

LABORATORIO PARA EL DESARROLLO DE SISTEMAS DE NAVEGACIÓN Y CONTROL PARA UAVS MULTIRROTOR

David Gallarta Sáenz , Javier Rico-Azagra, Silvano Nájera Canal, Montserrat Gil-Martínez
david.gallarta@alum.unirioja.es, {javier.rico,silvano.najera,montse.gil}@unirioja.es

Resumen

Este trabajo presenta el desarrollo de un laboratorio para la investigación en sistemas de navegación y control de UAVs (Unmanned Aerial Vehicle) multirrotores. El laboratorio está compuesto por un recinto en el que realizar las pruebas de vuelo con seguridad, un sistema de captura que permite determinar la posición y orientación espacial del vehículo, las aeronaves junto con sus controladoras de vuelo, y una estación de tierra en la que se centraliza la información. La configuración y programación de todos los elementos del sistema se realiza íntegramente empleando Matlab-Simulink. El laboratorio permite reducir de forma notable el tiempo de desarrollo de nuevos algoritmos de estimación y control. Además, se simplifica el análisis de las pruebas de vuelo al disponer de información más precisa y centralizada en un único equipo. En el documento se presenta el diseño de la infraestructura hardware y software que componen el laboratorio. Por último, se muestran resultados obtenidos en una prueba de vuelo y un ejemplo de aplicación para la validación de estimadores de actitud.

Palabras clave: Vehículos aéreos no tripulados (UAV), Control de actitud, Control de posición, Sistemas de estimación, Ingeniería de Control.

1. Introducción

La industria de los vehículos aéreos no tripulados (UAVs), también conocidos como drones, ha experimentado un gran crecimiento en los últimos años. A pesar de que existen multitud de tipos de aeronaves [1], los vehículos de tipo multirrotores [2] son los que actualmente presentan mayor potencial de desarrollo, debido a su facilidad de uso y a que presentan características VTOL (*Vertical Take-Off and Landing*) que permiten su empleo en un gran número de aplicaciones reales [3, 4, 5, 6]. Por todo ello, la popularidad de este tipo de equipos ha aumentado notablemente en la comunidad científica y el ámbito profesional.

Pese a todo, la introducción de estos vehículos en

el sector civil no ha alcanzado su cenit, debido a que existen multitud de problemas que deben ser resueltos y capacidades que deben ser mejoradas. Entre los principales retos destacan aumentar la carga útil, mejorar la autonomía de vuelo, incrementar la seguridad, desarrollar sistemas de navegación más precisos que permitan vuelos autónomos en espacios reducidos y en interiores, mejorar los sistemas de control de actitud, orientación y posición, y desarrollar estrategias de vuelo coordinado.

Algunos de los problemas descritos anteriormente son puramente tecnológicos y solo pueden ser solventados mediante el desarrollo de nuevas tecnologías [7, 8]. Sin embargo, otros dependen directamente de los algoritmos que gobiernan el UAV en diferentes niveles jerárquicos. Algunos trabajos en esta línea son [9, 10].

Uno de los principales obstáculos para la investigación en estos problemas de control es el tiempo que debe invertirse en las pruebas de vuelo. Téngase en cuenta que cada vez que el investigador desea validar en el equipo real un nuevo desarrollo debe desplazarse para realizar una prueba de vuelo, con la que generar datos que posteriormente deben ser procesados y analizados. Todo esto supone un gran coste en horas de investigación, que pueden reducirse si se disponen de instalaciones *indoor* en las que poder realizar las pruebas de vuelo de forma segura y en un menor tiempo.

Bajo las premisas anteriores, este trabajo presenta el diseño y construcción de un laboratorio para el desarrollo de sistemas de navegación y control para UAVs multirrotores de pequeño tamaño. La solución propuesta se compone de un espacio de trabajo en el que volar las aeronaves con seguridad, un sistema de *tracking* por visión que permite determinar la posición y la orientación del vehículo en tiempo real, las aeronaves y sus controladoras de vuelo, la estación de tierra en la que se centraliza la información, y toda la arquitectura software que permite la comunicación entre los diferentes elementos. La configuración y programación del sistema micro-controlador empleado en los vehículos, de las comunicaciones, así como del sistema encargado de centralizar la información se desa-

rolla íntegramente en *Matlab-Simulink*. De este modo se mantiene un entorno de trabajo unificado que simplifica el trabajo. El objetivo principal del sistema es reducir el tiempo de desarrollo de nuevas estrategias de navegación y control para UAVs multirrotor, eliminando las fases de pruebas en exterior. Además el sistema permite recolectar datos de vuelo en tiempo real, tanto del sistema de captura por visión como del UAV. Toda esta información facilita un análisis más preciso del comportamiento del vehículo y una interpretación de los resultados con mayor fiabilidad. Por último, el sistema permite cerrar los lazos de control que gobiernan el guiado autónomo y desarrollar nuevas estrategias de guiado y aplicaciones de vuelo colaborativo.

