

Desarrollo e implementación de una plataforma experimental para la consecución de tareas colaborativas mediante el uso de UAV autónomos

Autor: Félix Orjales Saavedra

Tesis Doctoral UDC / 2017

Directores: Álvaro Deibe Díaz
Fernando López Peña

Programa Oficial de Doctorado en Ingeniería Industrial





D. Fernando López Peña, Profesor Titular de Universidad del Departamento de Ingeniería Naval y Oceánica de la de la Universidade da Coruña,

D. Álvaro Deibe Díaz, Profesor Titular de Universidad del Departamento de Métodos Matemáticos y de Representación de la Universidade da Coruña,

CERTIFICAN:

Que la memoria titulada “Desarrollo e implementación de una plataforma experimental para la consecución de tareas colaborativas mediante el uso de UAV autónomos” ha sido realizada por D. Félix Orjales Saavedra bajo nuestra dirección en el Departamento de Ingeniería Industrial II de la Universidade da Coruña, y constituye la Tesis que presenta para optar al grado de Doctor.

Fdo. Fernando López Peña
Codirector de la Tesis Doctoral

Fdo. Álvaro Deibe Díaz
Codirector de la Tesis Doctoral

Agradecimientos

Aunque este trabajo se presente bajo una única autoría, constituye realmente el fruto del esfuerzo de muchas otras personas además del propio, sin las que difícilmente se podría haber llegado a su conclusión.

En primer lugar, me gustaría agradecer a mis directores de tesis, Fernando y Álvaro, por su apoyo, guía y consejo, así como por transmitirme parte de su conocimiento y guiarme por el camino de los “bichos voladores” con enorme paciencia.

En segundo lugar, deseo agradecer a todos mis compañeros del Grupo Integrado Ingeniería con los que he compartido durante los últimos años mis ilusiones, trabajo y esfuerzo, muchos de los que tengo el privilegio de considerar amigos. Más concretamente a Richard, por ofrecerme la oportunidad de realizar este trabajo e introducirme en el mundo de la investigación; a Gervasio y Pilar, por enseñarme tanto sobre algoritmos, programación y por haber ayudado a centrar la realización de este trabajo; al sector hardware, a Fai y Souto, por haberme enseñado la mayor parte de lo que sé de electrónica y mecánica y sacarme una sonrisa cada vez que oigo la canción carros de fuego; a Martín, por los debates para resolver el mundo en el portal y por cuidar mi espalda, sabes que puedes esperar lo mismo cuando lo necesites; y por último a Juan Carlos, por esas partidas de los viernes, por la portada de este trabajo, por el modelo del UAV, por ayudarme a mantener la cordura... seguramente me olvido de varios nombres importantes, ruego me disculpéis... os lo compensaré con unos pinchos y unas cañas a todos, sólo tenéis que reclamarlas!

En tercer lugar, a todos los proyectandos, la gente de prácticas y demás personas a la que en un momento u otro he arrastrado en la espiral de locura para llevar este proyecto a buen término. A Jaime, por la primera versión de la placa de control; a Dani, por las estaciones en tierra; a Rogelio e Isabel, por los dos aviones (bueno ahora uno... fallo mío); a Rubén, por las mejoras en el modelo del UAV; a Pablo, por enseñarme las maravillas del mecanizado en el taller de la EUDI, suerte con la aventura de ser papi!

Finalmente, a mis amigos y familia en general, por entender que haya desaparecido este tiempo. *El que algo quiere, mucho le cuesta*; y el sacrificio necesario para realizar este trabajo era dedicarle muchas de las horas que hubiera pasado con todos vosotros.

A aquellos que ya no están, se os echa de menos en ocasiones y sobre todo se os recuerda con cariño y se os agradece todo el saber transmitido, el cariño y la paciencia a lo largo de los años. Si hay algo al otro lado, nos veremos allí; sino, os agradezco haber compartido vuestro tiempo aquí conmigo.

A mis padres, por su cariño y ánimo incondicional durante todo este tiempo, por entender las ausencias y las tardanzas... sin duda alguna, sois los mejores. Si algún día soy padre, espero llegar a ser la mitad de lo bueno y paciente que habéis sido (y sois) conmigo.

Por último, a Cuqui, por acompañarme en este largo camino desde hace más de tres años, y porque aunque a veces ha sido complicado, siempre ha estado a mi lado.

Resumo

O presente traballo céntrase en facilitar o desenvolvemente de estratexias de coordinación realistas e flexibles que permitan unha cooperación efectiva entre varios UAV para a realización dunha tarefa común. Para conseguilo propónse unha aproximación formada por unha metodoloxía e unha plataforma software. A metodoloxía proposta baséase en tres aspectos fundamentais: unha estrutura normalizada e modular para o software, a independencia da estratexia de coordinación do hardware concreto e a incorporación de forma iterativa e progresiva da complexidade da realidade. A plataforma software desenvólta emprega a realidade mixta para permitir a implementación progresiva da estratexia de coordinación sobre os UAV reais. Coa finalidade de demostrar a utilidade da aproximación proposta desenvólveuse unha plataforma e implementouse sobre a mesma unha aplicación de exemplo para a búsqueda dunha fonte de polución mediante o emprego de UAV autónomos instrumentados.

Resumen

El presente trabajo se centra en facilitar el desarrollo de estrategias de coordinación realistas y flexibles que permitan una cooperación efectiva entre varios UAV para la realización de una tarea común. Para conseguirlo se propone una aproximación formada por una metodología y una plataforma software. La metodología propuesta se basa en tres aspectos fundamentales: una estructura normalizada y modular para el software, la independencia de la estrategia de coordinación del hardware concreto y la incorporación de forma iterativa y progresiva de la complejidad de la realidad. La plataforma software desarrollada emplea la realidad mixta para permitir la implementación progresiva de la estrategia de coordinación sobre los UAV reales. Con la finalidad de demostrar la utilidad de la aproximación propuesta se ha desarrollado una plataforma y se ha implementado sobre la misma una aplicación de ejemplo para la búsqueda de una fuente de polución mediante el uso de UAV autónomos instrumentados.

Abstract

The present work focuses on ease the development of realistic and flexible coordination strategies that allow effective cooperation between several UAVs to carry out a common task. To achieve this, we propose an approach formed by a methodology and a software platform. The proposed methodology is based on three fundamental foundations: a standardized and modular structure for software, the independence of the coordination strategy from the concrete hardware and the iterative and progressive incorporation of the complexity of reality. The software platform developed rely in mixed reality to allow the progressive implementation of the coordination strategy on real UAVs. In order to demonstrate the utility of the approach, a platform has been developed and an example application for the search of a source of pollution through the use of autonomous instrumented UAVs has been implemented.

Publicaciones

Durante la realización de esta tesis se han publicado los siguientes trabajos en congresos y revistas científicas:

- Fernando López-Peña, Pilar Caamaño, Gervasio Varela, Félix Orjales and Álvaro Deibe, Setting up a Mixed Reality Simulator for using Teams of Autonomous UAV in Air Pollution Monitoring, *International Journal of Sustainable Development and Planning*, pp 616 - 626, 2016
- Gervasio Varela, Pilar Caamaño, Félix Orjales, Álvaro Deibe, Fernando López-Peña and Richard J. Duro, Differential Evolution in Constrained Sampling Problems, *IEEE World Congress on Computational Intelligence WCCI 2014*, pp 2375 - 2382, 2014
- Gervasio Varela, Pilar Caamaño, Félix Orjales, Álvaro Deibe, Fernando López-Peña and Richard J. Duro, Autonomous UAV Based Search Operations Using Constrained Sampling Evolutionary Algorithms, *Neurocomputing*, pp 54 - 67, 2014
- Gervasio Varela, Pilar Caamaño, Félix Orjales, Álvaro Deibe, Fernando López-Peña and Richard J. Duro, Swarm Intelligence based Approach for Real Time UAV Team Coordination in Search Operations, *Third World Congress on Nature and Biologically Inspired Computing*, pp 372 - 377, 2011

Índice general

1. Introducción	1
2. Objetivos	11
3. Antecedentes	13
3.1. Introducción	13
3.1.1. Historia y evolución de los UAV	14
3.1.2. Los UAV autónomos	16
3.2. Equipos de UAV	18
3.2.1. Tipos de equipos de UAV	18
3.2.2. Aplicaciones potenciales para equipos de UAV	19
3.2.3. Retos para el control cooperativo de múltiples UAV	21
3.3. Estrategias de coordinación de equipos de UAV	23
3.3.1. Técnicas metaheurísticas	25
3.3.2. Algoritmos para la coordinación de UAV	28
3.4. Plataformas para el desarrollo de estrategias de coordinación	29
3.4.1. C3UV	30
3.4.2. Plataforma de la Universidad Técnica Checa	32
3.4.3. A Framework for Distributed Autonomous Systems (AF-DAS)	35
3.5. Conclusiones	37
4. Metodología propuesta	39
4.1. Introducción	39
4.2. Estructura normalizada y modular para el software	40

4.2.1.	Estructura software para los UAV	44
4.2.2.	Estructura software para la estación en tierra	49
4.3.	Independencia del hardware	51
4.3.1.	Niveles de software	53
4.3.2.	Implementación del middleware	55
4.3.2.1.	Middleware para sensores	56
4.3.2.2.	Middleware para actuadores	57
4.3.2.3.	Middleware para unidad de control	58
4.4.	Incorporación iterativa y progresiva de la complejidad real	60
4.4.1.	La realidad mixta	61
4.4.2.	Estrategia para la superación del reality gap	63
4.5.	Conclusiones	66
5.	Diseño y desarrollo de la plataforma software	67
5.1.	Introducción	67
5.1.1.	Visión general	69
5.2.	Estructura normalizada y modular para el software	70
5.2.1.	UAV virtuales	70
5.2.1.1.	Control	72
5.2.1.2.	Estado	76
5.2.1.3.	Seguridad	77
5.2.1.4.	Comunicación	80
5.2.2.	Estación de tierra	81
5.2.2.1.	Control	81
5.2.2.2.	Estado	83
5.2.2.3.	Seguridad	83
5.2.2.4.	Comunicación	84
5.2.3.	Simulador del mundo	84
5.2.4.	Ejemplo de las ventajas de la estructura modular	84
5.2.4.1.	JMonkey	85
5.2.4.2.	Flightgear	85
5.2.4.3.	Conclusiones	87

5.3. Independencia del hardware	87
5.3.1. Estructura del firmware	88
5.3.1.1. Firmware para sensores y transductores	89
5.3.1.2. Firmware para actuadores	90
5.3.1.3. Firmware para unidades de control	90
5.3.2. Estructura del middleware	91
5.3.2.1. Middleware para sensores	93
5.3.2.2. Middleware para actuadores	94
5.3.2.3. Middleware para unidades de control o pilotos automáticos	94
5.3.3. Ejemplo de las ventajas de la independencia del hardware	95
5.3.3.1. Primeros pasos, DMD y Eagle Tree	96
5.3.3.2. La solución, lógica de control propia	97
5.3.3.3. Conclusiones	98
5.4. Incorporación iterativa y progresiva de la complejidad real	99
5.4.1. UAV virtuales	99
5.4.2. Simulador del mundo	102
5.4.3. La incorporación de la complejidad real	103
5.4.3.1. Desarrollo e implementación de las estrategias de coordinación	104
5.4.4. Ejemplo de las ventajas de la incorporación de la infor- mación real	106
5.4.4.1. Montaje y modelo inicial de las radiobalizas	106
5.4.4.2. Modelo final de las radiobalizas	107
5.4.4.3. Conclusiones	108
5.5. Conclusiones	109
6. Implementación y pruebas	111
6.1. Introducción	112
6.2. Caracterización del modelo de pluma	114
6.2.1. Descripción del modelo	116
6.2.2. Parametrización para el caso de prueba	119
6.3. UAV reales	121

6.3.1.	Selección de los UAV	122
6.3.2.	Especificaciones de los UAV diseñados	123
6.3.3.	Electrónica de vuelo integrada	124
6.3.3.1.	Electrónica de control	124
6.3.3.2.	Comunicaciones	128
6.3.3.3.	Sensor de CO ₂	129
6.3.4.	Seguridad y prevención de accidentes	129
6.4.	UAV simulados	131
6.4.1.	Elección del software de simulación	131
6.4.2.	Modelado del UAV	131
6.5.	Estrategia de coordinación	133
6.5.1.	El muestreo restringido o Constrained Sampling	134
6.5.2.	Constrained Sampling Differential Evolution (CS-DE)	138
6.6.	Pruebas	139
6.6.1.	Vuelos para el ajuste del controlador	140
6.6.2.	Vuelos para el ajuste del entorno virtual	141
6.6.2.1.	Ajuste del viento real	142
6.6.2.2.	Ajuste del modelo de la pluma	143
6.6.3.	Vuelos realidad mixta en As Pontes	146
6.6.3.1.	Finalización prematura de las pruebas por el cambio de legislación	147
6.6.4.	Reajuste de las pruebas para la adaptación a la nueva legislación	148
6.6.4.1.	Vuelos de realidad mixta en Pedroso	150
6.6.4.2.	Incremento progresivo del número de UAV reales	151
6.7.	Conclusiones	151
7.	Conclusiones y trabajo futuro	155
7.1.	Trabajo futuro	157

Capítulo 1

Introducción

Los vehículos aéreos no tripulados o UAV (Unmanned Aerial Vehicle) se definen como vehículos sin tripulación, reutilizables, autopropulsados y capaces de mantener un nivel de vuelo controlado y sostenido. Cabe aclarar que existen diversas formas de referirse a este mismo concepto y se ha optado por la de UAV debido a su brevedad, simplicidad y por tratarse de una de las más usadas en la publicaciones de investigación actualmente [Valavanis and Vachtsevanos, 2015].

Se trata de una definición en la que se engloban vehículos aéreos con peso, tamaño o incluso tecnologías de propulsión diferentes y que, sin embargo, poseen en común la capacidad de desplazarse por el medio aéreo con propulsión propia y sin un piloto a bordo. En lo referente al control, puede tratarse tanto de vehículos controlados remotamente como de vehículos autónomos, capaces de realizar acciones basándose en una inteligencia artificial, siendo estos últimos los que presentan un mayor potencial al no requerir la presencia constante de un operador para cada UAV en vuelo.

Los UAV presentan diversas ventajas y características diferenciadoras frente a otras tecnologías disponibles actualmente. Por una parte, si se comparan con los vehículos aéreos tripulados tradicionales destacan dos ventajas principales: un coste menor, dado que la aeronave puede simplificarse al no contar con un piloto a bordo y una minimización del riesgo, ya que si existe un piloto puede encontrarse en tierra y alejado de la zona de vuelo del UAV. Esto último resulta especialmente importante cuando la misión se desarrolla en entornos difícilmente accesibles o peligrosos tales como ambientes de alta toxicidad, radiactivos, etc. Por otra parte, en comparación con los vehículos terrestres los UAV poseen una mayor movilidad y maniobrabilidad, lo que los convierte en herramientas más versátiles y que permiten realizar trabajos en zonas inaccesibles

o difícilmente accesibles por medios terrestres.

