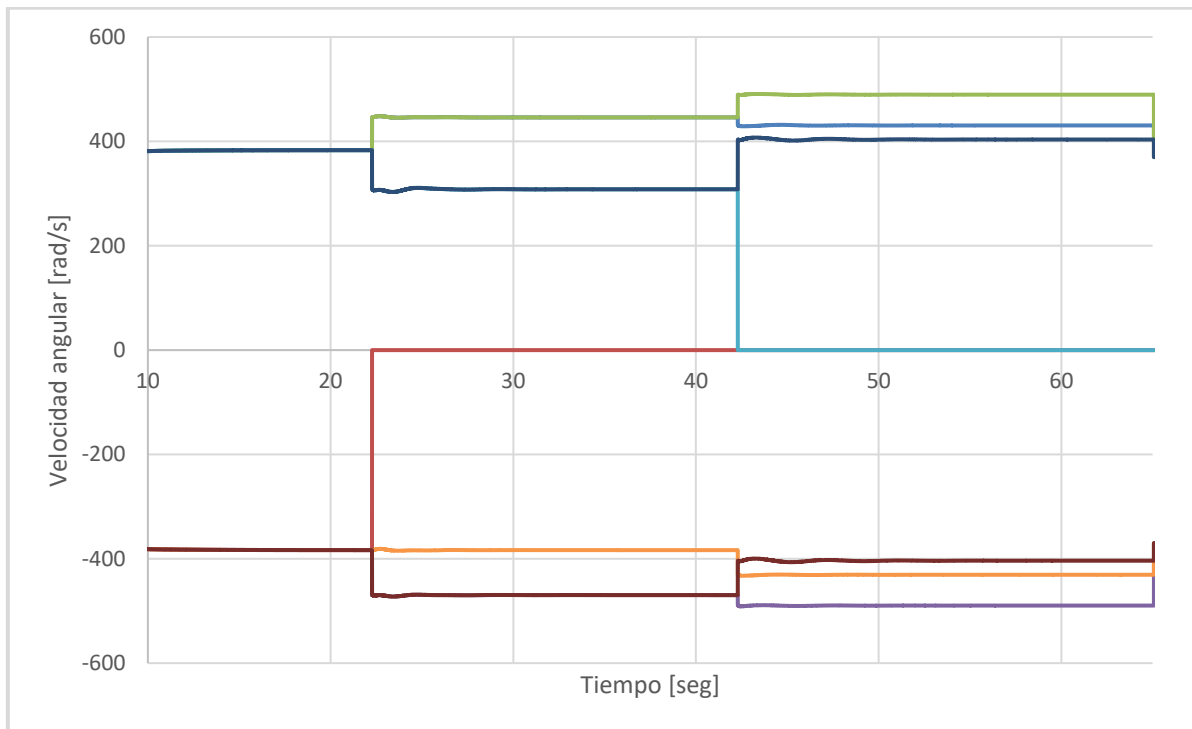


Figura 96. Velocidades angulares de los rotores del octocóptero durante la simulación

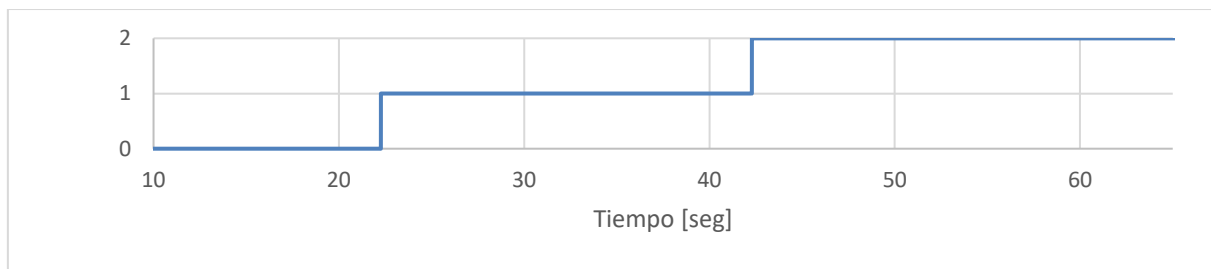


Figura 97. Número de rotores del octocóptero comandados parar durante la simulación

De la simulación se puede extraer la conclusión de cómo al introducir el primer fallo motor, solo sólo se apaga el motor 3, representado en rojo en la Figura 96. Esto difiere del comportamiento en el hexacóptero porque al tener este multirrotor dos rotores más existe una combinación de velocidades angulares de los otros siete que permita cumplir la orden  $\vec{H}$  del módulo de control, en este caso particular ángulo de guiñada nulo y empuje igual al peso del vehículo.