El documento se organiza en cinco secciones. En la segunda se describe el sistema, los elementos *hardware* que intervienen en el mismo y su instalación. En la tercera se describe la arquitectura *software* que permite comunicar todos los elementos. La sección cuarta muestra los resultados obtenidos en dos aplicaciones. Por último, la quinta contiene las conclusiones.

2. Descripción del sistema

Como se ha explicado anteriormente, el laboratorio está compuesto por una zona de vuelo, en la que se han instalado las medidas de seguridad necesarias para operar los UAVs con seguridad, un sistema de captura por visión *OptiTrack* [11] que permite determinar la posición y actitud de uno o varios vehículos en tiempo real, los UAVs junto con sus controladoras de vuelo, y una estación de tierra que centraliza la información.

El funcionamiento del conjunto se ilustra en la Figura 1. Por un lado, el sistema de captura *OptiTrack* es el encargado de determinar la posición y actitud del vehículo, siendo esta información transferida a la estación de tierra. La comunicación entre el sistema *OptiTrack* y *Simulink* permite actualizar los datos con una cadencia de 10 ms. Por lo tanto, los datos obtenidos pueden ser empleados para cerrar los lazos de control o para monitorizar el estado del UAV.

Por otro lado, los UAVs establecen una comunicación bidireccional e inalámbrica con la estación de tierra, que nuevamente recoge los datos a través de *Simulink*. Mediante el empleo de este enlace de comunicaciones los UAVs pueden enviar la información de sus sensores o del estado interno de su sistema. Del mismo modo, la estación de tierra puede transferir en tiempo real información a la aeronave. Esta información puede emplearse para conmutar los modos de operación del equipo o

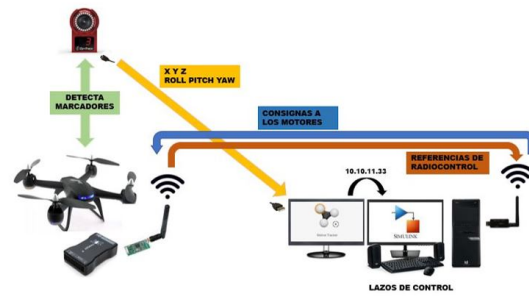


Figura 1: Esquema de funcionamiento del laboratorio.

para cerrar lazos de control con los resultados generados en la estación de tierra. En este caso la tasa de refresco depende de la cantidad de datos a transferir, ya que el enlace de comunicaciones presenta una velocidad máxima de 115200 baudios. Como ejemplo, en las aplicaciones presentadas en la sección 4 se ha empleado una frecuencia de muestreo de 20 ms. Con esta configuración, pueden transmitirse en cada envío 57 datos en formato *single* (230 bytes).

Por último, la estación de tierra actúa como elemento centralizador de información. En ella se ejecutan bajo *Matlab-Simulink* las aplicaciones desarrolladas para cada uno de los experimentos. Las tareas a ejecutar por la estación de tierra pueden ser diversas en función del objetivo del experimento a realizar. Se establecen diferentes niveles de integración, los cuales determinan el grado de complejidad de la aplicación y su importancia en el adecuado funcionamiento del sistema. A continuación se describen estos niveles:

- Nivel 0: Registro de datos. La estación de tierra realiza funciones de *data logger*. Por lo tanto, su única misión es la de registrar los datos generados por la controladora de vuelo y por el sistema *OptiTrack*. En estos casos el UAV debe ser totalmente funcional, ya que trabaja de forma autónoma. Este nivel puede emplearse una vez que los diseños se encuentran implementados en el UAV para validar su correcto funcionamiento.
- Nivel 1: Pasarela de comunicaciones. En este caso la estación de tierra actúa como interfaz de comunicaciones entre el sistema de captura de datos por visión y el UAV. Este modo de operación se emplea preferentemente para realizar pruebas de sistemas de guiado autónomo, cerrando las medidas del sistema *OptiTrack* los lazos de control de posición.
- Nivel 2: Sistema de control. En este caso la estación de tierra actúa como controladora de

vuelo del UAV. La estación de tierra incorpora el sistema de navegación y control del vehículo. Para ello puede emplear los datos provenientes del sistema *OptiTrack* y de los sensores de la controladora de vuelo, que son transmitidos en modo *raw* (*en crudo*) a la estación de tierra. Todas las acciones de control son calculadas en la estación de tierra y transferidas a la aeronave. En este modo de operación la controladora de vuelo únicamente transfiere las acciones de control recibidas a los motores correspondientes. La ventaja principal de este modo es la capacidad para ajustar nuevos diseños en un menor tiempo, ya que éstos no deben ser compilados y programados en la controladora de vuelo. El principal problema aparece en las comunicaciones, ya que pueden impactar negativamente en el comportamiento del vehículo.