Todo esto provoca que los UAV sean una elección lógica a la hora de realizar múltiples aplicaciones, especialmente aquellas en las que la zona de trabajo sea extensa, el fenómeno que se va a estudiar sea dinámico o el desplazamiento por el medio aéreo aporte una ventaja y no se requiera el transporte de cargas demasiado elevadas. Existen una gran variedad de aplicaciones potencialmente realizables con UAV como la toma de datos, adquisición de imágenes, localización de objetivos, seguimiento, vigilancia... algunas de las cuales, como la fotogrametría o generación de mapas mediante la adquisición de imágenes aéreas ya cuentan con ejemplos de aplicación reales y empresas que las utilizan de forma habitual. Este potencial ha impulsado el importante desarrollo que se ha producido en los últimos años en las tecnologías para los UAV, lo que ha provocado un abaratamiento de los mismos, haciéndolos asequibles para las universidades y centros de investigación.

Una de las principales líneas de investigación con UAV es el comportamiento autónomo, es decir, el desarrollo de vehículos capaces de realizar misiones satisfactoriamente sin intervención humana alguna. Se trata de una meta extremadamente ambiciosa y para la que se debería trabajar esencialmente en tres aspectos: el primero es la percepción, ya que se requiere una consciencia de la situación y del entorno en el que se desarrolla la misión para lograr el comportamiento autónomo. El segundo es la toma de decisiones, una vez conocida la situación se debe disponer de la capacidad para analizar y decidir al respecto. Finalmente, la comunicación y cooperación, ya que los UAV deben ser capaces de intercambiar información tanto entre ellos como con otros vehículos con el fin de coordinarse y, en última instancia, colaborar en los posibles objetivos comunes.

En lo referente a los desarrollos en la percepción durante los últimos años destacan las mejoras en la sensorización que los UAV pueden llevar a bordo (sensores más precisos, con un tamaño, peso y consumo energético más reducido, mayores frecuencias de muestreo...) y de los algoritmos de tratamiento de esos datos. Entre ellos tienen una especial relevancia la mejora en las cámaras y algoritmos de visión artificial. El aumento de la calidad de las imágenes y las frecuencias de refresco permiten emplear la visión como forma de detectar y evitar obstáculos, incluso a altas velocidades [Barry, 2016], en otros se usan cámaras hiperespectrales, capaces de captar más allá del espectro visible, para tareas como la monitorización ambiental [Dunbabin and Marques, 2012], [Suomalainen et al., 2014] o la agricultura de precisión [Han, 2009], [Kooistra et al., 2013], [Honkavaara et al., 2013].

En la toma de decisiones también se ha avanzado, haciendo posible realizar el proceso de decisión en tiempo real en base a la información disponible. De

esta forma los UAV pueden ajustar sus acciones en función de las condiciones actuales de la misión que están realizando. En el caso de un único UAV, esto permite ajustar el control teniendo en cuenta las condiciones meteorológicas reales, siendo capaces de compensar fenómenos dinámicos como los vientos cruzados [Brezoescu et al., 2012]. En el caso de múltiples UAV, se trabaja activamente en conseguir una asignación de tareas rápida y cercana al óptimo [Bertuccelli et al., 2009] [Karaman et al., 2015] y se aplica lo mismo a las trayectorias a seguir [Oh et al., 2015] por los miembros del equipo.

El tercer aspecto, la comunicación y cooperación, tiene una gran importancia incluso en misiones con un único UAV debido a que ha de compartir el espacio aéreo con otros vehículos y, por tanto, debe existir una comunicación y coordinación con ellos para evitar colisiones. Sin embargo, este aspecto adquiere su mayor protagonismo en las misiones en las que se emplean múltiples UAV. El uso de múltiples UAV presenta diversas ventajas, tales como la mayor tolerancia a fallos al no depender únicamente de la sensorización y actuación de un único UAV; la aceleración de misiones con un área de trabajo extensa de terreno y la optimización de otras en las que interesa obtener información de varios puntos de forma simultánea. Misiones como la búsqueda y localización, vigilancia de un perímetro o área, toma de datos de fenómenos dinámicos... son ejemplos de misiones en las que lo óptimo es contar con un equipo de UAV que realice la tarea de forma simultánea. En algunas ocasiones no sólo resulta óptimo, sino necesario, contar con varios UAV para realizar una tarea, por ejemplo, para el movimiento de cargas que superan la capacidad de un único UAV [Maza et al., 2010].

Dentro de las misiones con múltiples UAV se pueden diferenciar dos tipos fundamentales en función de los UAV que forman parte del equipo y sus capacidades: Puede tratarse de equipos homogéneos, si todos los UAV presentan las mismas características físicas y la misma sensorización; o pueden ser heterogéneos, si las características físicas y/o la sensorización de cada uno de los miembros puede ser diferente. Dicha variedad permite la combinación de miembros con diversas capacidades, que podrían aportar mejores soluciones en tareas complejas y mejoras en cuanto a coste, disponibilidad, capacidad, etc.

Sin embargo, el uso de múltiples UAV acarrea también un aumento de complejidad, especialmente en el caso de los UAV heterogéneos, y una serie de retos comparado con el uso de un único UAV. Uno de los más evidentes es que los UAV han de coordinarse entre ellos para lograr una colaboración efectiva y evitar interacciones que afecten negativamente al funcionamiento del sistema tales como solapamientos o repeticiones en las tareas que se han de realizar, o incluso colisiones entre los UAV. Con esta finalidad, surgen las estrategias de coordinación, que se pueden definir como un conjunto de reglas a partir de las

que cada UAV decide en función de sus capacidades, su conocimiento propio y el de su entorno, la siguiente acción que debe realizar para cumplir la tarea encomendada. Debido a que las potenciales misiones con UAV son muy diversas (vigilancia, seguimiento, exploración, caracterización, actuación mecánica...) y el entorno en el que se desarrollan es dinámico y difícilmente predecible, la definición de esta serie de reglas resulta un reto muy complejo.

Además de los tres aspectos descritos, para supervisar y controlar el desarrollo de la misión resulta necesario disponer de un software de control de misión. Este software será el encargado de recopilar la información de todos los UAV tal y como la posición, orientación, información de la sensorización, posibles averías, etc. y de mostrarla de forma adecuada al operador. Idealmente este software debe actualizarse en tiempo real, para que la información visualizada sea lo más reciente posible y permitir una comunicación directa con los UAV, de forma que se puedan transmitir cambios en los objetivos de la misión, también en tiempo real, si es necesario.

Aunque es necesario trabajar en todos los aspectos mencionados anteriormente, el presente trabajo pondrá un mayor énfasis en la coordinación de los equipos de UAV heterogéneos, tratando de facilitar el desarrollo de nuevas estrategias de coordinación desde una visión práctica y funcional en la que se pretende trazar un camino que sirva para continuar el avance hacia la autonomía de los UAV.

Por lo general, existen dos formas de enfocar el desarrollo de una nueva estrategia de coordinación. Se puede partir de una aplicación concreta y generar una estrategia ad-hoc a partir de ella o, por el contrario, se puede tomar una estrategia de coordinación más general y adaptarla a la tarea concreta que se ha de realizar. Cada uno de estos enfoques tiene sus puntos fuertes y débiles. En el caso del enfoque basado en la tarea, el desarrollo de una estrategia ad-hoc suele conllevar un menor esfuerzo temporal y suele dar lugar a estrategias más sencillas, ya que la estrategia se plantea únicamente para un tipo de tarea y sus peculiaridades. Sin embargo, se trata de estrategias aisladas y restringidas a un problema, un entorno y un hardware concreto. Esta dependencia del hardware, el entorno y la aplicación concreta limita en gran medida su reutilización para trabajos posteriores.

Por el contrario en el caso del enfoque más general, el tiempo de desarrollo inicial suele ser mayor, ya que requiere adaptar la estrategia a las peculiaridades de la tarea. A cambio, el resultado suele ser una estrategia con menores dependencias y con un mayor potencial para ser reutilizada en posteriores aplicaciones comparada con la estrategia ad-hoc. El principal reto en este tipo de estrategias es ser capaces de alcanzar el grado de concreción necesario para que sean aplicables en la realidad con el menor esfuerzo posible.

En la revisión de la literatura se citan diversas aproximaciones de estrategias generales [Göktogan and Sukkarieh, 2009] [Selecký et al., 2015]. En algunos casos incluso se proponen metodologías y estructuras para los diversos niveles de control de los miembros del equipo, de forma que se maximice el reaprovechamiento de código. Dichas aproximaciones se describen de forma detallada en el capítulo 3.

Teniendo todo esto en cuenta, en este trabajo se propone una aproximación para la generación de estrategias de coordinación factibles y flexibles para equipos de UAV heterogéneos. Esta aproximación se basa principalmente en la combinación de una metodología sencilla y una plataforma de *realidad mixta*, que pretende ayudar al desarrollador de estrategias a lo largo de todo el proceso, incluyendo la implementación en la realidad. El objetivo es reducir la división existente entre la investigación teórica y las implementaciones reales con la intención última de favorecer la transferencia de resultados de la investigación a la realidad.

La metodología de desarrollo que se propone se basa fundamentalmente en tres conceptos: una estructura normalizada y modular del software, la independencia de la estrategia de coordinación del hardware concreto y la incorporación de forma iterativa y progresiva de la complejidad que acompaña a la realidad mediante el uso de la realidad mixta. Estos conceptos se detallan a continuación:

1. El primer concepto, la estructura normalizada y modular del software busca aumentar la facilidad tanto a la hora de depurar y corregir los posibles errores del código como a la hora de aumentar o modificar funcionalidades tales como sensores, actuadores, características del UAV o la electrónica a bordo. Se propone una estructura para el código con la intención de facilitar tanto el desarrollo de estrategias como la generación de modelos de los UAV y su instrumentación, además de favorecer el desarrollo de código de forma simultánea por varias personas al delimitar de forma clara fronteras entre los componentes del sistema. Esta estructura constituye la base sobre la que se apoyarán los otros dos conceptos.
2. El segundo concepto, la independencia de la estrategia de coordinación del hardware tiene como objetivo aumentar la flexibilidad de las estrategias generadas, de forma que se consigue una mayor reusabilidad en proyectos posteriores. Esto es especialmente interesante por la rapidez con la que evoluciona y cambia el hardware actualmente, debido a que el entorno de los UAV se encuentra en unas fases tempranas de desarrollo, y tiene asociada una carga importante de innovación. Si bien las capacidades y características generales del UAV deben considerarse en el desarrollo de la estrategia de coordinación, eso no implica que la misma deba estar es-

trechamente unida a un hardware concreto, ya que en ese caso, cualquier cambio en el mismo, ya sea por una variación física en los UAV o en la electrónica a bordo afectaría a la estrategia y esta debería ser modificada. Para conseguir dicha independencia en este trabajo se propone la generación de tres niveles en el código:

- El software de bajo nivel, que comprende tanto el firmware (la parte del código más cercana al hardware que se encarga de la lectura de la sensorización y la actuación), como el piloto automático o controlador de vuelo básico, que mantiene un nivel de vuelo controlado mediante la lectura de la sensorización a bordo y la actuación sobre las superficies de control y/o el motor del UAV.
 - La estrategia de coordinación y planificación de la misión, la parte del código de más alto nivel y que aporta la inteligencia al conjunto de UAV. Es la encargada de gestionar las acciones de cada uno de los UAV (movimiento, comunicación, actuación y sensorización) en cada momento de forma que se logre el éxito en las misiones mediante la gestión de los recursos disponibles por el equipo de UAV.
 - El middleware o código intermedio entre el software de bajo nivel y la estrategia de coordinación y que se encarga de ofrecer una interfaz común a esta última. De esta forma, el middleware permite controlar con los mismos comandos UAV diferentes y realizar cambios en el hardware sin afectar a la estrategia de coordinación, modificando únicamente el software de bajo nivel y/o el middleware.
3. Finalmente, la incorporación iterativa y progresiva de la realidad, constituye un método para tratar de reducir el *reality gap*, esto es, el problema que se produce al tratar de transferir resultados del simulador a la realidad. Este problema se produce debido a las diferencias provocadas por las simplificaciones y asunciones realizadas al modelizar la realidad en el simulador, por lo que se trata de un problema inevitable, al estar unido intrínsecamente al uso de simuladores. Sin embargo se puede minimizar su efecto mediante el modelado preciso de los UAV y los sensores y, sobre todo se puede facilitar su superación, suavizando el salto del simulador al mundo real.

Comúnmente el proceso que se sigue para la generación de una estrategia de coordinación comienza con una etapa de modelado, seguida de la generación de la estrategia, la prueba en un simulador, el ajuste de la estrategia y finalmente una serie de pruebas reales y corrección de los errores detectados. El principal problema con este proceso es que la prueba en la realidad de una estrategia de coordinación implica mantener en el aire a varios UAV cuyo controlador sólo

ha sido probado en un simulador y que, por tanto, se encuentra sometido al problema del reality gap, lo que puede llevar a resultados poco deseables. Para solucionar esto, se propone una aproximación más progresiva a las complejidades de la realidad mediante el uso de la realidad mixta: este concepto se basa en la combinación del mundo virtual y el mundo real, de forma que se generan nuevos espacios en los que los objetos virtuales y reales interactúan en tiempo real.

La realidad mixta permite el aporte de información real en el entorno simulado, por ejemplo, para mejorar y ajustar el entorno virtual y que se aproxime lo más posible al entorno real (condiciones meteorológicas, lecturas de sensores...). De la misma forma permite la generación de sensores virtuales, aumentando virtualmente la percepción del UAV real y ofreciendo así la posibilidad de realizar pruebas en cualquier momento y lugar usando modelos de fenómenos reales de los que se disponga de datos previamente. Este intercambio de información entre el entorno real y virtual permite trabajar con equipos con UAV reales y simulados de forma conjunta, simulando los sensores y el entorno en el que vuelan estos últimos a partir de los datos reales obtenidos por los primeros. Esto permite comprobar las estrategias de coordinación empleando un único UAV real y varios simulados e ir incrementando progresivamente el número de UAV reales, reduciendo así el riesgo asociado a la transición entre la simulación y la realidad. Además, permite que el entorno virtual se realimente con información real, mejorando así los modelos de los sensores virtuales y acercando el entorno virtual y real.

En este punto conviene aclarar la diferencia entre la estrategia de misión y la de coordinación. La estrategia de misión se refiere a la forma en la que se va a actuar para obtener el éxito en la misión, mientras que la estrategia de coordinación se refiere a las normas para conseguir establecer una cooperación entre los UAV. Cuando únicamente se trabaja con UAV reales, la estrategia de coordinación constituye la forma de resolver la misión real. Sin embargo, cuando existe una mezcla entre UAV reales y simulados, la estrategia de misión únicamente abarca la misión y los UAV reales, mientras la estrategia de coordinación trata de forma indistinta a todos los UAV, sean estos reales o simulados. Esto provoca que, en los casos en los que existen UAV simulados, la estrategia de coordinación constituye una forma de aportar a los UAV reales información proveniente de simulaciones y datos anteriores; además de una forma de comprobar el funcionamiento de la estrategia.

Las potenciales ventajas del uso de la metodología propuesta van desde un aumento de la reusabilidad del código, una mayor facilidad para la depuración o una mayor transferibilidad y seguridad, especialmente en las pruebas iniciales de la estrategia de coordinación, hasta una aceleración del proceso de generación e implementación de estrategias de coordinación más realistas, facilitando el

trabajo en paralelo de varias personas.