Cuando se acciona el segundo fallo rotor, un segundo motor representado en azul se apaga, este motor lleva velocidad de rotación opuesta al primer rotor fallido. Este comportamiento de fallo está programado y no es representativo de una condición de fallo real, pero se controla por un motivo, si el siguiente motor que falla tiene el mismo sentido de rotación el control de guiñada manteniendo la altura se haría muy difícil y el dron perdería estabilidad. También se evita que dos motores contiguos fallen ya que la capacidad de control del dron sería nula.

La planta propulsora del octocóptero funcionando con la condición de fallo rotor es capaz de sustentar el vehículo incluso con dos rotores averiados, lo que es una prueba concluyente de

que la planta propulsora de este vehículo es más potente que la del hexacóptero, conclusión que se deduce del estudio comparativo del apartado 8.1, no obstante, aunque es capaz de sustentarse en el aire el vehículo octocóptero ante esta condición de dos fallos rotores pierde completamente su capacidad de maniobra.

### 8.3.5 Automatismos

#### 8.3.5.1 Movimiento frontal

En este apartado se monitorea las coordenadas cartesianas  $X\{0\}$  y  $Z\{0\}$  del vehículo en tiempo real, así como, la variable discreta que sirve para accionar el automatismo movimiento frontal.

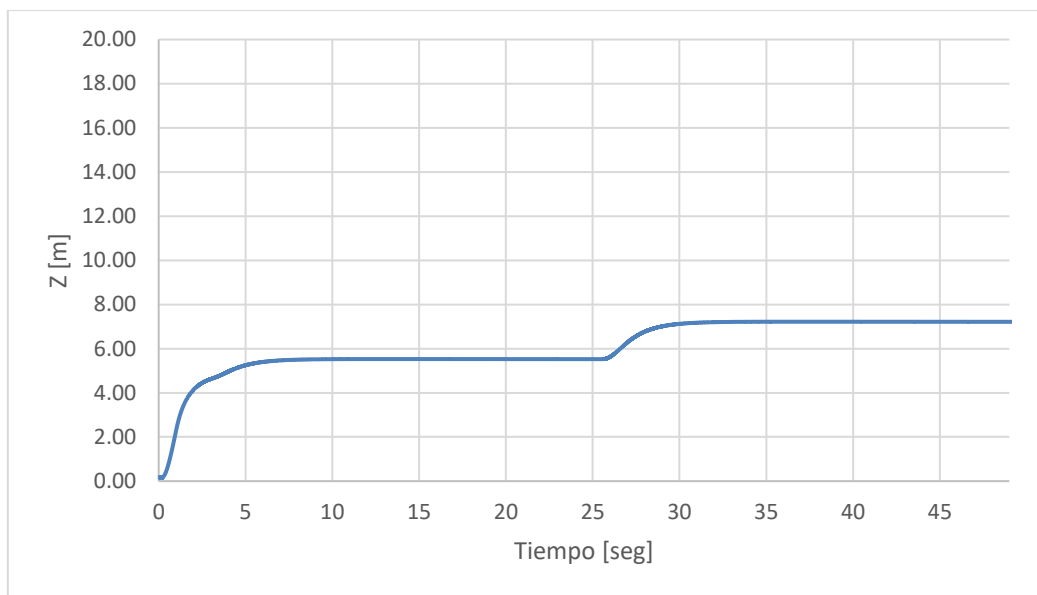


Figura 98. Evolución de la altitud de vuelo durante la simulación

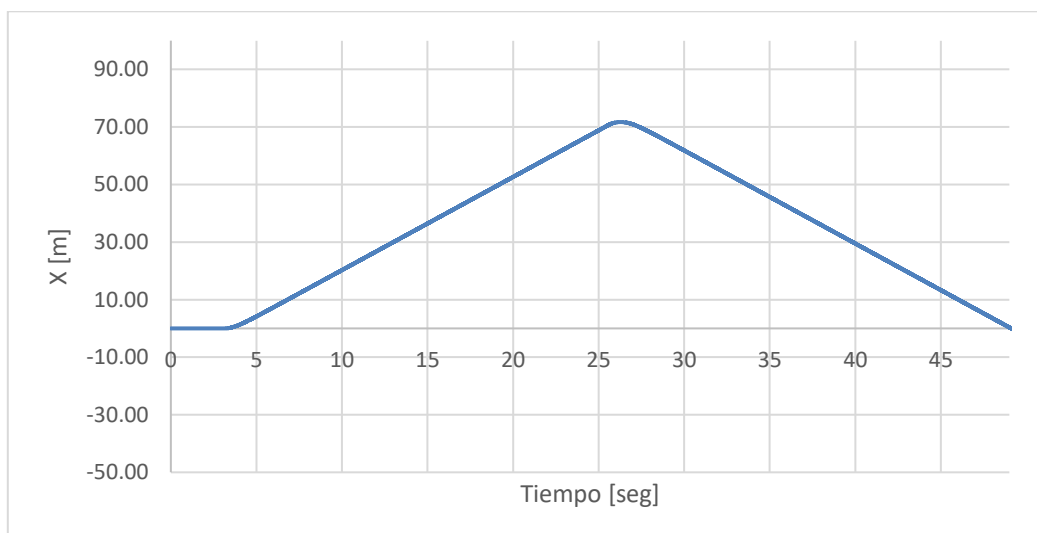


Figura 99. Evolución de la coordenada X durante la simulación



Figura 100. Evolución de la flag del modo avance frontal

Como puede observarse en las figuras de este apartado el valor uno positivo de la “flag” para el modo “forward-backward” hace que el vehículo avance en el sentido positivo de  $X\{0\}$  manteniendo una altura constante, mientras que si la “flag” adquiere el valor menos uno, el avance se produce en sentido contrario.

### 8.3.5.2 Movimiento lateral

En este apartado siguiendo la línea del apartado 8.3.5.1 se monitorea las coordenadas cartesianas  $Y\{0\}$  y  $Z\{0\}$  del vehículo en tiempo real, junto a la variable discreta que sirve para accionar el automatismo que desplaza lateralmente el vehículo por el entorno.