Para que todo el sistema funcione de forma adecuada es necesario que los elementos descritos anteriormente cumplan una serie de requisitos. A continuación, se describen las características a cumplir por cada uno de ellos y la solución adoptada en el desarrollo realizado.

2.1. Zona de vuelo

Para que el laboratorio sea funcional, la zona de vuelo debe ser lo suficientemente amplia como para poder efectuar maniobras con UAVs de pequeño tamaño. En principio, para poder probar estrategias de control con ciertas garantías debe disponerse de una área de vuelo de al menos $5\text{ m} \times 5\text{ m} \times 2,5\text{ m}$. Además, todo el área de vuelo debe ser visible por el sistema de cámaras *OptiTrack*.

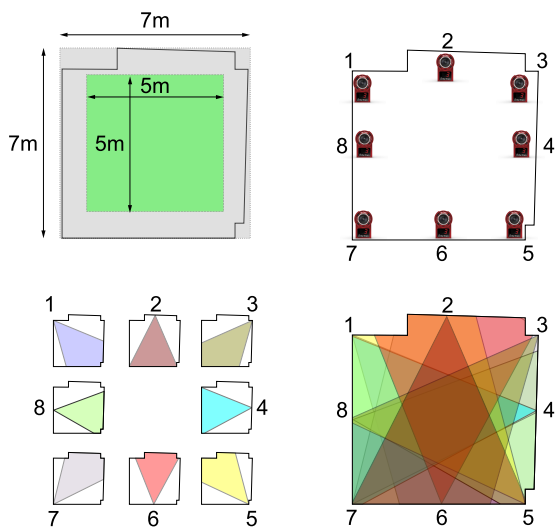


Figura 2: Descripción de la zona de captura.

En el laboratorio donde se ha instalado el sistema se dispone de un área irregular de aproximadamente $7\text{ m} \times 7\text{ m} \times 3\text{ m}$. En la esquina superior izquierda de la Figura 2 se muestra el perímetro disponible junto con el área mínima de vuelo. Para configurar el área de captura del sistema *OptiTrack* se han empleado ocho cámaras *Flex 3*, que se han ubicado y numerado según el esquema mostrado en la esquina superior derecha de la Figura 2. Cada una de las cámaras presenta una apertura focal de 58° , por lo que estas cámaras deben ser ubicadas de forma estratégica para que se cubra todo el área de vuelo. Téngase en cuenta que cada punto del área de vuelo debe ser visto por al menos tres cámaras para que el sistema funcione correctamente. En la esquina inferior izquierda de la Figura 2 se muestra la orientación de cada una de las cámaras junto con su área de captura. Por último, en la esquina inferior derecha de la Figura 2 se muestra la superposición de las áreas capturadas por cada una de las cámaras. En ella puede comprobarse como todo el área de vuelo cumple con los requisitos expuestos anteriormente, garantizando que el sistema es capaz de operar en todo el recinto. Una vez configurado y calibrado, el sistema trabaja presentando errores de estimación de posición y actitud inferiores a $0,2\text{ mm}$ y $0,1^\circ$.

Un factor que a priori puede parecer poco importante es la forma de sujeción de las cámaras, pero resulta fundamental para un correcto funcionamiento del sistema. El anclaje debe permitir que las cámaras queden fijas en una orientación durante el vuelo, pero que a su vez pueda ser modificada en un futuro. Otro aspecto fundamental para la calibración de las cámaras es el ajuste en altura, ya que en caso de no ser adecuado se perderá volumen de captura en la parte inferior o superior del recinto. Por estos motivos se ha realizado un soporte para cámaras que permite la regulación de la actitud y altura de las mismas. El soporte desarrollado consiste en un tubo metálico hueco, de 1 m de longitud y 5 cm de diámetro, con una base cuadrada perforada que permite su fijación al techo. La cámara se fija al tubo mediante una pinza de plástico que incorpora una rótula. En la Figura 3 se observa el sistema final una vez fijado al techo.



Figura 3: Ubicación del sistema *OptiTrack*

Para garantizar la seguridad se ha colocado una red alrededor de todo el perímetro. Este elemento, además de proteger a las personas que se encuentran en el exterior del recinto, permite atrapar los UAVs cuando estos pierden el control, ya que se quedan enganchados en la misma debido al movimiento de las hélices. Con este sistema se evitan los impactos contra superficies duras y se alarga la vida útil de los vehículos. Para que la red de protección pueda cumplir con su propósito se recomienda separarla al menos 20 cm de cualquier pared u objeto. La Figura 3 muestra una imagen del laboratorio con la red instalada.