Por otro lado, esta metodología también acarrea una serie de requerimientos, entre los que se encuentra el desarrollo de software que permita aplicar sus principios. Se ha propuesto que las herramientas desarrolladas sean sencillas y flexibles, permitiendo su ampliación posterior con nuevas funcionalidades. La mayor parte del esfuerzo en el desarrollo software se centrará en el simulador de realidad mixta y todas las herramientas alrededor del mismo. Tras la revisión de la literatura, se ha realizado un diseño que permite la aplicación de la metodología propuesta y se ha procedido a desarrollarlo partiendo de un simulador de vuelo de código abierto, añadiendo al mismo las herramientas necesarias. La estructura que se ha seguido es la siguiente:

- Un simulador del entorno de los UAV, que se encarga de generar la información para los sensores virtuales, para lo que incluye modelos de los fenómenos que se van a estudiar. Dichos modelos permiten su ajuste mediante la realimentación de los datos de los sensores de los UAV reales, lo que aumenta su precisión, especialmente en el caso de fenómenos dinámicos o que se vean afectados por factores externos no controlables y difícilmente predecibles como pueden ser las condiciones meteorológicas (viento, humedad, temperatura, etc.).
- Un control de equipo, que constituye el punto central de información y en el que se implementan las herramientas de visualización de datos y supervisión de la misión; así como los comportamientos centralizados de la estrategia de coordinación, si fuese el caso.
- Un UAV virtual, que es la representación en el entorno virtual de un UAV y que puede ser indistintamente un UAV real o simulado. Incluye:
 - Los comportamientos distribuidos de la estrategia de coordinación que se va a implementar y tiene por tanto capacidad para decidir sobre la actuación del UAV.
 - La instrumentación virtual para los UAV simulados, incluyendo los sensores, la batería, las comunicaciones... Haciendo posible así la equivalencia entre UAV simulados y reales.
 - El comportamiento dinámico del vehículo, en el caso del UAV real, y una simulación del mismo basado en un modelo dinámico numérico en el caso de los UAV simulados.
 - El controlador de vuelo básico, encargado de mantener un nivel de vuelo controlado mediante la lectura de la sensorización y la actuación sobre las superficies de control y/o el motor del UAV. Compar-

tido por ambos tipos de UAV, en el caso de los reales se ejecuta en una unidad de procesamiento embarcada.

Para demostrar las ventajas del seguimiento de la metodología propuesta en el diseño e implementación de la plataforma, se recogen tres ejemplos, uno para cada aspecto de la misma. Estos ejemplos surgen de la necesidad de cambiar o adaptar partes de la plataforma para la realización de las diferentes aplicaciones que se han llevado a cabo en el tiempo de desarrollo de este trabajo. Además de poner de manifiesto las ventajas del uso de la metodología, son fruto de la experiencia obtenida en el uso real de la plataforma software propuesta.

En lo referente a la plataforma hardware, se ha realizado la selección de un modelo UAV comercial de ala fija teniendo en cuenta criterios de eficiencia en vuelo, así como de seguridad en caso de accidente o pérdida de control. Varias partes del mismo han sido rediseñadas para adaptarse al objetivo de este trabajo. Con la finalidad de hacerlo autónomo, se ha instrumentado el modelo con una combinación de electrónica comercial y otra diseñada y desarrollada a tal efecto durante el presente trabajo. Se ha incluido la sensorización necesaria para permitir al UAV conocer tanto su posición como su orientación en el espacio; un microcontrolador que contiene el controlador de vuelo básico, módulos de RF de largo alcance para la comunicación entre los miembros del equipo y un mini-pc con capacidad de computación suficiente para implementar estrategias de coordinación distribuidas complejas.

Finalmente, con la finalidad de demostrar la validez de la aproximación propuesta, se ha realizado un ejemplo de aplicación basado en una tarea de búsqueda y localización. La estrategia seleccionada es centralizada, basada en un algoritmo evolutivo Differential Evolution (DE) que se ha denominado Constrained Sampling-DE (CS-DE), desarrollado especialmente teniendo en cuenta las limitaciones y la problemática de los UAV.

Estructura del documento

Se detalla a continuación la estructura de las partes en las que se ha dividido el presente documento.,

- En el capítulo 2 se describe el objetivo global de esta tesis así como los hitos concretos que deben completarse para alcanzarlo.
- En el capítulo 3 se muestran los resultados de la revisión realizada sobre los principales trabajos desarrollados en el campo de la coordinación de UAV autónomos.

- En el capítulo 4 se presenta la metodología propuesta, detallando sus principios básicos y cómo afronta los principales retos presentes en el campo de la coordinación de UAV autónomos.
- En el capítulo 5 se define la estructura de la plataforma software desarrollada para la demostración de la metodología propuesta y se describen las partes que la forman.
- En el capítulo 6 se muestra una implementación concreta, con un ejemplo de aplicación y pruebas reales con hardware específico desarrollado para ellas.
- En el capítulo 7 se presentan las conclusiones derivadas del presente trabajo y se describen las posibles líneas de investigación y desarrollo planteables a partir de esta tesis doctoral.

Capítulo 2

Objetivos

El objetivo principal de esta tesis doctoral es proponer una aproximación que facilite el desarrollo e implementación de las estrategias de coordinación de múltiples UAV para la realización de tareas colaborativas. Esta aproximación consiste en una metodología que, unido a una plataforma de realidad mixta, permite agilizar, facilitar y hacer más seguro el proceso de desarrollo e implementación de nuevas estrategias para lograr que varios UAV colaboren para una tarea común. Este objetivo general se puede dividir en dos:

1. Desarrollar una metodología y unos principios de diseño para agilizar y facilitar el desarrollo y la implementación de las estrategias. Para ello, es necesario facilitar el trabajo de varias personas en paralelo, la mejora y ampliación de las funcionalidades, la reutilización del código generado, y la depuración de errores en el mismo. Para alcanzar esto se propone:
 - Generar una estructura normalizada y modular para el software, basada en bloques funcionales independientes.
 - Independizar el código del hardware concreto que ofrece las funcionalidades mediante un middleware.
 - Mantener el código de la estrategia de coordinación desde el simulador a la realidad.
2. Incorporar la seguridad como un aspecto más en el proceso de desarrollo e implementación de las estrategias de coordinación. Además, para facilitar la implementación se necesita un método con el que afrontar el reality gap, o diferencia entre el simulador y la realidad. En este caso se ha optado por el uso de la realidad mixta, para permitir la introducción progresiva e iterativa de la complejidad real. Para ello, se debe:

- Tener en cuenta la seguridad en todas las etapas del proceso, desde el diseño a la implementación en la realidad.
- Diseñar y generar un entorno de simulación de realidad mixta que permita el uso indistinto de UAV reales y simulados.
- Desarrollar herramientas de visualización de datos en tiempo real que permitan la supervisión y control de los UAV.

Finalmente, para demostrar la validez de la aproximación propuesta, se deberán llevar a cabo ejemplos de aplicación en los que se desarrollen e implementen estrategias de coordinación siguiendo los preceptos anteriores y en los que se realimenten y validen los modelos desarrollados con datos reales. Para ello es necesario:

- Diseñar y construir una plataforma de UAV reales con capacidad de vuelo autónomo, comunicación inalámbrica y sensorización adecuada para la realización de pruebas reales.

Capítulo 3

Antecedentes

3.1. Introducción

Los vehículos aéreos no tripulados o UAV se definen como vehículos sin tripulación, reutilizables, autopropulsados y capaces de mantener un nivel de vuelo controlado y sostenido. Cabe aclarar que existen diversas formas de referirse a este mismo concepto y se ha optado por esta debido a su brevedad, simplicidad y por tratarse de una de las más usadas en la publicaciones de investigación actualmente [Valavanis and Vachtsevanos, 2015].

Dentro de esta definición se incluyen vehículos aéreos de características muy diferentes; por lo que generalmente se suelen dividir los UAV según diversas características físicas comunes: el tamaño, la autonomía, la capacidad de carga, la forma de generar sustentación... Aunque no existe una clasificación universalmente aceptada, una de las más empleadas es la propuesta por la OTAN y que se reproduce en la Tabla 3.1.

En esta clasificación se agrupan los UAV en clases y categorías según su peso en orden de vuelo, es decir, según el peso del UAV justo antes de despegar para una misión incluyendo las baterías o combustible y toda la carga específica para la misión a realizar.

El uso de UAV presenta diversas ventajas frente a otras tecnologías que podrían emplearse para las mismas tareas. Si los comparamos con los vehículos aéreos tripulados tenemos un coste menor. A esto se suma el hecho de que el piloto, si existe, puede encontrarse en tierra y alejado de la zona de vuelo del UAV. Esto último minimiza el riesgo, especialmente cuando la misión se desarrolla en entornos difícilmente accesibles o peligrosos, como pueden ser ambientes de alta toxicidad, radiactivos, etc. En general los UAV son la alternativa elegida para

Class	Category	Normal employment	Normal Operating Altitude	Normal Mission Radius	Primary Supported Commander	Example platform
CLASS I (less than 150 kg)	SMALL >20 KG	Tactical Unit (employs launch system)	Up to 5K ft AGL	50 km (LOS)	BN/Regt, BG	Hermes 90 Luna
	MINI 2-20 kg	Tactical Sub-unit (manual launch)	Up to 3K ft AGL	25 km (LOS)	Coy/Sqn	Aladin DH3 DRAC Eagle Raven Scan Skylark Strix T-Hawk
	MICRO <2 kg	Tactical PI, Sect, Individual (single operator)	Up to 200 ft AGL	5 km (LOS)	PI, Sect	Black Widow
CLASS II (150 kg to 600 kg)	TACTICAL	Tactical Formation	Up to 10,000 ft AGL	200 km (LOS)	Bde Comd	Aerostar Hermes 450 iView 250 Ranger Sperwer
CLASS III (more than 600 kg)	Strike/ Combat	Strategic/National	Up to 65,000 ft	Unlimited (BLOS)	Theater COM	
	HALE	Strategic/National	Up to 65,000 ft	Unlimited (BLOS)	Theater COM	Global Hawk
	MALE	Operational/theater	Up to 45,000 ft MSL	Unlimited (BLOS)	JTF COM	Predator B Predator A Harfang Heron Heron TP Hermes 900

Table 1 - NATO UAS Classification Guide. September 2009 JCGUAV meeting

Tabla 3.1: Clasificación de UAV según la OTAN

las tareas monótonas, sucias o peligrosas («dull, dirty and dangerous») frente a la opción tripulada.

Actualmente se han producido grandes avances y una tremenda popularización de los UAV y han mostrado un gran potencial, sin embargo, el beneficio que pueden aportar realmente este tipo de sistemas será una realidad cuando sean completamente autónomos.

3.1.1. Historia y evolución de los UAV

Resulta difícil fijar una fecha o vehículo concreto para el primer UAV. Si bien puede considerarse que una de las primeras aproximaciones a lo que entendemos actualmente por UAV fueron los llamados torpedos aéreos que se desarrollaron a principios del siglo XX, poco después de los primeros vuelos de aviones autopropulsados. Estos vehículos, también conocidos como bombas voladoras, consistían en una aeronave pilotada remotamente que tenía como única misión llevar una carga explosiva hasta el objetivo. No eran reutilizables, por lo que actualmente no se calificarían como UAV, sin embargo constituyen uno de los primeros ejemplos de vehículos sin tripulación autopropulsados.



Figura 3.1: MQM-57 Falconer, uno de los primeros UAV para toma de datos

El desarrollo de los UAV continuó a lo largo de los años, aprovechando y a la vez impulsando mejoras de los instrumentos de medida de la posición y orientación y de los medios de transmisión inalámbricos. Estas mejoras han permitido a los UAV disponer de un mayor alcance y mejoras en la estimación de la posición y orientación, lo que se ha traducido en UAV más estables y controlados, capaces de volar más lejos. La mayor parte de las aplicaciones de esos primeros vehículos no tripulados tenían fines militares, como los OQ-2 Radioplane que se emplearon a partir de 1940, como objetivos para el entrenamiento de los soldados de las baterías antiaéreas.

Pocos años después comenzaron a aparecer UAV que se encargaban de misiones de reconocimiento, toma de datos y vigilancia. Concretamente en 1946, durante la operación «Crossroads», se emplearon varias aeronaves modificadas y pilotadas remotamente para la toma de muestras de partículas radioactivas después de la explosión.

En lo referente a los UAV de reconocimiento, uno de los primeros fue el MQM-57 Falconer, que puede verse en la Figura 3.1. Este UAV, que comenzó a volar en 1955, surgió como rediseño de un UAV para prácticas de tiro con un nuevo objetivo, la obtención de información (imágenes y/o vídeo) de zonas hostiles. El resultado fue un UAV que contaba con un tiempo de vuelo de menos de una hora y un alcance bastante limitado, lo que no le impidió abrir una nueva

línea de trabajo con los UAV.

El desarrollo de los UAV ha continuado hasta nuestros días, incorporando mejoras en la sensorización, la transmisión de información, la capacidad de procesamiento, la miniaturización de los componentes electrónicos, etc. Todas ellas han propiciado mejoras en el tiempo de vuelo, el alcance, los sistemas de control autónomo... Así, se pueden embarcar multitud de sensores, cámaras de vídeo de alta definición y transmisores que permiten recibir en tiempo real datos del UAV o controlarlos remotamente. El límite de alcance lo marca la tecnología de comunicación empleada, pudiendo llegar a todo el planeta si se emplean comunicaciones por satélite. Todo este desarrollo tecnológico, junto con el abaratamiento de los componentes han provocado la gran popularización y proliferación actual de los UAV, tanto para fines recreativos como para la investigación.

3.1.2. Los UAV autónomos

Un concepto que conviene aclarar antes de profundizar en mayor detalle en este documento es el de autonomía, algo importante para la comprensión completa de este trabajo y que será usado en numerosas ocasiones. En primer lugar, se refiere a la autonomía en la toma de decisiones y no a la distancia de vuelo del UAV. Dentro de este ámbito existen numerosas definiciones de la autonomía, una de las más apropiadas para este contexto es la del Dr. Lawrence J. Fogel, uno de los pioneros en la programación evolutiva, que definió la autonomía como «la habilidad de generar los propios objetivos sin recibir ninguna instrucción del exterior».

Otro concepto importante es el de inteligencia, que puede definirse como la capacidad de adquirir conocimiento y usarlo para un fin. Frecuentemente se confunde con la autonomía y si bien están relacionados, se tiende a considerar que un sistema que posee mayor inteligencia, posee automáticamente una mayor autonomía. Sin embargo, un sistema únicamente necesita la inteligencia suficiente para realizar la tarea para la que fue concebido, cualquier aportación extra de inteligencia no constituirá una mejora de su autonomía.

En [Clough, 2002] se recoge una clasificación de los UAV según su autonomía en 10 niveles que puede ayudar a clarificar esta distinción.

Para medir la autonomía se tienen en cuenta tres aspectos:

- La percepción, ya que se requiere una consciencia de la situación y del entorno en el que se desarrolla la misión para lograr la autonomía.
- La toma de decisiones, debido a que es necesario disponer de la capacidad para analizar y decidir al respecto.

- La comunicación y cooperación, los UAV deben ser capaces de intercambiar información tanto entre ellos como con otros vehículos con el fin de coordinarse y, en última instancia, colaborar en los objetivos comunes.

Así, el nivel más bajo de autonomía le corresponde a un vehículo pilotado remotamente y que requiere la atención constante de un operador. Por contra, el nivel superior se refiere a un vehículo con capacidades de percepción, toma de decisiones y cooperación similar a la humana. Esto le permitiría realizar una misión completa, incluyendo el despegue y aterrizaje sin intervención humana alguna.