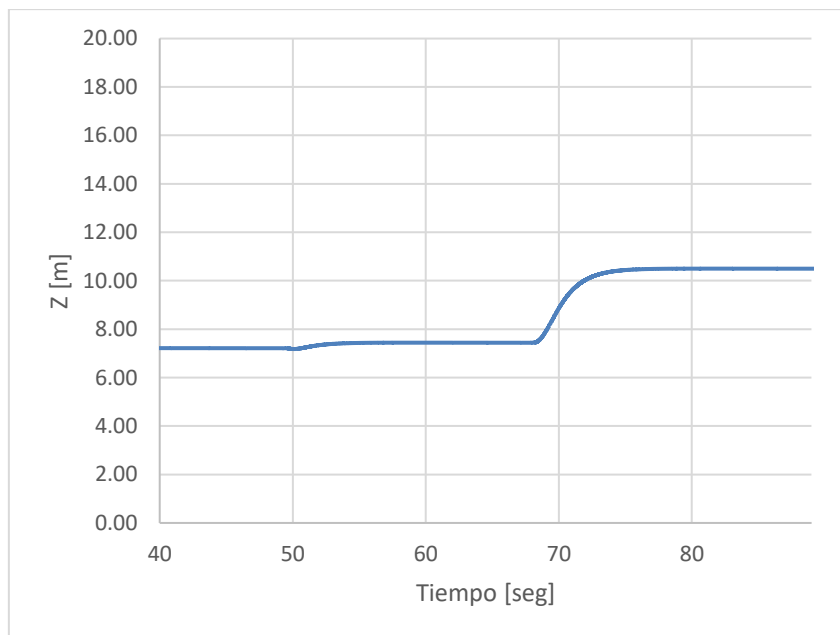


Figura 101. Evolución de la altitud de vuelo durante la simulación

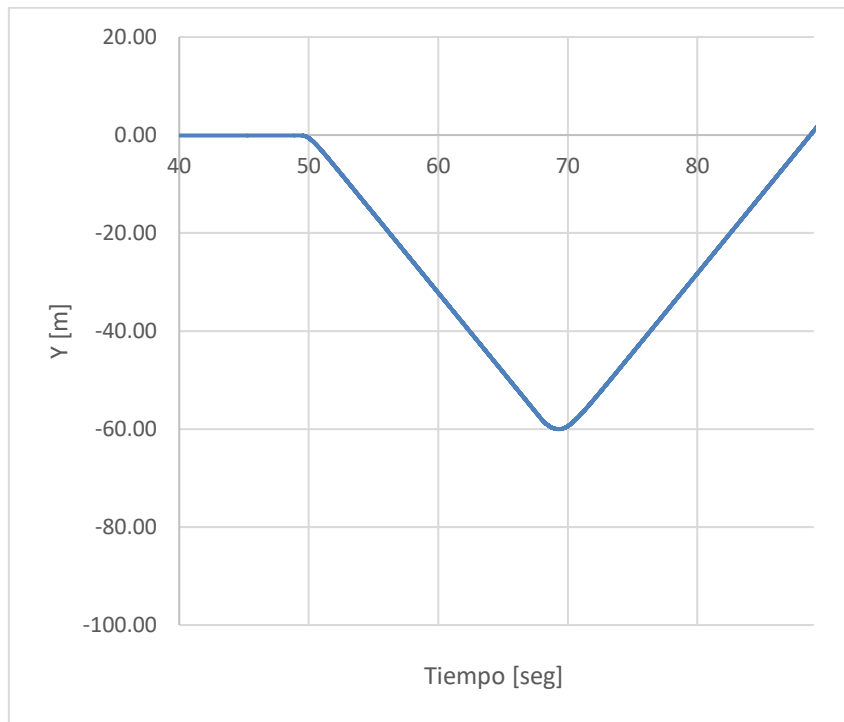


Figura 102. Evolución de la coordenada Y durante la simulación

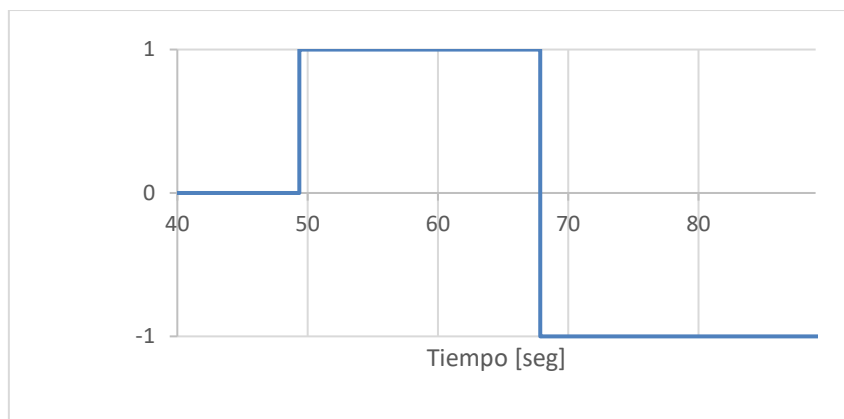


Figura 103. Evolución de la flag del modo avance frontal

De las figuras de este apartado extraemos que el valor más uno para la “flag” del modo “right-left” hace que el vehículo avance en el sentido negativo de  $Y\{0\}$  manteniendo una altura constante, mientras que si la “flag” adquiere el valor menos uno, el avance se produce en sentido positivo de  $Y\{0\}$ .

### 8.3.5.3 Hover basado en altura

En la Figura 104 viene representada en azul la altura real en tiempo real que tiene el vehículo en la simulación, mientras que en rojo aparecen los distintos valores de la variable “Hover Target” procedente de la interfaz gráfica.