2.2. Cableado y sincronización

Cada una de las cámaras *Flex 3* mencionadas en la sección anterior debe conectarse a un concentrador *OptiHub* empleando una conexión USB 2.0 (ver Figura 4). Cada módulo *OptiHub* admite la conexión de hasta seis cámaras. Este dispositivo es el encargado de dotar de alimentación a las cámaras, y de sincronizar y transmitir la información generada a la estación de tierra utilizando el protocolo de sincronización *OptiSync*. En caso de disponer de una zona de captura que contenga más de seis cámaras, estas deberán conectarse a diferentes módulos *OptiHub*, que a su vez deberán estar interconectados para poder sincronizar la toma de imágenes.

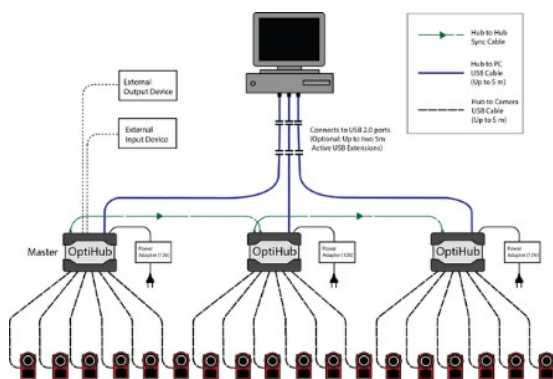


Figura 4: Esquema para la interconexión de las cámaras *Flex 3*. Figura extraída de [11].

A la hora de distribuir las cámaras en el recinto, además del estudio realizado en la sección anterior, es necesario tener en cuenta las limitaciones físicas de la transmisión de datos mediante enlaces USB. El emparejamiento entre cada cámara y su *OptiHub* debe acometerse con un cable de extensión igual o inferior a 5 metros. Por lo tanto, es necesario realizar una planificación adecuada de la distribución de los diferentes *OptiHub* y de las cámaras alrededor del recinto. Debe tenerse en cuenta que para la conexión de los diferentes *Op-*

tiHub al PC *host*, sí es posible emplear cableado que supere los 5 m de longitud.

Como puede comprobarse en la Figura 3, el recinto en el que se ha llevado a cabo la instalación presenta falso techo, el cual se ha aprovechado para distribuir el cableado permitiendo situar todos los *OptiHub* cerca de las cámaras a las que dan servicio. Además, el empleo de falso techo permite eliminar todos los cables de la zona de vuelo, logrando un espacio diáfano y libre de obstáculos.

2.3. Multirrotores

Se emplean multirrotores de diferentes envergaduras, en concreto de 250 mm, 450 mm y 550 mm, en configuraciones de cuatro y seis ejes. No obstante, se recomienda emplear UAVs de 250 mm de tamaño, ya que presentan un comportamiento más ágil y apropiado para el espacio de trabajo disponible. Todos los vehículos empleados se encuentran gobernados con la controladora de vuelo APM 2.5 (*Ardupilot Mega 2.5*) [12], que corresponde con una modificación del micro-controlador Arduino Mega 2560 realizada por 3DR Robotics. Se ha decidido emplear la controladora de vuelo APM 2.5 debido a su facilidad para ser programada empleando *Matlab-Simulink*. La *toolbox AMP2* [13] permite programar el *firmware* de la controladora de vuelo empleando *Simulink*. Esta característica simplifica el proceso de diseño de nuevos algoritmos de control, manteniendo un entorno de trabajo unificado.

Por último, hay que destacar que todos los UAVs empleados en las pruebas de vuelo deben de estar equipados con marcadores *OptiTrack*, para que puedan ser identificados dentro del espacio habilitado. Como número mínimo, se recomienda que el dron presente cuatro marcadores. De este modo, aunque uno de ellos quede oculto para las cámaras, el sistema seguirá siendo capaz de localizar el UAV. La disposición de dichos marcadores también es relevante, siendo la mejor opción distribuirlos de forma aleatoria y evitar que estos formen cuerpos regulares. La Figura 5 ilustra la distribución de los marcadores instalados en un hexacoptero de 550 mm de envergadura. En este UAV se han empleado nueve marcadores de 12,7 mm de diámetro. En la parte derecha de la Figura 5 se observa como *Motive* genera un cuerpo sólido a partir de los marcadores detectados.