La autonomía constituye la forma de descargar a los operadores de los UAV de las tareas más tediosas y repetitivas. Esto permite que un único operador pueda controlar varios UAV y que se centre en supervisar el correcto desarrollo de la aplicación, aumentando así la seguridad de los vuelos.

Actualmente los UAV suelen combinar control remoto y autónomo. Por ejemplo, los sistemas de control y orientación incorporan capacidad para realizar tareas de vuelo, tales como control de velocidad, de estabilización e incluso de la trayectoria de vuelo hasta un punto de referencia. Este tipo de ayudas resultan especialmente importantes cuando se producen perturbaciones tales como corrientes térmicas, rachas de viento, etc. Sin embargo, las tareas más complejas, como el despegue y el aterrizaje, suelen realizarse de forma manual controlados por un operador.

La autonomía de los UAV constituye una línea de investigación que todavía se encuentra lejos de su madurez y que presenta diversos retos entre los que destacan:

- La fusión de sensores, la combinación de información proveniente de diferentes sensores para usar a bordo del vehículo.
- La gestión de comunicaciones y coordinación entre múltiples agentes en presencia de información incompleta e imperfecta.
- Planificación de trayectorias, el cálculo de la trayectoria óptima para el vehículo que cumpla los objetivos de la misión con las restricciones correspondientes como obstáculos o requisitos de combustible.
- Control de trayectorias, las estrategias de control para mantener a un vehículo con una tolerancia siguiendo una determinada trayectoria.
- Asignación de tareas y programación, la determinación de la distribución óptima de las tareas entre un grupo de agentes, con las limitaciones de tiempo y equipamiento.

- Cooperación, definir una secuencia de las actividades entre los agentes y una distribución espacial óptimas, con el fin de maximizar la probabilidad de éxito en cualquier escenario.

El presente trabajo se centrará especialmente en la comunicación y coordinación, con la finalidad de permitir así una mayor cooperación entre los UAV que formen el equipo.

3.2. Equipos de UAV

En comparación con los vehículos autónomos que realizan misiones en solitario, se puede obtener una mayor eficiencia y capacidad operativa mediante equipos de vehículos autónomos que operan de forma coordinada. Es fácil entender que un grupo de vehículos aéreos no tripulados es más capaz que un único UAV, ya que la carga de trabajo se puede dividir entre los componentes del equipo.

3.2.1. Tipos de equipos de UAV

Atendiendo a la composición de los mismos existen dos clases principales de equipos de UAV, los homogéneos y los heterogéneos:

En el primer caso los UAV que forman el equipo son prácticamente iguales, presentando características muy similares tanto en lo referente al propio UAV como a la instrumentación y electrónica embarcada. Esto se traduce en algunas ventajas, como que únicamente es necesario desarrollar un modelo para realizar simulaciones y un único controlador de vuelo, si sumamos a esto que la sensorización de todos los UAV es la misma tenemos que la respuesta de los UAV será prácticamente idéntica, lo que simplifica el desarrollo de estrategias de coordinación. Por otro lado, esta clase de equipos presenta también varias limitaciones importantes; Al tener que seleccionar un único UAV físico podemos vernos limitados para introducir sensores voluminosos, si seleccionamos un UAV pequeño o para mantener la maniobrabilidad en espacios limitados si optamos por un UAV mayor. Además, la selección de una misma sensorización para todos los miembros del equipo puede llevar a cargar instrumentación innecesaria y/o infrutilizada para tratar de aumentar la versatilidad del equipo.

Por el contrario, en el caso de los equipos heterogéneos existen diferencias que pueden afectar al UAV en si mismo o a la electrónica y sensorización embarcada. En el caso de que se empleen UAV físicamente diferentes, se complica el desarrollo de los modelos de simulación y los controladores de vuelos correspondientes, ya que la respuesta de cada uno a unas consignas será diferente. En

el caso de que se varíe la sensorización, la estrategia de coordinación ha de tener presente qué capacidades posee cada uno de los UAV y actuar en consecuencia. A cambio de este aumento en la complejidad, los equipos heterogéneos ofrecen una versatilidad mucho mayor permitiendo la optimización de las características físicas y de la instrumentación del UAV para la tarea a realizar, lo que en aplicaciones complejas puede suponer una ventaja significativa.

Teniendo esto en cuenta se deberá llegar a una solución de compromiso entre la facilidad de desarrollo de estrategias y la versatilidad del equipo de UAV según los recursos disponibles y los requisitos de la aplicación o aplicaciones que pretendan realizarse.

3.2.2. Aplicaciones potenciales para equipos de UAV

Existen multitud de aplicaciones especialmente apropiadas para los UAV, ya sea porque el fenómeno a estudiar se produce en el medio aéreo, el área de trabajo es extensa y/o existen dificultades de acceso por medios terrestres. Algunas de ellas, ya sea por su complejidad, la amplitud del espacio de trabajo o la necesidad de tomar datos y/o actuar en más de un punto simultáneamente requieren más de un UAV. En general, para la mayoría de aplicaciones el uso de varios UAV puede aportar una mayor rapidez en la resolución de la tarea y/o una mayor resistencia frente a fallos o averías.

Este beneficio puede ser especialmente importante en las aplicaciones que estudian fenómenos dinámicos, cuyo tiempo de aparición o desarrollo es corto. Un ejemplo de este caso es el de fenómenos meteorológicos rápidos, como las tormentas locales severas, que tienen escalas de tiempo del orden de minutos u horas. El estudio y observación de estos fenómenos constituye un desafío debido a su intensidad, su pequeña área de influencia y su tendencia al desarrollo rápido. Las condiciones meteorológicas asociadas a estas tormentas, que frecuentemente generan tornados, hacen inviable el uso de aviones tripulados tradicionales para la toma de datos. Por ello, en [Frew et al., 2012] se propone el empleo de un UAV, bautizado como «Tempest», para la toma de datos *in situ* de los vientos, la temperatura y diversas características en la región posterior del flanco de las supercélulas de la tormenta. En este caso, además del beneficio de obtener varias mediciones en puntos distantes para un mismo instante de tiempo existen posibilidades de que alguno de los UAV falle, dadas las duras condiciones para el vuelo, por lo que contar con varios UAV aumenta las posibilidades de una consecución exitosa de la misión.

Otras aplicaciones se caracterizan por poseer un espacio de trabajo muy amplio, por lo que la tarea se acelera al aumentar el número de UAV y repartir así este espacio entre ellos. Es el caso de la lucha contra los incendios forestales,

un problema de especial importancia en España y que provoca que cada año la pérdida de decenas de hectáreas de bosques junto con la desaparición de gran parte de la vida salvaje de los mismos. En algunos casos, como en el año 2012, esta cifra se eleva a los cientos de miles de hectáreas de bosques quemados. Los UAV pueden ayudar en este caso a la detección rápida y la localización precisa de los conatos de incendio, vigilando las áreas más propensas a los mismos. Además, tal y como se propone en [Merino et al., 2010] mediante el uso de varios UAV equipados con cámaras de infrarrojos es posible monitorizar el avance de las llamas, ofreciendo así información mucho más precisa del desarrollo del incendio que la tradicional estimación visual.

El mismo problema de la amplitud del área de trabajo se presenta en el caso de otras aplicaciones de monitorización ambiental y de los cultivos como la detección de plagas. En el caso concreto de [Tahir and Brooker, 2009] se propone el empleo de un sensor óptico que permita realizar el rastreo de las plagas de langostas. Se trata de un problema que se produce frecuentemente en Australia y que acarrea cuantiosas pérdidas, especialmente a los agricultores de las zonas por las que se desplazan. Por otra parte, también se está trabajando de una forma muy activa en este campo con cámaras hiperespectrales. Estas cámaras captan ondas electromagnéticas más allá del espectro visible, ofreciendo mayor información. Esta información parece especialmente prometedora en la agricultura de precisión, que se refiere a una gestión de las parcelas agrícolas en la que se da mucha importancia a la observación, medida y actuación frente a la variabilidad entre cultivos con la finalidad de optimizar la producción y reducir el uso de abonos y pesticidas. El uso de cámaras de este tipo se propone tanto en [Kooistra et al., 2013], como en [Honkavaara et al., 2013] y en el trabajo que se está realizando en el GII conjuntamente con otros grupos de investigación, en el proyecto SIMMAP (Sistema Integral de Monitorización Multiresolución para la Agricultura de Precisión) del Ministerio de Economía y Competitividad.

Existen sin embargo, otras aplicaciones en las que el principal reto consiste en la coordinación e interacción de vehículos autónomos con las personas y otros robots, para acelerar tareas en las que el tiempo es un factor crítico o realizar labores peligrosas minimizando el riesgo. Es el caso de las misiones de búsqueda y rescate en catástrofes naturales o en accidentes. En [Kruijff et al., 2012] se propone como solución óptima la colaboración entre equipos humanos y robots inteligentes, tanto UAV como UGV y se analizan las problemáticas que ello conlleva y en [Maza et al., 2011] se propone una arquitectura de decisión que permita la coordinación de equipos de varios UAV en la gestión de desastres.

3.2.3. Retos para el control cooperativo de múltiples UAV

Todas las aplicaciones del apartado anterior tienen en común un gran potencial de mejora si se realizan con un equipo de UAV coordinados. Para convertir este potencial en una realidad es necesario desarrollar diferentes aspectos del control cooperativo.

Existen cuatro aspectos básicos a tener en cuenta a la hora de realizar un control cooperativo: el solapamiento de tareas, la incertidumbre, la comunicación imperfecta y la implementabilidad. Generalmente se recurre a una descomposición jerárquica para dividir el problema en partes, lo que nos lleva a soluciones no óptimas al ignorar ciertos grados de acoplamiento para conseguir dicha descomposición. En muchos casos, esto es necesario para ser capaces de resolver el problema y para que las estrategias sean más robustas. Esta robustez es especialmente necesaria en el caso de misiones en entornos dinámicos y no predecibles.

Solapamiento de tareas Algunas misiones tales como el seguimiento de un objetivo o el reparto de un área para realizar una búsqueda o de un perímetro para realizar una vigilancia tienen múltiples tareas solapadas asociadas con restricciones temporales. Existen numerosos problemas asociados a resolver la asignación de este tipo de tareas.

Al existir una conexión entre una tarea y las siguientes y ser esta dependiente del tiempo el problema no es separable, ya que las acciones de uno de los UAV afectan al resto y que además dicha relación varía o puede variar con respecto al tiempo, por lo que no se puede establecer *a priori*.

Uno de los principales problemas es la separación entre la asignación y el cálculo de trayectorias en una tarea. Si se busca el óptimo de la colaboración se ha de tener en cuenta la trayectoria que debe recorrer cada uno de los UAV para realizar una tarea concreta, y optimizarla seleccionando el UAV con la trayectoria más corta o rápida.

Otro problema son las restricciones temporales, ya que existen tareas que se han de realizar en un determinado momento y, en general, las tareas de cada UAV afectan al resto de miembros del equipo. Por ejemplo, en una misión en la que se trate de maximizar el área vigilada por un equipo de UAV, la trayectoria óptima para un UAV concreto dependerá de las trayectorias del resto de los UAV del equipo.

Incertidumbre Por el contrario, en otras aplicaciones el problema de la incertidumbre puede tener mucha más importancia que el solapamiento de las tareas. Es el caso de aplicaciones de búsqueda o localización en las que se desconoce el

número de objetivos y, en general, aquellas en que existe una información limitada del entorno, los objetivos o la duración de la misión. El reto en este tipo de aplicaciones consiste en estimar el valor futuro de las acciones emprendidas en el momento; para ello se puede recurrir a distribuciones probabilísticas, datos experimentales anteriores, etc.

En general, se trata de maximizar la utilidad de los recursos disponibles en los UAV, teniendo en cuenta su potencial de utilización a lo largo de toda la misión. Todo esto cobra una especial importancia cuando el recurso resulta central en el desarrollo de la misión, por ej. el consumo de batería en muchas misiones con UAV.

Aunque el entorno suele ser la principal fuente de incertidumbre, existen así mismo otros aspectos que pueden aumentar la misma dentro del mismo UAV tales como averías, fallos de sensores, etc que conviene tener en cuenta al desarrollar la estrategia de coordinación.

Comunicación imperfecta Una de las simplificaciones que se suele realizar normalmente es que las comunicaciones se producen de forma instantánea y los UAV pueden intercambiar tanta información entre ellos como necesiten. Sin embargo los enlaces de comunicación reales cuentan con un ancho de banda limitado, se producen retrasos entre el envío y la recepción de los mensajes y hay un porcentaje de los mismos que nunca alcanza su objetivo, lo que lleva a la presencia de información desfasada, limitada o parcial.

Una elección muy importante es si las decisiones del equipo se toman de forma síncrona o asíncrona. La elección del modo síncrono implica que existe una base de datos común y actualizada accesible para todos los vehículos. Si ocurre un evento localmente, dicho evento y la información asociada se comparte a través de la red y la toma de decisiones basada en dicho evento no puede ocurrir hasta que esto suceda. Siguiendo estrictamente esta máxima, un único UAV puede ralentizar al equipo completo y comprometer las tareas que requieran una respuesta rápida, por lo que este tipo de aproximación necesita estrategias que mantengan la coherencia y el rendimiento del equipo de UAV.

Por su parte, los protocolos de decisión asíncronos son mucho más robustos a los retrasos en las comunicaciones, sin embargo son susceptibles de que exista información inconsistente a lo largo de la red. Esto puede llevar a tomar decisiones inapropiadas, como la repetición de una tarea ya realizada por otro UAV. Para evitar esta clase de problemas se puede optar por la presencia de un sistema de consenso entre los UAV. En todo caso, si existe falsa información y retraso suficiente en la transmisión de la misma se pueden dar casos en los que no se alcance el consenso. Idealmente, debe haber información suficiente para asegurar que todas las tareas a realizar están cubiertas y las tareas son consis-

tentes. Además, los vehículos pueden tener modelos de estimación y predicción del estado para proporcionar información no disponible localmente.

Implementabilidad Los problemas de control cooperativo son complejos debido a las características del problema que se van a resolver, esto es, el número de vehículos, el número de objetivos, y el acoplamiento entre las acciones de cada uno de los UAV, lo que lleva a la no separabilidad del problema de coordinación. Mientras en algunos sistemas existe la posibilidad de optimizar el comportamiento de cada agente por separado, dividiendo el problema general en problemas más sencillos, en el caso de los UAV cada acción realizada afecta al resto. Además, en la mayoría de los casos el entorno es dinámico y cambiante, lo que dificulta el modelado del mismo.

A lo largo de los años se ha ido mejorando la capacidad computacional disponible en todos los equipos electrónicos y se ha producido una disminución del consumo y el volumen de los mismos. Dichos avances permiten embarcar procesadores más potentes en los UAV, facilitando la incorporación de estrategias de coordinación de mayor complejidad. De todas formas en la mayor parte de los casos resulta inabordable la formulación exacta del problema por lo que se suele recurrir a simplificaciones que permitan obtener una solución cercana al óptimo. Además el reparto de esta carga computacional dependerá del tipo de estrategia a emplear, si se trata de una estrategia centralizada, la mayor parte de la misma recaerá sobre el nodo central y si, por el contrario, se opta por una solución distribuida, esta carga estará repartida entre los diversos UAV del equipo.

