



UNIVERSITAT  
POLITÈCNICA  
DE VALÈNCIA



ESCUELA TÉCNICA  
SUPERIOR INGENIEROS  
INDUSTRIALES VALENCIA

Curso Académico:



## RESUMEN

El propósito de este TFG es el diseño de algoritmos para estabilización y control de la trayectoria de un cuatrirrotor sin piloto. Para ello se plantea la programación de un modelo dinámico no lineal que represente el comportamiento de un cuadricóptero y que permita un diseño realista y su validación sin necesidad de realizar ensayos “destruictivos” sobre el proceso real. Se trata de un desarrollo complejo, multivariable y no-lineal. La primera capa de control consiste en introducir una serie de controladores para estabilizar los ángulos, esta capa requerirá además un desacoplamiento de las variables, abordando en la segunda capa el control de la trayectoria. El ajuste de los parámetros de estos reguladores se realiza mediante análisis en el lugar de las raíces, pudiendo aplicar este método a partir de los modelos lineales obtenidos para el ajuste de controladores.

Previamente a la obtención de los modelos lineales para el control de estabilización, se ha desacoplado el sistema. El nuevo bloque de desacoplamiento permite introducir pares y empuje en lugar de las velocidades de los rotores, dando como resultado los ángulos de Euler (pitch, roll y yaw) de manera independiente. A parte de encontrar el sistema de ecuaciones que lo rige, también se persigue programar en el código estándar Modelica.

Este tipo de lenguaje está ampliamente extendido en la industria por su capacidad para representar sistemas dinámicos complejos, ya que por razones de seguridad y altos costes en la producción de prototipos, las simulaciones sofisticadas son cada vez más demandadas. Por ejemplo, Modelica sirve de motor de simulación del programa CATIA, uno de los más importantes softwares de CAD.

Mediante la programación de bloques que caracterizan la dinámica de un cuatrimotor, se puede analizar el comportamiento que ofrecería éste ante señales referencia de trayectoria (X, Y y Z) y ángulo de guiñada (yaw). Los desplazamientos se han conseguido controlar por medio de reguladores PD (proporcional y derivativo) que establecen las referencias de los controladores de la estructura en cascada. En este proceso los controles de trayectoria establecen las referencias de ángulos. Los controladores angulares (también de tipo PD) se encargan de estabilizar y orientar el cuatrirrotor. Las salidas de los controles de ángulos (pares en cada uno de los ejes) son traducidas a acciones sobre cada motor gracias a la aplicación de las ecuaciones de desacoplamiento.

Por lo tanto, las tareas que se han tenido que realizar han sido:

- Búsqueda y comprensión de las ecuaciones que lo modelan.

- Selección de un lenguaje de simulación.

- Búsqueda de un compilador adecuado para el tipo de lenguaje seleccionado.

- Programación de bloques que describen la dinámica del cuatrimotor.

- Comprobación de los resultados obtenidos en la simulación.

- Desarrollo del bloque de desacoplamiento.

- Obtención de los modelos lineales para el diseño del sistema de control de estabilización a partir del modelo desacoplado.

- Determinación del tipo de controlador y ajuste de sus parámetros a partir de los modelos lineales, atendiendo las especificaciones estáticas y dinámicas.

- Obtención de los modelos lineales para el diseño del sistema de control de la trayectoria.

- Determinación del tipo de controlador y ajuste de sus parámetros a partir de los modelos lineales.

Incluir en el modelo de OpenModelica todos los bucles de control.  
Validación en la plataforma de simulación de todos los desarrollos realizados.

Concluyendo este apartado, el objetivo colateral del trabajo ha sido el conocer la escritura y funcionamiento del lenguaje Modelica (muy extendido en software comercial de simulación dinámica). Se han empleado algunas herramientas disponibles de la *Modelica Standard Library*, consiguiendo alcanzar una visión general del potencial que posee éste código. Cabe destacar que el software libre empleado para la programación tiene un optimizador bastante inestable, por lo que no se han podido aplicar algoritmos genéticos en la búsqueda de los parámetros óptimos del controlador (objetivo que se planteo inicialmente) por lo que se han utilizado herramientas de análisis y diseño.



## **INDICE GENERAL**

Documento N°1: Memoria

Documento N°2: Presupuesto

Documento N°3: Anexo de Desarrollo Matemático

Documento N°4: Anexo de Programación