3. Sistema de comunicaciones

Como se ha descrito en secciones anteriores, tanto los UAVs como el sistema de captura *OptiTrack* deben establecer comunicación con la estación de tierra en la que se centraliza la información. En la

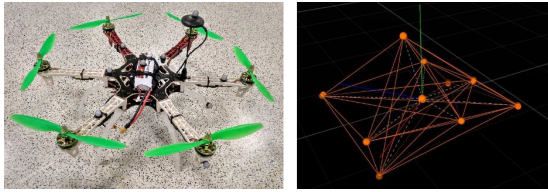


Figura 5: Distribución de marcadores en un UAV y representación del mismo en *Motive*.

presente sección se describe la arquitectura software empleada para realizar dicha comunicación.

3.1. Comunicación Motive-Simulink

El sistema *OptiTrack* necesita su propio software, *Motive*, para controlar sistemas de captura de movimiento en aplicaciones de *tracking*. Este software debe ser empleado por el usuario para calibrar y configurar el sistema, crear los objetos tridimensionales, y procesar en 3D los datos obtenidos. Junto con *Motive*, *OptiTrack* facilita el entorno de desarrollo *NatNet*, que consta de un conjunto de herramientas cliente-servidor que emplean protocolo UDP y comunicación de tipo *multicast* o *unicast*. Empleando estas herramientas pueden enviarse los datos registrados en las pruebas de vuelo desde un servidor (*Motive*) a aplicaciones cliente (en la aplicación descrita *Matlab-Simulink*).

Para recibir los datos en *Simulink* desde *Motive* se emplea una *S-Function* que utiliza la API facilitada por *NatNet*. En concreto, se emplean los recursos disponibles en *NPTrackingTools.h* y *RigidBodySettings.h*, que contienen las funciones para obtener las características de los cuerpos rígidos detectados por las cámaras. A partir de estas herramientas se obtienen las posiciones tridimensionales y los cuaternios que definen la orientación espacial del vehículo. Estos últimos son transformados en ángulos de Euler y enviados junto con las posiciones espaciales a la salida de la *S-function* desarrollada.

3.2. Comunicación Simulink-UAV

Como ya se ha comentado, cada UAV establece una comunicación bidireccional con la estación de tierra. En principio, el UAV podrá enviar a la estación de tierra la información generada por los sensores de la controladora de vuelo y señales recibidas a través de la emisora de radio. Por otro lado, la estación de tierra enviará al UAV las consignas para los diferentes modos de operación o las acciones de control para los motores.

Para lograr la comunicación en tiempo real se emplea un enlace de telemetría que opera a 915 MHz. Dicho enlace se conecta a la UART0 de la controladora

de vuelo APM y al puerto USB de la estación de tierra. Desde el punto de vista del software, el problema reside en establecer una comunicación a través de un puerto serie. Para ello deben desarrollarse módulos software que deben ser integrados en el *firmware* de la controladora de vuelo y en el programa que se ejecuta en la estación de tierra.

3.2.1. Envío de datos desde la estación de tierra al UAV

En el PC de la estación de tierra se emplea el bloque *Serial Send* que a través del puerto COM definido envía los datos deseados en forma de vector. Todos los datos transmitidos se envían como bytes sin codificación. Para ello, se emplea un algoritmo que trabaja de la siguiente manera: en primer lugar, los datos que se quieren transmitir (de tipo *single*) se trocean en paquetes de ocho bits codificados como *uint8*. A continuación, para establecer una comunicación más robusta se añade la cabecera del mensaje. En este caso se ha optado por emplear dos @@ (64 codificado en *uint8*). La cabecera es fácilmente modificable y escalable para sistemas que presenten más de un UAV. Por último, se aplica un código CRC para la detección de errores de comunicación. El código CRC empleado está formado por dos bytes que se añaden al final de la trama. La Figura 6 muestra cómo se genera la trama a enviar.

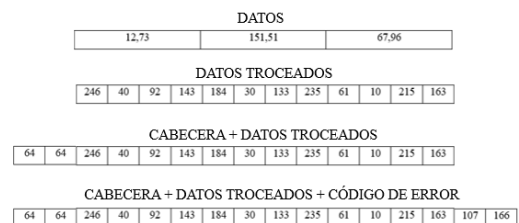


Figura 6: Generación de tramas para el envío de datos.

El *firmware* de la controladora de vuelo emplea el bloque *Serial Read*, que extrae byte a byte los datos que se almacenan en el *buffer* de entrada del puerto serie. Para recomponer la trama transmitida es necesario emplear un algoritmo encargado de extraer los datos de forma inteligente. Para ello, el algoritmo busca en primer lugar los dos bytes que marcan el inicio de la comunicación. Una vez detectada la cabecera de la trama se almacenan los valores correspondientes a los datos troceados, doce en el ejemplo, y al código de error, dos. Finalizado este proceso se generan los dos bytes del código CRC con los datos recibidos. Si el código de error recibido coincide con el calculado, la trama

se considera correcta. Por último, se recomponen los datos en formato *single* a partir de los bytes disponibles y se pasan a la salida de la función.