Uno de los principales causantes de la necesidad de potencia computacional es la ejecución de la estrategia en tiempo real, lo que constituye un aspecto muy importante de la misma, ya que le permite adaptarse a los cambios en el entorno y emprender acciones teniendo en cuenta la información más reciente disponible en cada momento.

3.3. Estrategias de coordinación de equipos de UAV

Una vez presentados los principales retos en la coordinación de UAV y con la finalidad de estructurar de forma correcta una plataforma para el desarrollo de estrategias de coordinación resulta necesario realizar una breve revisión de las mismas y analizar las principales características que presentan. El conocimiento del funcionamiento de dichas estrategias permitirá además seleccionar estrategias representativas para realizar la validación de la plataforma.

En primer lugar conviene aclarar que existen dos tipos de estrategias principales, las centralizadas y las distribuidas. En el caso de las estrategias centralizadas existe un nodo central encargado de tomar las decisiones para todos los miembros del equipo. Este nodo central asume todo el esfuerzo computacional liberando al resto, que pueden ser más sencillos ya que únicamente han de enviar al nodo central su información de estado y ejecutar las tareas provenientes de este. Sin embargo, dado que todo el funcionamiento del sistema depende del nodo central cualquier fallo en este inutiliza el sistema completo. Este tipo de aproximación requiere de una vía de comunicación de todos los UAV con el nodo central, ya sea de forma directa o mediante una red mallada.

Por el contrario, en las estrategias distribuidas no existe la figura del nodo central y la toma de decisiones se realiza en cada UAV. Para ello los UAV se intercambian información entre ellos y cada uno decide la siguiente acción a emprender. Esto se traduce en un requisito de mayor potencia computacional en los nodos y una mayor carga en la red de comunicaciones. Además, resulta bastante conveniente desarrollar un algoritmo de consenso para evitar solapamientos en el reparto de tareas. A cambio estas estrategias ofrecen una flexibilidad y resistencia a fallos mayor, ya que el sistema continuará funcionando aunque se produzca la caída de cualquier nodo.

La estrategia de coordinación óptima dependerá en gran medida de la tarea o tareas a realizar con los UAV. Por ejemplo, los requerimientos para lograr realizar vuelos en formación cerrada son muy diferentes a los de una misión de supervisión o vigilancia de un área, ya que los objetivos que se persiguen son diferentes. Probablemente en el primer caso se tratará de optimizar la exactitud en el cálculo de las trayectorias a seguir de forma lo más rápida posible mientras que en el segundo caso se tratará de maximizar el tiempo de vuelo y la distancia recorrida.

Para tratar de seleccionar las más adecuadas se puede partir de que las estrategias de coordinación para UAV constituyen un caso concreto dentro de las estrategias multirobot y tienen como principal característica diferenciadora con los mismos el medio físico en el que se desplazan los UAV. Así muchas de las estrategias desarrolladas en los últimos años en esta disciplina podrían también aplicarse a la coordinación de un equipo de UAV. Este mismo enfoque es el que se propone en [Scerri et al., 2008], en el que se realiza la transición a la aplicación real de un algoritmo multiagente desarrollado en simulación, demostrando que existe la posibilidad de realizar esta adaptación y describiendo las dificultades y problemas encontrados en el transcurso del proceso.

Al igual que en el caso de las estrategias multirobot en la gran mayoría de las misiones la solución óptima al problema de la estrategia de coordinación resulta inabordable, tanto desde el punto de vista de su formulación como desde el punto

de vista del uso de tiempo computacional para su resolución en tiempo real. Por tanto, generalmente se suele recurrir a heurísticos. Un heurístico se puede definir como: «Un procedimiento simple, a menudo basado en el sentido común, que se supone que ofrecerá una buena solución (aunque no necesariamente la óptima) a problemas difíciles, de un modo fácil y rápido» [Zanakis and Evans, 1981].

Es decir, se usa el término heurístico para referirse a un procedimiento que trata de aportar soluciones a un problema con un buen rendimiento, tanto en lo referente a la calidad de las soluciones como a los recursos empleados. Algunos autores [Silver et al., 1980] proponen la siguiente clasificación de métodos de resolución mediante heurísticos:

- Métodos constructivos, que se caracterizan por construir una solución definiendo diferentes partes de ella en sucesivos pasos.
- Métodos de descomposición, en los que se divide el problema en varios más pequeños y la solución se obtiene a partir de las soluciones de cada uno de estos.
- Métodos de reducción, que tratan de identificar alguna característica de la solución que permita simplificar el tratamiento del problema.
- Métodos de manipulación del modelo, que obtienen una solución del problema original a partir de otra de otro problema simplificado (con menos restricciones, linealizando el problema, etc.)
- Métodos de búsqueda por entornos, en las que se parte de una solución inicial a la que se realizan modificaciones en sucesivas iteraciones para obtener una solución final. En cada iteración existe un conjunto de soluciones vecinas candidatas a ser nueva solución en el proceso.

3.3.1. Técnicas metaheurísticas

Las técnicas metaheurísticas se encuadran en los heurísticos de búsqueda por entornos y el término fue empleado por primera vez por Fred Glover [Glover, 1986].

Las técnicas metaheurísticas se diferencian de otros métodos heurísticos en que tratan de huir de óptimos locales orientando la búsqueda en cada momento dependiendo de la evolución del proceso de búsqueda. Concretamente, la lógica que siguen es partir en primer lugar de una solución (o conjunto de soluciones) que típicamente no es óptima. A partir de ella se obtienen otras parecidas, de entre las cuales se elige una que satisface algún criterio. A partir de esta última comienza de nuevo el proceso, hasta que se detiene cuando se cumple alguna condición establecida previamente.

En general, las técnicas metaheurísticas tienen las siguientes características:

- Son ciegas, es decir, no saben si llegan a la solución óptima y, por lo tanto, se les debe indicar cuándo deben detenerse.
- Son algoritmos aproximativos, por lo que no garantizan la obtención de la solución óptima.
- Aceptan ocasionalmente malos movimientos; es decir, cada nueva solución no es necesariamente mejor, en términos de la función objetivo, que la inmediatamente anterior. Algunas veces aceptan incluso soluciones no factibles como paso intermedio para acceder a nuevas regiones no exploradas.
- Son relativamente sencillas; todo lo que se necesita es una representación adecuada del espacio de soluciones, una solución inicial (o un conjunto de ellas) y un mecanismo para explorar el campo de soluciones.
- Son generales, prácticamente se pueden aplicar en la resolución de cualquier problema de optimización. Sin embargo, la definición de la técnica será más o menos eficiente en la medida en que las operaciones tengan relación con el problema considerado.
- La regla de selección depende del instante del proceso y de la historia hasta ese momento. Si en dos iteraciones determinadas la solución es la misma, la nueva solución de la siguiente iteración no tiene por qué ser necesariamente la misma.

Una metaheurística tendrá éxito en un problema de optimización dado si puede proporcionar un equilibrio entre la explotación de la experiencia de búsqueda acumulada y la exploración del espacio de búsqueda para identificar zonas con soluciones de alta calidad, cercanas al óptimo. La principal diferencia existente entre las metaheurísticas reside en la forma particular en la que tratan de conseguir este equilibrio. Hay una amplia variedad de metaheurísticas y se pueden clasificar atendiendo a diferentes aspectos referentes a su funcionamiento, tal y como puede verse en la Figura 3.2:

Búsqueda global o local: Uno de los métodos consiste en clasificar según el tipo de estrategia de búsqueda. Muchas ideas de metaheurísticas se introdujeron para mejorar heurísticas de búsqueda local, con el fin de encontrar mejores soluciones. Las metaheurísticas de búsqueda global por el contrario no se basan en búsqueda local y suelen ser poblacionales.

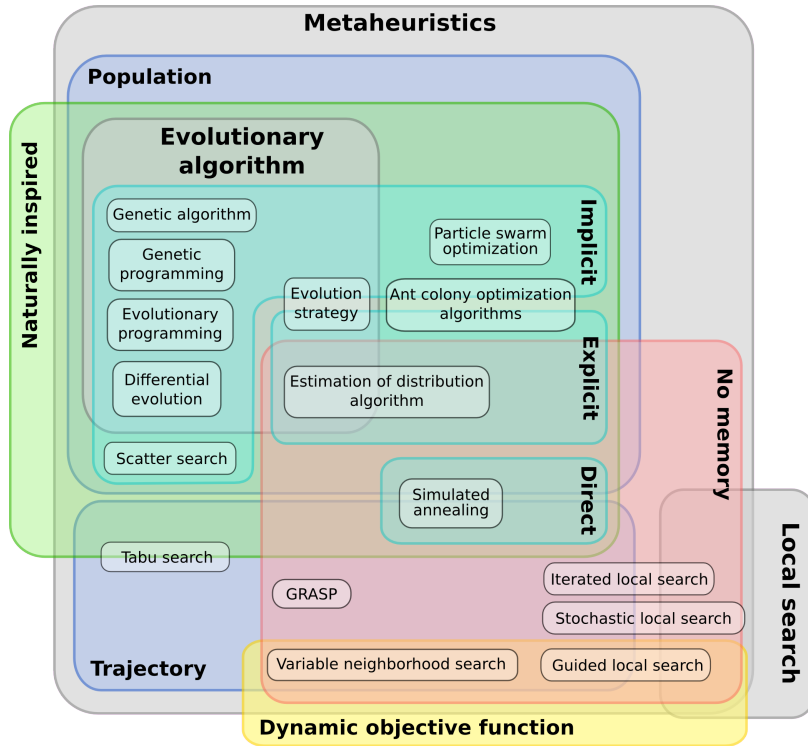


Figura 3.2: Clasificación de las metaheurísticas

Métodos de trayectoria o discontinuos: Otra distinción importante entre las diferentes metaheurísticas es la trayectoria de búsqueda, si se corresponde a un camino cerrado o si se producen saltos en el gráfico de vecindad. En aquellas en las que se usan métodos de trayectoria se permite habitualmente moverse a soluciones peores con el fin de escapar de máximos locales.

Por el contrario, en otros se generan los puntos iniciales y se realiza posteriormente una búsqueda local. La generación de soluciones iniciales corresponde a saltos en el espacio de búsqueda, por lo que estos algoritmos siguen en general un recorrido discontinuo con respecto al gráfico de vecindad usado en la búsqueda local.

Poblacionales o basados en solución única: Otra dimensión para la clasificación de las metaheurísticas se basa en el número de soluciones candidatas; en el caso de las aproximaciones de solución única, las metaheurísticas se basan en modificar y mejorar una única solución candidata.

A diferencia de estos, las aproximaciones poblacionales mantienen y mejoran múltiples soluciones candidatas, frecuentemente empleando características

de la población para guiar la búsqueda. El uso de un algoritmo poblacional proporciona una forma conveniente de exploración del espacio de búsqueda, aunque el rendimiento de la solución final depende en gran medida en la forma en que se manipule la población

Empleo de memoria: Otra posible forma de caracterizar las metaheurísticas es según su uso de la experiencia de búsqueda, o memoria, para influenciar las futuras direcciones de búsqueda. Generalmente esto supone una ventaja y actualmente la mayoría de metaheurísticas empleadas presenta memoria.

Hibridación y algoritmos meméticos: Una metaheurística híbrida es aquella en la que se combina una metaheurística con otras aproximaciones de optimización, como pueden ser algoritmos de programación matemática, programación con restricciones y aprendizaje automático. Ambos componentes de una metaheurística híbrida pueden ejecutarse de forma concurrente e intercambiar información para guiar la búsqueda.

Por otro lado, los algoritmos meméticos representan la sinergia de los algoritmos evolutivos o cualquier otro algoritmo poblacional con aprendizaje individual independiente o procedimientos de mejora locales para la búsqueda de soluciones.

Bioinspirado o no bioinspirado: Otro aspecto a tener en cuenta para clasificar las metaheurísticas es la fuente original de inspiración. Mientras algunas se basan en consideraciones de eficiencia en la solución de problemas combinatorios otras están inspiradas por fenómenos naturales. La intención es tratar de obtener ventaja de un fenómeno que se produzca en la naturaleza, aplicándolo a la solución de problemas de optimización.

3.3.2. Algoritmos para la coordinación de UAV

Una vez explicadas las características más importantes de los heurísticos se realizará una breve recopilación de los más empleados en la coordinación de UAV. En general no se suelen abordar todos los retos enumerados en el apartado 3.2.3 en todas las aproximaciones. De hecho lo más común es que cada publicación se centre en uno o dos aspectos del problema:

- En lo que se refiere a la asignación de tareas, dada la imposibilidad de resolver de forma exacta el problema en el tiempo disponible se suele optar por aproximar el coste para cada uno de los UAV de realizar una tarea y se escoge aquel que presente el coste menor mediante *algoritmos*

de subasta [Bertucelli et al., 2009] o mediante heurísticos, tales como *algoritmos genéticos* [Choi et al., 2011].

- Por su parte, para minimizar el problema generado por la incertidumbre se pueden emplear aproximaciones como la *programación dinámica estocástica* [Giard et al., 2007], los *procesos de decisión de Markov*, *redes Bayesianas* y en caso de que no se disponga de información, *teoría de juegos*.
- Con respecto a las comunicaciones y el problema de la información parcial se suele recurrir a algoritmos de consenso. En el caso de estrategias distribuidas se suelen emplear métodos como la *votación de consenso* [Saber and Murray, 2003] o *job shop scheduling* [Liu and Sycara, 1996].
- Finalmente en lo que se refiere a la implementabilidad, en la mayoría de publicaciones en las que se incluyen vuelos reales se describe el hardware empleado y en caso contrario se suele describir el equipo sobre el que se ejecutan las simulaciones. De esta forma se garantiza que se pueden ejecutar en tiempo real siempre que se cuente al menos con esa capacidad de cómputo.

En general se ha observado una abundancia de soluciones mediante metaheurísticas, especialmente de metaheurísticas bioinspiradas tales como las basadas en *colonias de hormigas* y sus variantes, [Cekmez et al., 2014] o [Freitas et al., 2010] *enjambres de partículas* [Ho et al., 2013] o algoritmos evolutivos como la *Evolución Diferencial* [Zhang and Duan, 2015] o [Varela et al., 2014b]. Sin embargo, no existe un enfoque capaz de resolver todas las situaciones, por lo que se requiere el desarrollo de métodos heurísticos y esquemas iterativos cuyo objetivo sea solucionar los retos más importantes y las características específicas de cada escenario operacional en particular.

3.4. Plataformas para el desarrollo de estrategias de coordinación

A lo largo de los años se han realizado diversas propuestas que tratan de afrontar el problema de la coordinación de los UAV. A continuación se resumen los ejemplos más significativos de esfuerzos por facilitar el desarrollo y la implementación de estrategias de coordinación con UAV, encontrados durante la revisión bibliográfica realizada para este trabajo.

3.4.1. C3UV

El Centro para el Control de Colaboración de vehículos no tripulados (C3UV) de la Universidad de California desarrolló su propia plataforma de UAV autónomos como un medio para validar una serie de cuestiones fundamentales relacionadas con el funcionamiento autónomo y la colaboración de equipos heterogéneos, tales como la detección y el control, planificación de trayectorias, obstáculo y prevención de colisiones, las comunicaciones, la descomposición de tareas y la asignación, ejecución de cooperación e intercambio de información .