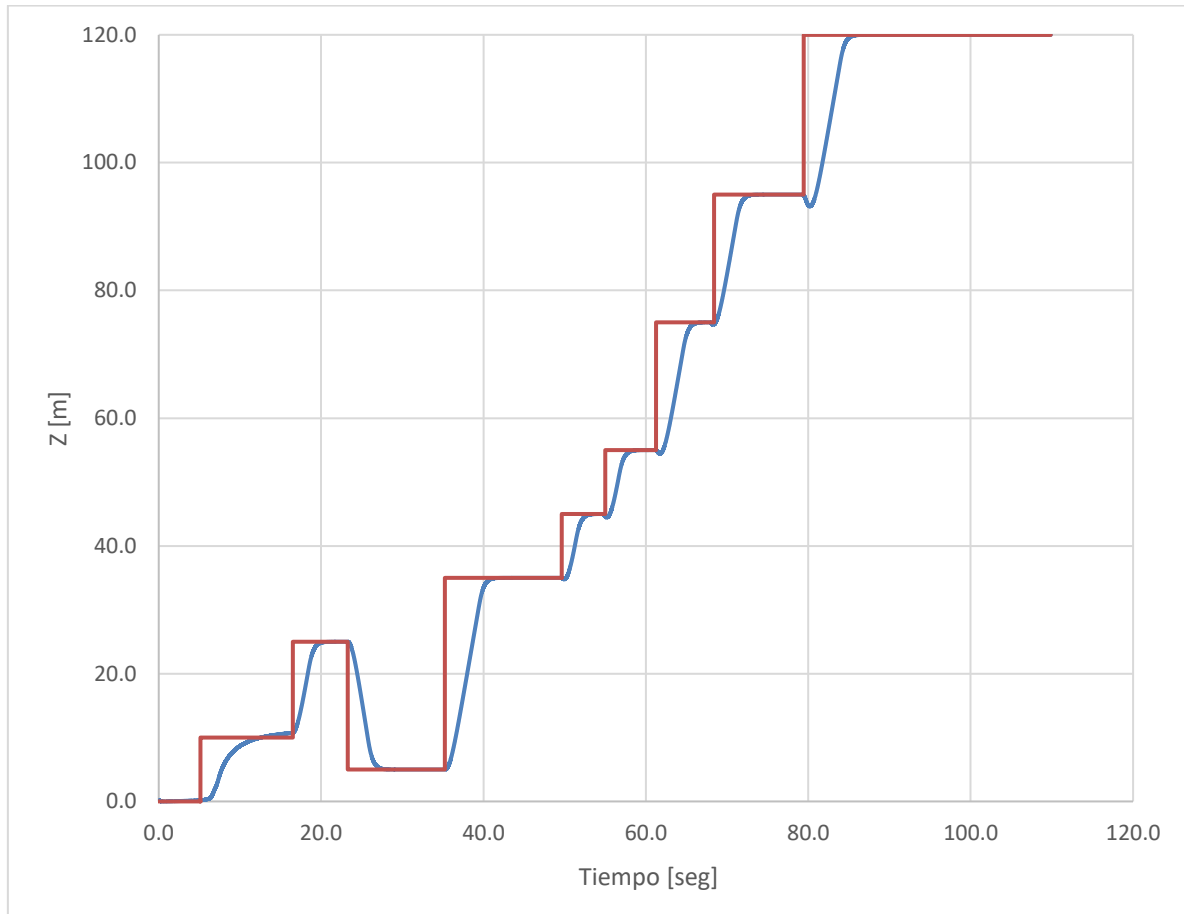


Figura 104. Evolución de la altitud y la variable HoverTarget durante la simulación

De los resultados de la Figura 104 podemos extraer la conclusión de que el control mediante PID realimentando con la altura de vuelo funciona perfectamente para el rango de vuelo previsto, altura máxima 120 metros, ya que la altura objetivo, “Hover target” es alcanzada de forma rápida, estable y permanente para todo el rango estudiado, en sentido ascendente y descendente.

## Chapter 9. Conclusiones

A continuación, se recogen los principales hitos que se han logrado durante el desarrollo de este proyecto fin de master.