3.2.2. Envío de datos desde el UAV a la estación de tierra

El *firmware* emplea el bloque *Serial Write* para enviar a través de la UART0 el vector de datos que tiene a su entrada. El algoritmo para realizar el envío de datos es igual al descrito en la sección anterior y se mantiene el mismo formato de trama y el método en el que se descomponen los datos.

En la aplicación *Simulink*, que se ejecuta en la estación de tierra, se emplea el bloque *Serial Receive*, el cual permite definir el tamaño del vector de datos que se desean leer del *buffer* de entrada. Por lo tanto, en este caso cambia el modo en el que se efectúa la re-composición de las tramas y la filosofía de programación. El algoritmo empleado busca dentro del vector de datos recibidos los bytes que marcan el inicio de la trama. Una vez detectada la cabecera se almacenan el resto de bytes de la misma (en el ejemplo doce datos provenientes de la descomposición de tres datos *single*, y dos del código CRC). Los datos a almacenar pueden pertenecer a dos vectores de datos muestreados en intervalos diferentes. Por lo que para solucionar este problema se emplea la siguiente estrategia: en primer lugar se genera un vector auxiliar en el que se almacenan los valores extraídos del primer vector de datos. Posteriormente, dicho vector auxiliar se rellena con los valores que se encuentran al comienzo del segundo vector de datos. Una vez se complete el vector auxiliar, se computa el código de error y se compara con el código de error recibido por el puerto serie. Si ambos coinciden se considera que la trama es correcta y se procesan las salidas. Un ejemplo se muestra en la Figura 7.

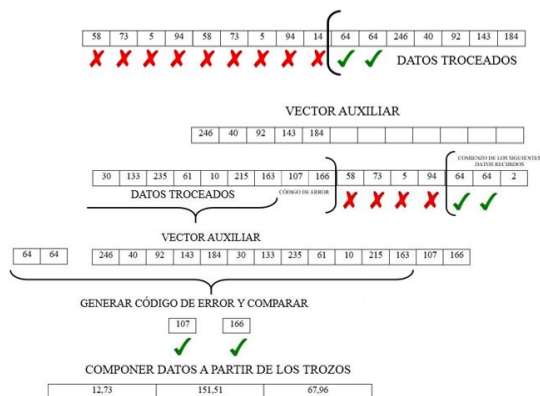


Figura 7: Recepción de tramas en la estación de tierra.

4. Ejemplos de aplicación

4.1. Ejemplo 1: registro de datos de vuelo

En este primer ejemplo se muestra cómo el sistema es capaz de registrar los datos de un vuelo. Esta es la aplicación más sencilla para la que puede emplearse el laboratorio desarrollado. Para ello se realiza un vuelo con un UAV multirrotor dentro de la zona habilitada. En dicho vuelo no se ha seguido una trayectoria concreta, únicamente se ha mantenido el vehículo en la zona de captura describiendo movimientos aleatorios.

Los resultados obtenidos en el experimento se muestran en la Figura 8. En la gráfica superior se observan los valores registrados para la actitud de la aeronave. En ellos puede comprobarse como el vehículo presenta inclinaciones máximas en el entorno de los $\pm 20^\circ$. Esto es debido a que el vuelo efectuado se ha realizado sin maniobras agresivas. En la gráfica inferior de la Figura 8 se muestran los desplazamientos en los tres ejes de movimiento. En ella se observa como el movimiento en los ejes *X* e *Y* sigue trayectorias que producen desplazamientos de un lado al otro del área de vuelo. Por el contrario, en el eje *Z* se observa como el UAV se ha mantenido a una altura inferior a 1 m, hasta que en los instantes finales ha descendido para aterrizar.

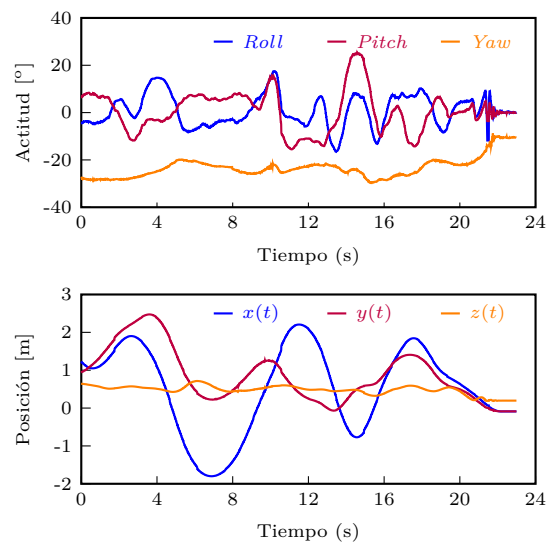


Figura 8: Datos registrados en la prueba de vuelo.