Su enfoque para la sensorización colaborativa se basa en la premisa de que la adquisición y el control de la información están estrechamente unidos. El sentido común indica que la estimación constituye un medio para lograr el control. Sin embargo ellos consideran en el control como un medio para lograr una estimación más eficaz y eficiente de las características del mundo. Un ejemplo de este mismo razonamiento se puede encontrar también en [Jackson et al., 2008], donde un algoritmo de seguimiento de ruta se diseña para mejorar la calidad de imagen mediante el control del tamaño real de la imagen capturada por la cámara.

A medida que las capacidades del sistema se incrementaron, generaron también una arquitectura de software estructurado, con interfaces y abstracciones que permitiesen acelerar el desarrollo del software para el control, los comportamientos interactivos y la sensorización. Esto condujo finalmente al diseño del lenguaje CSL («Colaborative Sensing Language») y DSL («Domain Specific Language») para la implementación de controles de nivel de misión que mantengan el reflejo necesario con los componentes físicos mientras se desacopla con el sistema subyacente para optimizar la latencia de la red y la escalabilidad. CSL se proporciona como un lenguaje gráfico, usando una interfaz gráfica de usuario y como una interfaz de programación de aplicaciones implementada utilizando servicios web.

La arquitectura de estimación y control se divide en cuatro capas diferentes y una plataforma física sobre la que se implementan:

- La primera, denominada capa 0 se encarga del control de bajo nivel de los vehículos, es decir, ir de un punto de GPS a otro o girar con una inclinación constante. Además se encarga de la telemetría, toda la adquisición de datos y la parte de la detección de objetivos en la cámara, en la que se clasifican los píxeles entre objetivo y no objetivo.
- La capa 1 se sitúa sobre la anterior y comprende los controladores para el seguimiento de rutas y del seguimiento de objetivos, ya sean estáticos o en movimiento. Además, incorpora la otra parte de la detección de objetivos, la segmentación de las imágenes para distinguir la forma del objetivo y

geolocalizarlo de ser necesario.

- Sobre la anterior se sitúa la capa de generación de rutas y estimación estocástica, denominada capa 2. La estimación de esta capa produce una aproximación de la posición de un objetivo estático o en movimiento basándose en las observaciones pasadas del objetivo.
- La capa 3 es la encargada de los comportamientos colaborativos, tanto los relativos al control como a la sensorización. La asignación de tareas se realiza en este nivel, así como la fusión de los datos de los sensores, por ej. en una aplicación de localización se comparten los mapas y la información acerca de los objetivos entre los miembros del equipo.

Los programas de lenguaje CSL pertenecen a esta última capa. Este lenguaje trata de solventar problemas de estos sistemas tales como los cambios en la topología de la red al añadir o quitar sensores a lo largo del tiempo, o la asignación de tareas, que no se realiza a un UAV concreto, sino que se deja al algoritmo seleccionar entre los agentes disponibles, proporcionando a los operadores la habilidad de cambiar los controladores en tiempo de ejecución. Esto resulta importante en misiones persistentes, ya que la estrategia de la misión probablemente cambiará a lo largo del tiempo debido a situaciones imprevistas.

La plataforma C3UV fue diseñada para organizar y controlar redes de UAV para la adquisición de información. Su primer enfoque se centró en el control de UAV colaborativos y en 2004 se utilizó para demostrar la viabilidad de realizar el control de múltiples UAV con un solo operador. Cuando la práctica vigente en aquel momento era tener 3 operadores para controlar cada UAV, mediante el uso de la plataforma se consiguió mantener 4 UAV operados por un único operador [Ryan et al., 2007]. El enfoque para lograr esto se basa en que el operador se centre en declarar tareas de alto nivel mediante interfaces gráficas. Estas interfaces reflejan el estado del sistema al operador a un nivel de abstracción relacionada con la capacidad cognitiva de un solo individuo, y confía tanto la asignación de tareas como su ejecución a la autonomía de la propia plataforma.

En una segunda fase, la plataforma cambió su enfoque hacia la adquisición de información de forma colaborativa. Con ella se han demostrado las aproximaciones desarrolladas en tareas de búsqueda y localización de estructuras [Rathinam et al., 2005] e incluso objetivos en movimiento [Sengupta, 2008], [Garvey et al., 2011] además de sensorización distribuida y colaborativa [Ryan et al., 2007].

En el año 2007 se realizaron pruebas con esta plataforma para la detección y situación de estructuras tales como ríos o líneas costeras en tiempo real empleando la fusión de los datos de los sensores. Para la detección se empleaba el tratamiento de imágenes capturadas por la cámara y para la localización el

histórico de imágenes capturadas. En el año 2009 y 2010 se probó un sistema de seguimiento de peatones basado en imágenes capturadas por cámaras equipadas con gimbals. El equipo, formado por dos UAV monitorizaba una tienda de campaña. Al detectar a la persona salir de la tienda este avisaba al segundo UAV para que procediese a seguir al peatón mientras él continuaba la vigilancia.

A partir de esta fecha se redujo considerablemente la actividad de este grupo de investigación, como puede comprobarse en su página web. Parte de los integrantes del mismo continúan trabajando con UAV pero su investigación se centra ahora en la seguridad y el control del tráfico aéreo [Berkeley, 2016].

3.4.2. Plataforma de la Universidad Técnica Checa

Esta plataforma surge a partir de «Agentfly», un sistema multi-agente desarrollado en el «Agent Technology Center» de la Universidad Técnica Checa que permite la simulación a gran escala de tráfico aéreo civil y vehículos no tripulados. El sistema integra la planificación avanzada de la trayectoria de vuelo y la prevención de colisiones descentralizada con modelos muy detallados de los aviones y del medio ambiente. Si bien su objetivo principal era la simulación de la supervisión del tráfico aéreo, el conocimiento adquirido sirvió para trabajar en una plataforma que permitiera desarrollar aplicaciones con múltiples UAV. Esta plataforma se basa en la evolución progresiva de comportamientos inteligentes mediante una arquitectura modular de mando y control (C2) para equipos de UAV y el empleo de la realidad mixta [Jakob et al., 2012]. Entre estos comportamientos han puesto un especial énfasis en la evasión de colisiones entre los UAV, algo razonable teniendo en cuenta el origen de la misma.

La arquitectura propuesta tiene una estructura formada por siete capas. Las dos superiores son las mismas para todos los agentes independientemente de su nivel de virtualización mientras las demás pueden ser hardware o software.

- Empezando por la parte inferior, tenemos la capa de entorno, que representa el entorno en el que existen y desarrollan su actividad los agentes. Puede ser real, en caso de que se trabaje sobre el mundo real o simulado, para lo que se debe generar el mundo virtual simulado.
- La siguiente es la capa de activos en la que se define la dinámica de vuelo de las entidades del sistema. Dependiendo del nivel de virtualización, la dinámica de vuelo de un activo puede estar formada por comportamientos pre-codificados, modelos más complejos en MATLAB, un simulador de vuelo o en caso de nivel cero de virtualización, un UAV real.
- Sobre esta se sitúa la capa del piloto automático en la que pueden integrarse pilotos automáticos hardware o un conjunto de algoritmos o sistemas de

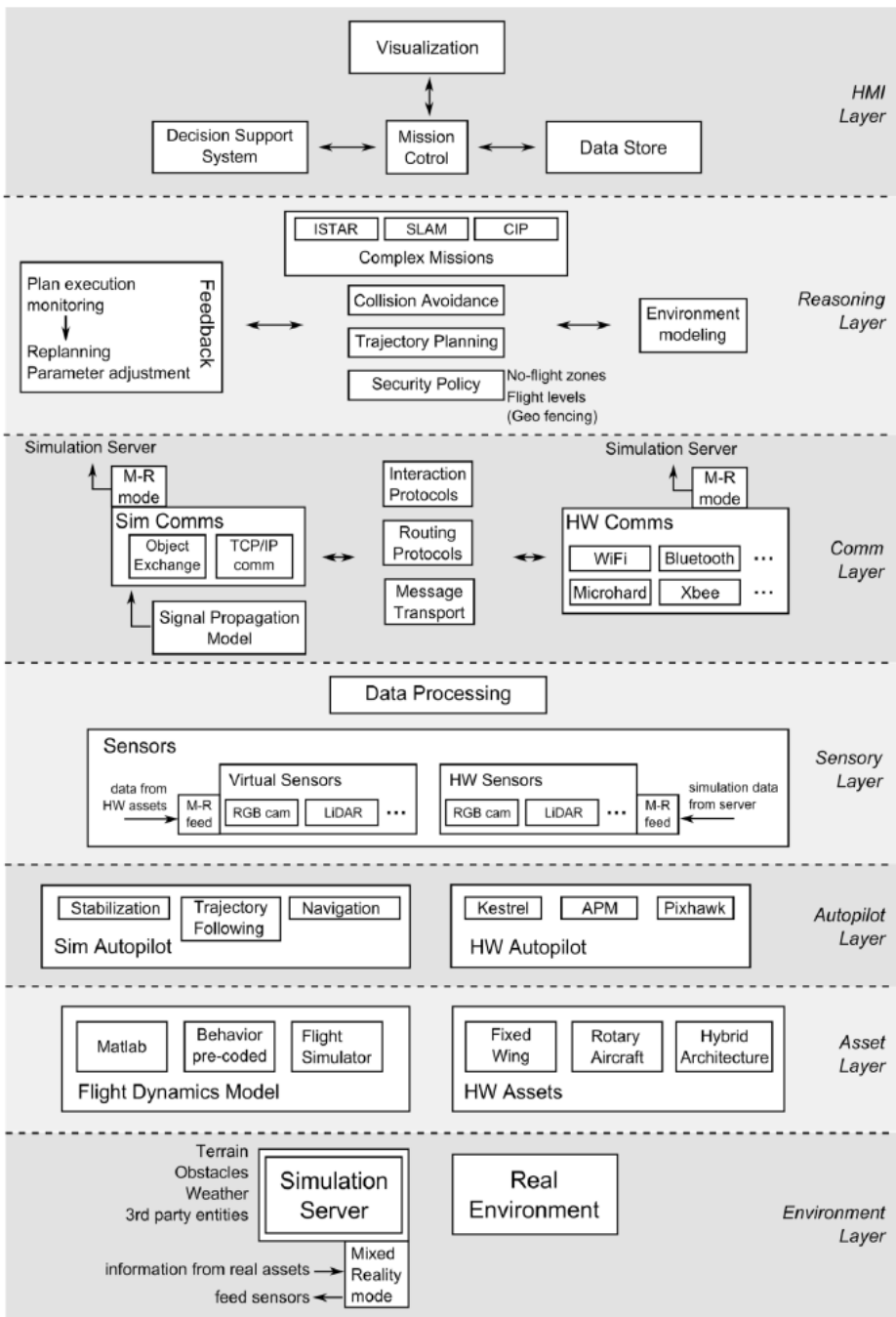


Figura 3.3: Esquema de la estructura software de la plataforma de la Universidad Técnica Checa

control que simulan el piloto automático y los controles de estabilización, seguimiento de trayectorias y navegación.

- La capa de sensores define la carga útil que se implementa en los agentes, tales como la cámara, Lidar, etc. Cualquiera de los dispositivos de hardware reales o sus modelos pueden ser utilizados. En el caso de escenarios de realidad mixta en que se utilizan tanto sensores reales como virtuales, los sensores emplean el servidor de simulación de forma que se pueda usar en la simulación datos reales provenientes de los sensores.
- Para permitir la coordinación de las acciones los agentes emplean una capa de comunicaciones para intercambiar los mensajes. Si se trata de comunicaciones reales se emplean módulos inalámbricos e interactúan con los virtuales emulando comunicaciones TCP/IP.
- El control de alto nivel y razonamiento de los UAV se define en la capa de razonamiento, que es la misma para todos los agentes y no depende del nivel de virtualización.
- Finalmente la capa superior, la de interfaz hombre máquina, es la empleada por los operadores del sistema para controlar los equipos de UAV. Consiste en un bloque de control que permite la asignación de tareas a los UAV y el procesamiento de la información entrante. Esta información se visualiza y almacena para su posterior análisis y/o reproducción.

En lo referente a la propuesta para el desarrollo, se parte de un entorno completamente virtual en el que los UAV son agentes con sensores básicos simulados. A partir de aquí se pasa a los escenarios de realidad mixta, en los que dependiendo del nivel de virtualización, puede tratarse de agentes con una mezcla de capas de la arquitectura simuladas y otras hardware (configuración Hardware-in-the-loop) o agentes simulados interactuando con agentes hardware.

En 2014 se realizaron vuelos de prueba con la plataforma formada por dos UAV reales de ala fija y se diseñó un escenario de prueba para demostrar el funcionamiento de un sistema de control para evitar activamente las colisiones. En dicho escenario se generaban trayectorias con riesgo de colisión y se analizaba la respuesta de los dos UAV [Selecký et al., 2015]. Las pruebas se realizaron en un primer momento con dos UAV simulados, luego uno real y otro simulado y se ajustó el algoritmo para subsanar desviaciones entre la simulación y la realidad y finalmente se realizó un vuelo de prueba con los dos UAV reales.

3.4.3. A Framework for Distributed Autonomous Systems (AFDAS)

El Australian Center for Field Robotics (ACFR) de la Universidad de Sídney cuenta con gran experiencia en el desarrollo de aplicaciones para UAV y se caracteriza por la realización de pruebas reales de los algoritmos y estrategias planteados. Para optimizar el proceso han desarrollado una metodología que permite la transición, comprobación y validación de las estrategias complejas de la simulación a las pruebas reales. Dicha metodología se centra especialmente en minimizar la modificación del código de los algoritmos entre las diversas fases de prueba.

Las misiones para múltiples UAV típicas, como la fusión descentralizada de sensores, *SLAM* [Kim, 2004] y las misiones de demostración de control de cooperación [Cole et al., 2010] consisten en fases de desarrollo con tareas similares o idénticas: el modelado del sensor, la comunicación, el registro de datos y la reproducción, la visualización y la sincronización del tiempo. Con el fin de reducir al mínimo las soluciones repetidas se necesita un enfoque sistemático para el diseño, desarrollo, implementación y mantenimiento del software del sistema de UAV.

AFDAS utiliza un método de descomposición de arriba hacia abajo y estructura el software en varias capas. Las capas de control tienen diferentes niveles de abstracción, las capas de control de más alto nivel tienen tareas más abstractas y a medida que descendemos la abstracción disminuye, de forma inversa a lo que sucede con los requisitos de tiempo real. El software de la plataforma está diseñado de una forma jerárquica. Como se ilustra en la Figura 3.4 y se pueden distinguir cuatro capas; actuador, subsistema, sistema, y la capa del sistema de sistemas (SoS).