1. Viabilidad de adaptar el modelo: A el modelo heredado por el autor se le ha añadido los vehículos multirrotor de 6 rotores: hexacóptero y el multirrotor de 8 rotores: octocóptero. Una vez adaptado el modelo a cualquier tipo de multirrotor de N rotores, se ha procedido a caracterizar empíricamente los multirrotos hexacóptero y octocóptero con el fin de incorporarlos a la simulación.
2. Resultados experimentales coherentes: Los datos empíricos obtenidos de la planta propulsora real de cada vehículo no difieren excesivamente de los teorizados por el modelo matemático de la simulación, por lo que en primera aproximación este es un buen punto de partida para simular la dinámica de cada uno de los vehículos. Respecto al modelo de la batería los parámetros obtenidos del ensayo también son representativos de las curvas características de una batería tipo LiPo.
3. Análisis comparativo de vehículos: Con la batería y la planta propulsora se ha hecho un análisis comparativo de ambos vehículos, donde la conclusión a nivel propulsivo es que el motor del vehículo octocóptero, MN4014-9, es capaz de proporcionar más empuje, girando a menos revoluciones pero solicitando a la batería una mayor cantidad de potencia en comparación con el motor del vehículo hexacóptero, MT4006-13, lo que a la larga, a velocidades de rotación medias hace al motor MN4014-9 menos eficiente.
4. Incorporación satisfactoria de mejoras al modelo y automatismos: En relación a la incorporación de estos modelos, las nuevas funcionalidades automatizadas y la condición de fallo rotor en el capítulo 8 se han monitorizado de la simulación, a través de “scopes” de Simulink, los parámetros fundamentales que reflejan que el algoritmo o la mejora en cuestión ha sido implantada satisfactoriamente.
5. Interfaz gráfica mejorada: Con la ayuda de la interfaz desarrollada en [13] usando el motor gráfico Unity se ha conseguido mejorar el realismo de la física del modelo, haciendo uso de parámetros aportados por este motor gráfico como el Above Ground Level (AGL) o el Weight on Wheels (WoW), así como, mejorar la experiencia audiovisual por parte del usuario.

## Chapter 10. Trabajos futuros

Respecto a las futuras líneas de desarrollo que se pueden aportar a este modelo, son numerosas las posibilidades ya que el simulador, aunque cada vez más sofisticado, está lejos de cumplir su objetivo de entrenador de pilotos profesionales, por lo que los siguientes esfuerzos de mejora del modelo irían encaminados a mejorar el realismo de la experiencia, por lo que se proponen los siguientes puntos de mejora.

1. Caracterización precisa de la aerodinámica del dron haciendo uso de software CFD se podría conseguir una representación más precisa de la fuerza de resistencia sobre el dron.
2. Desarrollo de un autopiloto estable del modelo, que sea capaz de proporcionar al multirrotor un control autónomo de una ruta previamente cargada.
3. Funcionalidad “return to home”, sería interesante dotar al modelo de la posibilidad de volver al punto de partida cuando ocurre una situación adversa, por ejemplo, simular pérdida de localización GPS del vehículo.
4. Aprovechando la fuerza del viento y sabiendo que modificando el bloque PID se puede modificar el comportamiento del vehículo de manera significativa, una funcionalidad atractiva para este tipo de vehículos es la de contrarrestar la fuerza del viento con la intención de mantener el vehículo estático en vuelo.
5. Simulación de los datos obtenidos por una IMU real, si bien este proyecto tiene un bloque Simulink al que denominamos IMU, este dista mucho de lo que es una unidad de medida inercial real, donde la mayoría de los datos registrados en ella requiere un filtrado y postproceso para ser utilizados.