4.2. Ejemplo 2: validación del sistema de estimación

En este segundo ejemplo se presenta una de las potenciales aplicaciones del sistema desarrollado. Uno de los elementos más importantes de todo vehículo autónomo es el sistema de estimación de

actitud. En los UAVs este sistema es el encargado de inferir el estado de la aeronave a partir de las medidas obtenidas con sensores de bajo coste (giroscopios, acelerómetros y magnetómetros). Para lograr estos objetivos se emplean estrategias de fusión sensorial. Actualmente siguen desarrollándose nuevos estimadores que persiguen mejorar las medidas añadiendo medidas de nuevos sensores o mejorando los algoritmos ya existentes. En muchas ocasiones, los estimadores diseñados no pueden validarse correctamente por no disponer de sensores patrón con los que comparar las medidas. Este problema impide comprobar si realmente son adecuados para su empleo en multirrotores reales.

El laboratorio desarrollado permite validar de forma fiel el comportamiento de dichos estimadores, ya que el sistema de captura por visión puede actuar como sensor patrón con el que contrastar los resultados arrojados por el estimador. En la Figura 9 pueden compararse los resultados obtenidos durante un vuelo real y se comprueba la similitud entre las medidas ofrecidas por la aeronave y por el sistema *OptiTrack*. Sin embargo, pueden observarse errores en la estimación del ángulo de *roll* que llegan a ser superiores a $\pm 10^\circ$. Este error es debido a que la frecuencia de corte del filtro del estimador (se ha empleado el estimador propuesto en [9]) es demasiado alta¹. Cuando la frecuencia de corte se selecciona mal, las aceleraciones que se producen durante el desplazamiento del UAV producen errores de estimación. Este ejemplo muestra las posibilidades de la instalación desarrollada para detectar problemas que de otro modo pueden pasar inadvertidos. Nótese que el experimento se ha realizado cerrando el lazo de control del UAV con las medidas del estimador. A pesar de presentar errores de estimación, el UAV es totalmente operativo y el operador de vuelo no es capaz de detectar la distorsión, pero esta puede afectar a los sistemas de control que se empleen en niveles jerárquicos superiores.

5. Conclusiones

Se ha presentado un laboratorio para el desarrollo de estrategias de navegación y control para UAVs multirrotores. El laboratorio emplea un sistema de captura de imágenes *OptiTrack* que permite realizar el *tracking* de objetos en el espacio. Para llevar a cabo las pruebas de vuelo se emplean UAVs de bajo coste gobernados con la controladora de vuelo APM 2.5. La coordinación entre los dos sistemas se realiza en una estación de tierra en la que se concentra la información. La instalación cuenta con las medidas de seguridad necesarias para

¹Seleccionada de forma errónea para el ejemplo propuesto.

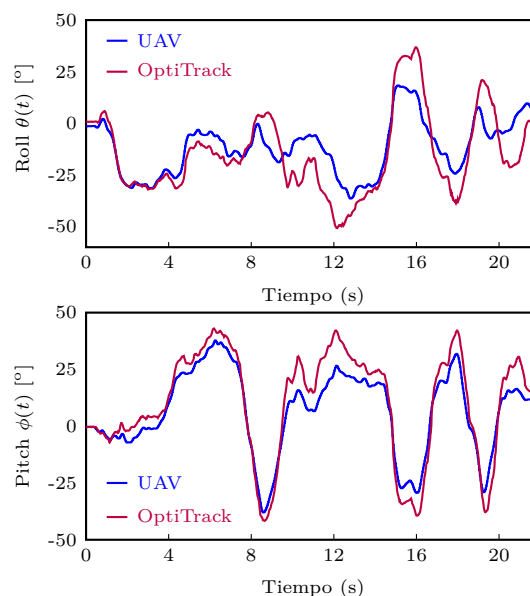


Figura 9: Validación de un estimador de actitud.

efectuar las pruebas de vuelo sin comprometer la integridad física del usuario.

Entre las ventajas de la instalación destaca la facilidad de construcción y configuración, su versatilidad, y el empleo de un entorno de desarrollo unificado bajo *Matlab-Simulink*. Estas características permiten reducir el número de horas necesarias desde el diseño de los algoritmos hasta su implementación en dispositivos reales.