- La capa de actuadores proporciona acceso a los actuadores a través de controladores electrónicos y módulos de software de controlador de dispositivo de bajo nivel. Es común emplear placas basadas en microcontroladores para controlar los actuadores. Estas placas tienen unos fuertes requerimientos de trabajo en tiempo real para manejar correctamente el hardware, por lo que los controladores de dispositivos suelen estar optimizados y a menudo programados en ensamblador o C con compiladores propietarios.
- La capa subsistema proporciona el control de bajo nivel y los procedimientos de fusión. La fusión de datos de sensores de bajo nivel, tales como la fusión INS-GPS para soluciones de navegación y los controles de los servomotores de bajo nivel, tales como bucles de servo PID son tareas típicas

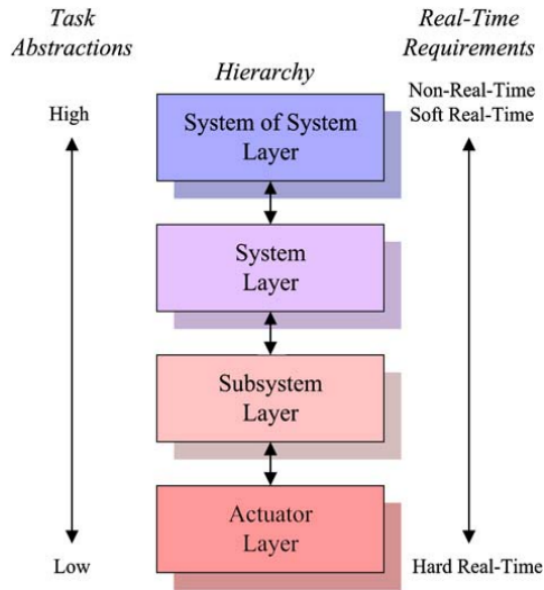


Figura 3.4: Niveles de software definidos en AFDAS

de este nivel. Esta capa se construye alrededor de múltiples micro PC o placas de microcontrolador, conectados a los controladores de servo y sensores. Las tareas en la capa de subsistema a menudo se programan en C / C ++ y se ejecutan de forma periódica en tiempo real. Las tareas concurrentes que se ejecutan en un único procesador o embebidos en varios equipos conectados en red utilizan mecanismos de comunicación entre procesos (IPC) para el intercambio de datos.

- La capa de sistema proporciona control de alto nivel y funcionalidades de toma de decisiones. Algunas de las tareas de esta capa son la diagnosis y el manejo de errores, el cálculo de trayectorias y la navegación basada en puntos de paso. Estas tareas pueden ser periódicas o estar basadas en eventos en tiempo real. La capa de sistema consiste en un conjunto heterogéneo de procesadores, unidos mediante red LAN o WAN. Los ordenadores de esta capa utilizan varios mecanismos de comunicación de alto nivel para el intercambio de datos entre procesos.
- Finalmente, la capa de sistema de sistemas (SoS) es la de más alto nivel en la jerarquía de control. Comparado con las capas inferiores los requerimientos de tiempo real son bastante menos exigentes. Las tareas de esta capa se concentran en la interacción e interoperabilidad de varios sistemas y puede haber operadores humanos en el lazo de control de esta capa.

El desarrollo de AFDAS se produjo a lo largo de 10 años de trabajo en las áreas de control y fusión de datos descentralizada, en el año 2006 aparecen las primeras publicaciones con la estructura propuesta [Cole et al., 2006], en 2007 realizan pruebas con 2 UAV cooperando para maximizar la obtención de información y en 2009 se presenta una tesis doctoral en la que se concentran gran parte de los avances de este grupo sobre el área [Cole, 2009]. Desde entonces a la actualidad han continuado empleando la arquitectura propuesta para las aplicaciones con múltiples UAV centrándose en la realización de maniobras complejas en las que se ve envuelto más de un UAV como una simulación de repostaje en vuelo [Wilson et al., 2015]

3.5. Conclusiones

En este capítulo se ha realizado una breve introducción que pone de manifiesto el potencial de los trabajos con UAV, con ejemplos concretos de investigaciones sobre aplicaciones reales y con utilidad práctica. Asimismo, se han razonado las ventajas asociadas al uso de equipos de múltiples UAV destacando las principales problemáticas que se han de afrontar. Se ha realizado una revisión de las estrategias de coordinación empleadas, destacando la proliferación en los últimos años de estrategias basadas en metaheurísticas, especialmente aquellas basadas en algoritmos bioinspirados.

Finalmente se han repasado brevemente los principales trabajos en lo referente a plataformas o arquitecturas para la coordinación de UAV. En el caso de la plataforma C3UV se trata de una aproximación en la que el nivel superior de coordinación se realiza mediante un operador, que empleando las ayudas de los sistemas electrónicos de bajo nivel es capaz de controlar más de un UAV simultáneamente. Sin embargo, no busca alcanzar la plena autonomía y que los equipos de UAV sean controlados por una estrategia de coordinación tal y como se propone en el presente trabajo.

La plataforma de la Universidad Técnica Checa se ha desarrollado al mismo tiempo que el presente trabajo y sigue una aproximación similar, especialmente en lo que se refiere al empleo de la realidad mixta como forma de superar el reality gap. Sin embargo, el objetivo final de la plataforma es el control de tráfico aéreo y el establecimiento de rutas aéreas automáticas con UAV y se aleja de la coordinación para lograr la colaboración en una tarea.

Finalmente la plataforma de la Universidad de Sídney persigue también el objetivo de crear una estructura software reutilizable para la realización de múltiples operaciones y actividades con UAV, pero no contempla el uso de la realidad mixta para la implementación en la realidad de las estrategias de coor-

dinación desarrolladas y tampoco se proponen estrategias para que múltiples UAV colaboren.

Por lo tanto, existen limitaciones en las aproximaciones existentes y, debido a ello, aspectos a mejorar en la definición de nuevas plataformas para facilitar el desarrollo de estrategias de coordinación para múltiples UAV que realicen una tarea de forma colaborativa.

Capítulo 4

Metodología propuesta

4.1. Introducción

Una vez realizado un análisis crítico del marco teórico en el que encaja el trabajo de esta tesis, en este capítulo se proporciona una visión global de la metodología que forma parte de la aproximación al proceso de desarrollo de estrategias de coordinación propuesto. Así, en lo que sigue, se hará una introducción a los conceptos y elementos más relevantes de dicha metodología. En el Capítulo 5, por su parte, se describirá con más detalle la plataforma software de realidad mixta que constituye la otra parte de la aproximación.

A lo largo de la revisión bibliográfica previa se han destacado diversos retos y problemas asociados al desarrollo de estrategias de coordinación, así como el enfoque de las plataformas y aproximaciones existentes. Este trabajo parte de las lecciones aprendidas a través del estudio de esas plataformas, sus limitaciones y aportaciones, tratando de aunar el mayor número posible de fortalezas, especialmente aquellas orientadas a resolver los problemas de los equipos de múltiples UAV para la consecución de tareas colaborativas.

Los objetivos que se persiguen con las plataformas están relacionados principalmente con la reducción del tiempo requerido para el desarrollo de estrategias de coordinación, el potencial para reutilizar la estrategia o adaptarla a UAV diferentes, así como un aumento de la seguridad en la implementación de la misma en la realidad. Si bien en todas las plataformas revisadas se trata de facilitar desarrollos con múltiples UAV se ha detectado una carencia de soluciones centradas en conseguir la colaboración entre múltiples UAV para una tarea común. Las soluciones existentes se centran en maniobras complejas, tales como los vuelos coordinados o las acciones evasivas en caso de riesgo de colisión, en las que generalmente se emplea un número pequeño de UAV.

El enfoque hacia la coordinación de equipos heterogéneos para lograr realizar tareas de forma colaborativa supone buscar formas de afrontar el problema de las diferencias en lo referente a plataformas y electrónica embarcada; así como el desarrollo de métodos rápidos y seguros de trasladar los resultados de las simulaciones a la realidad, con el objetivo de comprobar la efectividad de las estrategias propuestas. Para afrontar estas cuestiones se propone una metodología fundamentada en tres aspectos principales. Por una parte, una estructura normalizada y modular del software, que sirve de base para toda la metodología. En segundo lugar, una separación entre la estrategia de coordinación y el hardware concreto, igualando así la interacción con hardware heterogéneo. Finalmente, y apoyado en lo anterior se propone la introducción progresiva de la complejidad mediante el uso de la realidad mixta.

4.2. Estructura normalizada y modular para el software

Actualmente la investigación con UAV cuenta con una gran actividad, tanto en lo referente al hardware como al software. Todo este esfuerzo impulsa una generación continua de novedades y mejoras, que por una parte van solucionando las carencias y problemas de los productos anteriores, pero por otra generan una necesidad de actualización y adaptación a las novedades de forma constante. Para facilitar la adaptación a esos cambios resulta muy útil poseer un software estructurado y que permita modificar únicamente las secciones afectadas, permitiendo mantener el resto.

En esta tesis se defiende la premisa de que una buena aproximación para afrontar el desarrollo de estrategias de coordinación es la constitución de una estrategia basada en bloques funcionales y modulares, que proporcionen a los desarrolladores una serie de patrones de diseño y herramientas con las que resolver problemas y facilitar su trabajo. En oposición, otros estudios y trabajos se focalizan únicamente en objetivos funcionales que son abordados con sistemas ad-hoc. Esta última aproximación a menudo conduce a la construcción de sistemas aislados y la imposibilidad de reutilización de resultados, pero se suele justificar debido a que el tiempo de desarrollo es menor.

Para maximizar la utilidad de la estructura sin llegar a una rigidez que alargue demasiado el desarrollo o dificulte ceñirse a la misma se ha buscado seguir una serie de premisas principales a la hora de diseñarla:

- La simplicidad debe ser uno de los principios clave que guíen el diseño: el grado de complejidad a la hora de implementar un nuevo componente funcional sobre la estructura debe depender en mayor medida de los requi-

sitos de la propia funcionalidad que de los requisitos de implementación que imponga la estructura.

- La flexibilidad y la visión funcional como otro de los principios fundamentales de diseño: la estructura debe ser lo suficientemente flexible como para permitir al desarrollador elegir cuáles de los componentes funcionales implementa y con que nivel de complejidad. Asimismo, la estructura debe estar abierta a la utilización de diferentes lenguajes de programación, protocolos de comunicación y en general a las diferencias de hardware. Es decir, debe dar prioridad a la funcionalidad frente a la tecnología o la forma concreta en la que esta se ofrece.
- Fomentar la reutilización del código, de forma que se evite la reescritura de funcionalidades comunes en aplicaciones posteriores: frente a la concepción monolítica del código, presente especialmente en la concepción clásica de la programación de bajo nivel; la división del código en módulos funcionales más pequeños aporta independencia y a la vez una mayor facilidad para depurar los errores.
- Mantener el foco en el desarrollo de las funcionalidades principales: no todas las aplicaciones requieren la misma complejidad en todos los componentes funcionales; mediante la división en módulos se facilita que el desarrollador centre su esfuerzo en aquellos componentes que afectan más directamente a la aplicación, evitando que la atención y los esfuerzos se dispersen a áreas menos importantes.
- La concepción de la seguridad como una parte básica del sistema: debe ser uno de los aspectos principales a la hora de implementar la estrategia de coordinación en la realidad, por lo que se incluye dentro de los bloques funcionales básicos que se deben implementar desde el principio.
- Favorecer el trabajo simultáneo: actualmente los tiempos para el desarrollo de un proyecto son cada vez más cortos y en la mayoría de los casos varias personas han de coordinar su trabajo para alcanzar la metas fijadas. Una estructura modular permite que cada uno de ellos trabaje sobre un componente funcional y todos ellos se unan para formar el sistema final de la forma más sencilla posible.
- Permitir la combinación de UAV reales y simulados: mediante el empleo de la realidad mixta se busca mejorar varios aspectos problemáticos del desarrollo e implementación de estrategias de coordinación como las primeras pruebas reales, la mejora de los modelos de simulación... por lo que resulta imprescindible tener esto en cuenta a la hora de sentar las bases del sistema.

Uno de los métodos más conocidos para resolver un problema complejo es dividirlo en problemas más pequeños, de forma que en vez de resolver una tarea compleja y tediosa en conjunto se resuelven multitud de tareas más sencillas y mediante la suma de todas ellas se puede llegar a la solución global. Teniendo en cuenta estas premisas y la elevada complejidad de tratar el problema planteado de forma directa con un único programa se propone optar por definir una estructura formada por módulos o bloques funcionales. Estos bloques deben permitir, mediante la implementación de los mismos, dotar al sistema de las funcionalidades requeridas.

La estructura modular aporta una serie de beneficios, que van en la línea de las premisas que se persiguen con este trabajo. Entre estos beneficios destacan la sencillez, ya que cada uno de los subprogramas que se encarga de implementar un bloque funcional resulta mucho más sencillo que el programa global. Asimismo se promueve la reutilización del código y la seguridad; ya que una vez definidos los límites de un bloque este se puede reutilizar en proyectos posteriores de forma fácil y con garantías, ya que ya ha sido probado. Se facilita la depuración, prueba y mantenimiento del código; resulta mucho más sencillo entender el código y localizar errores en el mismo cuando su volumen es menor. Finalmente, la programación modular permite aumentar funcionalidades en uno de los componentes sin alterar el resto, lo que facilita añadir nuevas funciones o mejoras al sistemas de forma iterativa.

Por su parte, también acarrea un coste para el desarrollo de código. Para empezar se ha de desarrollar una estructura que sirva de guía y unos límites claros para los módulos, así como unos protocolos para intercambiar información con el resto del programa. Esta tarea puede acabar consumiendo mucho tiempo, por ello en este trabajo se propone una estructura genérica que se pueda seguir y que minimice este problema. Otro de los inconvenientes que presenta es un mayor uso de memoria y de tiempo de ejecución, comparado con una solución única. Si bien no resulta despreciable, la capacidad de cómputo de los sistemas actuales reduce en gran medida la importancia de este problema.

En la concepción de los equipos de UAV a lo largo de este trabajo existen dos figuras principales; por una parte están los UAV y por otro la estación en tierra:

- Los UAV, dado que los equipos con los que se trabaja son heterogéneos, abarcan todo tipo de vehículos aéreos: multirrotores, aviones, helicópteros... y la electrónica embarcada en los mismos, tanto para el control como la sensorización o actuación propia para la misión a realizar. La estructura propuesta es común para todos ellos, ya que desde un punto de vista funcional todos son muy similares, independientemente de su tamaño, forma de generar sustentación o electrónica embarcada. Se incluye también

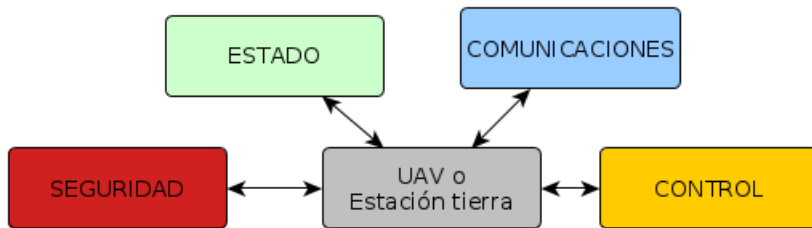


Figura 4.1: Categorías de la estructura software propuesta

en este grupo a los UAV y sensores simulados y los estados intermedios entre ambos como las pruebas HIL (Hardware In the Loop) en las que se combina el simulador con hardware de los UAV reales.

- En lo que se refiere a la estación en tierra, se entiende como tal el conjunto de herramientas que permitan la supervisión y seguimiento del desarrollo de la misión por parte del operador. Esto comprende el entorno virtual en el que desarrollan su actividad los UAV simulados y que incluye, por ejemplo, los modelos de los fenómenos que se van a estudiar en una misión de toma de datos. Dependiendo del tipo de estrategia que se vaya a desarrollar, puede contener también parte de la estrategia de coordinación. Si, por ejemplo, la estrategia es centralizada la mayor parte de la misma se implementa en la estación en tierra, mientras que los UAV únicamente siguen las órdenes enviadas desde esta.