## Bibliografía

- [1] I. Timmermans, «Entrenador basado en modelo parametrizado para pilotos de multirrotor,» ETSII, Universidad Polit écnica de Madrid, 2020. Tutor académico: Federico Martín de la Escalera.
- [2] A. F. López, «Adaptación de controlador de dron de ala fija a multirrotor e integración en simulador de vuelo,» Universidad Europea de Madrid, 2022. Tutor académico: Federico Martín de la Escalera.
- [3] G. Linares, «“Design and Integration a Software in the Loop for a Fixed Wing RPA,”,» Universidad Carlos III de Madrid, 2021. Tutor académico: Federico Martín de la Escalera.
- [4] «Wikipedia,» [En línea]. Available: [https://en.wikipedia.org/wiki/Ingenuity\\_\(helicopter\)](https://en.wikipedia.org/wiki/Ingenuity_(helicopter)) .
- [5] «CoolThings,» [En línea]. Available: <https://www.coolthings.com/dji-flight-simulator-drone-software/>.
- [6] Sriharsha Etigowni, Shamina Hossain-McKenzie, Maryam Kazerooni, Katherine Davis, and Saman Zonouz., «Crystal (ball): I Look at Physics and Predict Control Flow! Just-Ahead-Of-Time Controller Recovery,» Annual Computer Security Applications Conference (ACSAC '18), December, San Juan, PR, USA. ACM, New York, NY, USA, 2018.
- [7] NASA, «U.S Standard Atmosphere, 1976,» 1976.
- [8] S. Bouabdallah, «Design and control of quadrotors with application to autonomous flying,» ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE, 2007.
- [9] «Area dron,» [En línea]. Available: <https://www.areadron.com/que-tipos-de-drones-existen/#:~:text=Hexacopter%3A%20Tiene%20seis%20motores.,y%20tres%2C%20en%20sentido%20contrario..>
- [10] «HiModel,» [En línea]. Available: [https://www.himodel.com/electric/T-Motor\\_MT4006-13\\_740KV\\_Outrunner\\_Brushless\\_Motor\\_for\\_Multi-copter\\_3-4S.html](https://www.himodel.com/electric/T-Motor_MT4006-13_740KV_Outrunner_Brushless_Motor_for_Multi-copter_3-4S.html).
- [11] V. M. Martínez, F. M. de la Escalera, and S. G. Ruiz, «Curso para pilotos de RPAS: enfoque práctico, 5 ed.,» 2016.
- [12] «GetFpv,» [En línea]. Available: <https://www.getfpv.com/tiger-motor-mn4014-9-400kv.html>.
- [13] R. G. González, «Desarrollo de un simulador de vuelo de RPAS multirrotor e integración en un entorno gráfico,» Universidad Europea de Madrid, UEM, 2023. Tutor académico: Federico Martín de la Escalera.
- [14] K. C. Kim, «Analytical Calculations of Helicopter Torque Coefficient and Thrust (CQ) Coefficient (Or) Values for the Helicopter Performance (HELPE) Model, June 1999,» ARMY RESEARCH LABORATORY, 1999.
- [15] EASA, «Advance notice of proposed amendment 2015-10: Introduction of a regulatory framework for the operation of drones,» <https://www.easa.europa.eu/en/document-library>, 2015.
- [16] E. Commission, «Commision Delegated Regulation (EU) 2019/945,» <https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX>, 2019.

Título: Avances en el Desarrollo de un Simulador de RPAS Integrando Vehículo Hexacóptero y Octocóptero.

Autor: Alejandro Fernández Guerrero.



- [17] E. Commission, «Comission Delegated Regulation (EU) 2019/947,» <https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX>, 2019.
- [18] AESA, «Normativa europea de UAS/drones,» <https://www.seguridadaerea.gob.es/es/ambitos/drones/>, 2022.
- [19] K. Ogata, «Ingeniería de control moderna,» Pearson Educación, 2003.
- [20] FAA, US Department of Transportation, «Rotorcraft Flying Handbook,» FAA-H-8083-21, 2000.
- [21] G. Hattenberger, M. Bronz, and J.-P. Condomines, «Evaluation of drag coefficient for a quadrotor model, vol. 15,» International Journal of Micro Air Vehicles, 2023.
- [22] Vachtsevanos, K. P. Valavanis and G. J., «Handbook of Unmanned Aerial Vehicles,» Springer Dordrecht, 5 ed., 2015.
- [23] P. Sanchez-Cuevas, G. Heredia, and A. Ollero, «Characterization of the aerodynamic ground effect and its influence in multicopter control vol. 2017, p. 1823056, Aug 2017.,» International Journal of Aerospace Engineering, 2017.
- [24] Kumar, A. Nemati and M., «Modeling and control of a single axis tilting quadcopter,» American Control Conference, pp. 3077–3082, 2014., 2014.
- [25] A. P. Fernández, «Desarrollo de un modelo software in the loop (sitl) de una plataforma no tripulada,» ETSIAE, Universidad Politécnica de Madrid, 2022. Tutor académico: Federico Martín de la Escalera.