Por último, se han presentado dos ejemplos en los que el entorno propuesto puede ser de utilidad. En el primer ejemplo se muestran los datos generados durante una prueba de vuelo de un UAV. El segundo ilustra cómo el sistema puede ayudar a sintonizar los sistemas de estimación de actitud incorporados en la controladora de vuelo de la aeronave.

Agradecimientos

Los autores agradecen la ayuda prestada por el Gobierno de La Rioja a través del proyecto de I+D ADER 2017-I-IDD-00035 y por la Universidad de La Rioja a través de la Ayuda a Grupos de Investigación REGI 2018/42.

English summary

LABORATORY FOR THE DEVELOPMENT OF NAVIGATION AND CONTROL SYSTEMS FOR MULTIROTOR UAVS

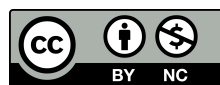
Abstract

This work presents the development of a laboratory for research on navigation and control systems of multirotor Unmanned Aerial Vehicles (UAV). The laboratory is composed of an enclosure in which flight tests are carried out with safety, a capture system that allows determining the position and orientation of the vehicle, the aircraft with their flight controllers, and a ground station where the information is centralized. The configuration and programming of all system elements is done using Matlab-Simulink. The laboratory allows to significantly reduce the development time of new estimation and control algorithms. In addition, it simplifies the analysis of flight tests with more accurate and centralized information in a single team. The document presents the design of the hardware and software infrastructures that make up the laboratory. Finally, results obtained in a flight test and an application example for the validation of attitude estimators are shown.

Keywords: Unmanned Aerial Vehicle (UAV), Attitude Control, Position Control, Estimation Systems, Control Engineering.

Referencias

- [1] M. Hassanalian and A. Abdelkefi. Classifications, applications, and design challenges of drones: A review. *Progress in Aerospace Sciences*, 91:99 – 131, 2017.
- [2] R. Mahony, V. Kumar, and P. Corke. Multirotor Aerial Vehicles: Modeling, Estimation, and Control of Quadrotor. *IEEE Robotics & Automation Magazine*, 19:20–32, 2012.
- [3] I. Sa and P. Corke. *Vertical infrastructure inspection using a Quadcopter and shared autonomy control*, volume 92 of *Springer Tracts in Advanced Robotics*. 2014.
- [4] S. Candiago, F. Remondino, M. De Giglio, M. Dubbini, and M. Gattelli. Evaluating multispectral images and vegetation indices for precision farming applications from uav images. *Remote Sensing*, 7(4):4026–4047, 2015.
- [5] S. Siebert and J. Teizer. Mobile 3d mapping for surveying earthwork projects using an unmanned aerial vehicle (uav) system. *Automation in Construction*, 41:1 – 14, 2014.
- [6] Y. Li and C. Liu. Applications of multirotor drone technologies in construction management. *International Journal of Construction Management*, 0(0):1–12, 2018.
- [7] B. Lee, P. Park, C. Kim, S. Yang, and S. Ahn. Power managements of a hybrid electric propulsion system for uavs. *Journal of Mechanical Science and Technology*, 26(8):2291–2299, Aug 2012.
- [8] W. Lu, D. Zhang, J. Zhang, T. Li, and T. Hu. Design and implementation of a gasoline-electric hybrid propulsion system for a micro triple tilt-rotor vtol uav. In *2017 6th Data Driven Control and Learning Systems (DDCLS)*, pages 433–438, May 2017.
- [9] J. Rico-Azagra, M. Gil-Martínez, R. Rico-Azagra, and P. Maisterra. Low-cost attitude estimation for a ground vehicle. In *Robot 2015: Second Iberian Robotics Conference*, Advances in Intelligent Systems and Computing, pages 121–132, Switzerland, 2016. Springer International Publishing.
- [10] C. Fu, W. Hong, H. Lu, L. Zhang, X. Guo, and Y. Tian. Adaptive robust backstepping attitude control for a multi-rotor unmanned aerial vehicle with time-varying output constraints. *Aerospace Science and Technology*, 78:593 – 603, 2018.
- [11] NaturalPoint Inc. <http://optitrack.com/>.
- [12] H. Lim, J. Park, D. Lee, and H. J. Kim. Build your own quadrotor: Open-source projects on unmanned aerial vehicles. *IEEE Robotics and Automation Magazine*, 19(3):33–45, 2012.
- [13] F. Hugon, R.F. Hartley, B. DeRosa, and C. Carvalho. <http://www.mathworks.com/matlabcentral/fileexchange/39037-apm2-simulink-blockset>.



© 2018 by the authors.
Submitted for possible
open access publication
under the terms and conditions of the Creative Commons Attribution CC-BY-NC 3.0 license (<http://creativecommons.org/licenses/by-nc/3.0/>).