Mientras que generalmente resulta sencillo distinguir ambas figuras, al tratarse de implementaciones sobre hardware diferente, al introducir los UAV y los sensores simulados puede diluirse esta diferencia, ya que el hardware para ambas partes puede ser el mismo. En nuestro caso la división se realiza siempre desde un punto de vista funcional y así, por ejemplo, se puede emplear un ordenador portátil para la visualización del desarrollo de la misión, y lanzar sobre el mismo, de forma simultánea la ejecución de uno o varios UAV simulados.

De acuerdo con las premisas descritas en el apartado anterior para el desarrollo de la estrategia de coordinación se propone una división de los bloques funcionales de los UAV o de la estación en tierra, se encuadran en cuatro categorías generales que comprenden las funcionalidades básicas, tal y como se muestra en la Figura 4.1:

- El control: comprende todo lo referente al manejo de la aeronave, por lo que se trata de una categoría en la que se incluyen varios niveles de código, ya que abarca desde la actuación de las superficies de control o el motor del

UAV, hasta la propia estrategia de coordinación. Dentro de esta categoría se definen los diversos modos de control de la aeronave y la forma de acceso a los mismos, la estrategia de coordinación colaborativa, el controlador de vuelo básico que extiende las capacidades del piloto automático, etc.

- Las comunicaciones: categoría en la que se incluye todo lo relativo al intercambio de información entre los UAV y la estación de tierra. Constituye una de las partes en las que la complejidad aumenta bastante al tratarse de una plataforma en la que se quiere trabajar simultáneamente con UAV reales y virtuales frente al equivalente únicamente con UAV reales. Todo el protocolo de comunicación, los tipos de mensajes que se van a emplear, la estructura de la red de comunicaciones así como el procedimiento que se debe seguir en caso de errores de transmisión se incluyen en esta categoría.
- La información de estado: comprende tanto la información de estado de cada uno de los UAV como la del entorno sobre el que se lleva a cabo la misión. Asimismo se incluyen también las herramientas que permiten al operador visualizar y supervisar de forma más sencilla toda la información relevante.
- La seguridad: recoge todos los mecanismos, características o prestaciones cuya función es evitar o prevenir accidentes. La evasión de obstáculos, sean estos fijos o móviles; la respuesta ante fallos en la sensorización, las comunicaciones o problemas mecánicos de los UAV, etc. Si bien es un aspecto que se ha de tener en cuenta también en el diseño y selección del hardware más apropiado, resulta indispensable definir todo lo necesario también en el código para permitir la operación segura del equipo de UAV.

Con la definición de estas categorías se pretende mostrar la visión funcional de este trabajo y trazar una delimitación previa, más general que la de los bloques funcionales, que concretan en mayor medida la estructura software propuesta. Dado que los bloques funcionales y los flujos de información entre ellos son diferentes para los UAV y la estación en tierra se trata la estructura de cada uno de ellos por separado.

4.2.1. Estructura software para los UAV

En lo que se refiere al UAV, los bloques funcionales que constituyen la estructura propuesta se muestran en la Figura 4.2

Estos bloques se engloban en las categorías definidas en la Figura 4.1 y constituyen la base o guía para el desarrollo de todo el software para el UAV. Se trata de una propuesta de mínimos, pudiendo ampliarse con nuevos bloques

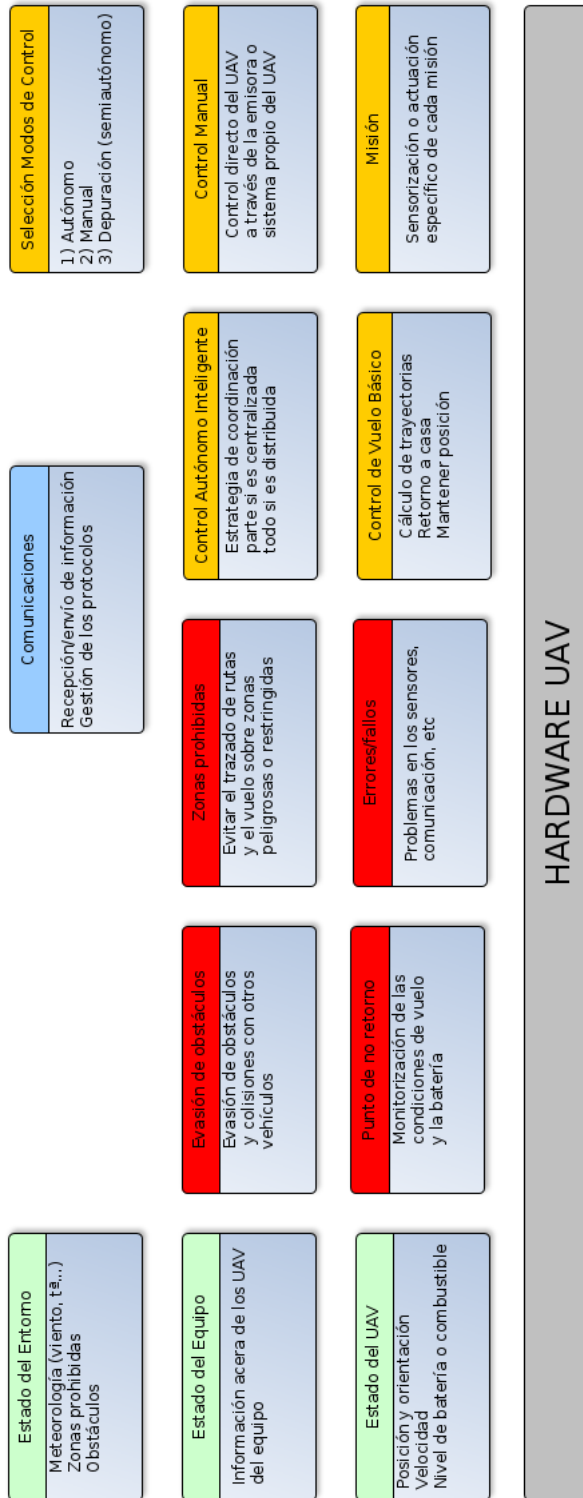


Figura 4.2: Bloques funcionales propuestos para los UAV

o subdividirse los presentados según la aplicación o las necesidades de cada caso concreto. La extensión o contenido de los mismos es variable y se ha optado por tratar de mantener un número limitado de los mismos para favorecer la claridad de los esquemas, pudiendo mostrar todas las categorías en una única imagen.

Control Comenzando por la parte de control, se distinguen cinco bloques funcionales, tal y como se puede ver en la Figura 4.2

Existe un bloque encargado de la selección del modo de control del UAV, que es controlado desde la estación de tierra y en el que se definen todos los modos y/o submodos que se consideren necesarios. Inicialmente, deben definirse al menos dos modos, uno completamente manual en el que el operador toma el mando directo de la aeronave y otro completamente autónomo en el que el control se realiza mediante las órdenes de la estrategia de coordinación, que se calculan en función de los objetivos de la misión. Además, para la realización de las pruebas iniciales y el ajuste de los controladores de vuelo resulta recomendable definir un estado intermedio o de depuración, en el que se maximice la toma de datos y se controle el UAV dentro de unos límites más estrictos para garantizar la seguridad. Pueden existir también modos asistidos que permitan el control manual con ayudas para determinadas maniobras o situaciones.

El control autónomo inteligente, que contiene la estrategia de coordinación y tiene una complejidad determinada principalmente por el tipo de estrategia seleccionada. En el caso de estrategias centralizadas únicamente transmite información, por contra, cuando se trabaja con estrategias distribuidas es el encargado de traducir los objetivos de la misión a comandos para el UAV. Se comunica principalmente con el control de vuelo básico y con la parte de misión.

El control de vuelo básico constituye la parte de más bajo nivel del control autónomo, y es una extensión del piloto automático para permitir funciones más avanzadas como el cálculo de trayectorias, el retorno autónomo a la zona de despegue (denominado a veces «vuelta a casa») y el mantenimiento de la posición o vuelo en círculo sobre un punto. Su principal función es traducir los comandos del control autónomo inteligente a posiciones de las superficies de control del vehículo y a la regulación de la potencia de los motores, manteniendo la aeronave en vuelo estable y controlado.

El bloque de misión condensa todo aquello relacionado con la aplicación concreta que se pretende realizar con el equipo de UAV. Comprende tanto la lectura como el filtrado de la sensorización específica o, en caso de misiones en las que se requiera, también la actuación. Si bien puede considerarse que no forma parte directa de la categoría de control, su estrecha relación con el mismo junto con la posibilidad de incluir el control de la actuación contrarrestan esta idea.

Finalmente el control manual se encarga de recibir y transmitir a los servomotores la señal de control proveniente directamente del sistema de comunicación. Se trata de un modo que busca garantizar la seguridad en caso de que se produzca algún problema con el control autónomo o se esté depurando y trabajando todavía en el ajuste del control de vuelo básico.

Estado En lo referente a la información de estado, se han definido con tres bloques funcionales diferentes, tal y como se muestra en la Figura 4.2:

El estado del entorno o ambiente en el que se desarrolla la misión recoge todos los datos referentes a las características relevantes del mismo para la actividad que se va a realizar. Dada la importancia de los cambios que provocan en el medio aéreo las condiciones meteorológicas, estas son una de las principales fuentes de información a tener en cuenta en este bloque, ya que las variaciones afectan directamente al desempeño y comportamiento de los UAV. Asimismo, en este apartado se definen las zonas prohibidas o peligrosas, así como aquellos obstáculos de los que se tenga constancia para evitarlos.

La información sobre el estado del equipo de UAV condensa toda la información sobre los UAV de la que se dispone, incluyendo la del UAV propio. Resulta importante disponer de información como la posición y objetivos del resto de UAV para que la estrategia generada evite los solapamientos y seleccione los objetivos para el UAV de la forma más eficiente posible. La mayor parte de esta información proviene de la comunicación con el resto de UAV o de la estación en tierra.

Finalmente, el estado del propio UAV se obtiene principalmente de la lectura de los sensores embarcados (IMU, acelerómetros, pitot, altímetro...). La información proveniente de los mismos se emplea también para el control de vuelo básico así como para su transmisión a través de la red de comunicaciones para informar al resto de UAV del equipo y proceder de forma informada a la toma de decisiones. Resulta especialmente importante para el caso de misiones reales el nivel de batería o combustible, tanto para tomar decisiones estratégicas al respecto, como para garantizar la capacidad de los UAV de alcanzar el punto de aterrizaje previsto.

Seguridad La seguridad comprende cuatro bloques funcionales, tal y como se puede ver en la Figura 4.2:

En primer lugar, la evasión de colisiones con obstáculos tanto fijos como móviles. Puede tratarse de obstáculos físicos estáticos (edificios, árboles, cables de tendido eléctrico...), obstáculos móviles (tráfico rodado) o de otros vehículos aéreos en movimiento en la misma zona. Ha de recogerse el comportamiento del

UAV frente a estas eventualidades con el fin de evitar los accidentes.

La segunda posible subdivisión es la que define la evasión de las zonas prohibidas o peligrosas que se hayan definido. Esta parte se encarga de mantener el UAV en la zona de vuelo seguro prevista, evitando la entrada del mismo en zonas en las que no se tenga permitido el acceso o que, por diversos motivos, puedan generar situaciones peligrosas o no deseadas para los UAV.

El cálculo de la energía necesaria para realizar el retorno a la zona prevista para el aterrizaje constituye también una parte importante dentro de la seguridad en la operación de las aeronaves. Resulta primordial que los UAV se mantengan en un radio que permita el retorno seguro y controlado de los mismos. En caso de producirse una situación que no permita el retorno seguro debido a condiciones no previstas deben definirse los protocolos que debe seguir el UAV.

Finalmente, se recoge en todos los comportamientos previstos que se puedan realizar ante un fallo en los sensores o en el propio UAV. Para ello resulta imprescindible contar con sensorización e información de estado del UAV actualizada, así como una clasificación de la importancia del fallo que permita una respuesta acorde con el mismo. Dentro de este bloque funcional también se recoge la posibilidad de detectar fallos en la comunicación, que puede afectar, por ejemplo a la ejecución del control manual y el cambio del mismo hacia el comportamiento que se considere más seguro (retorno a la base, mantenimiento de la posición...).

Comunicación La comunicación se incluye como un único bloque funcional que se encarga de proveer al resto de bloques de la posibilidad de intercambiar información, abstrayéndolos de las problemáticas o limitaciones del funcionamiento concreto de la tecnologías de comunicación existente para la transmisión de datos.

El funcionamiento básico de la plataforma requiere de la existencia de al menos un enlace de comunicaciones inalámbrico entre los UAV y de estos con la estación en tierra. Este enlace permite la coordinación de los UAV y el conocimiento de la información de estado del equipo, así como la transmisión de información relevante que se pueda conocer sobre el entorno en el que se lleva a cabo la misión. Asimismo, permite cambiar entre los diferentes modos de control definidos y realizar el control manual del UAV.

Para garantizar la seguridad en la operación o para la transmisión de volúmenes mayores de datos puede resultar útil contar con más de un enlace de datos entre los UAV y la estación en tierra. De esta forma se puede contar con enlaces redundantes, que permitan mantener la funcionalidad ante posibles

eventualidades.

4.2.2. Estructura software para la estación en tierra

En lo que se refiere a la estación en tierra, los bloques funcionales que constituyen la estructura propuesta se muestran en la Figura 4.3

Estos bloques se engloban en las categorías definidas en la Figura 4.1 y constituyen la base o guía para el desarrollo de todo el software para la estación en tierra. Resulta similar a la planteada para los UAV, aunque presenta algunas diferencias significativas que merecen ser comentadas.

Control Comenzando por la parte de control, se han definido en este caso tres bloques funcionales, tal y como se puede ver en la Figura 4.3.

Existe un bloque encargado de la selección del modo de control del UAV, que contiene las condiciones que han de darse para que entre en funcionamiento cada uno de los modos definidos. Tal y como se describió en la parte del UAV es necesario definir un mínimo de dos modos y resulta recomendable un tercero para realizar la depuración. Además se pueden definir cuantos se consideren necesarios.

El control autónomo inteligente, cuya complejidad se fija, igual que para el UAV, según el tipo de estrategia de coordinación. En el caso de estrategias centralizadas, este bloque es el encargado de traducir los objetivos de la misión a consignas para los UAV y si se trata de una estrategia distribuida, la inteligencia se implementa sobre los bloques de los UAV, dejando a este vacío.

Por último, el control manual se encarga de transmitir a los UAV la señal de control proveniente de los joysticks, palancas o similar que emplee el operador para controlarlos. Se trata de un modo que busca garantizar la seguridad en caso de que se produzca algún problema con el control autónomo.

Estado En lo referente a la información de estado, se han definido con tres bloques funcionales, tal y como se muestra en la Figura 4.3:

El estado del entorno o ambiente en el que se desarrolla la misión contiene todos los datos referentes a las características relevantes del mismo para la actividad a realizar. Dada la importancia de los cambios que provocan en el medio aéreo las condiciones meteorológicas, estas es una de las principales fuentes de información a tener en cuenta en este bloque, ya que las variaciones afectan directamente al desempeño y comportamiento de los UAV. Asimismo, en este apartado se definen las zonas prohibidas o peligrosas, así como aquellos obstáculos de los que se tenga constancia para evitarlos.

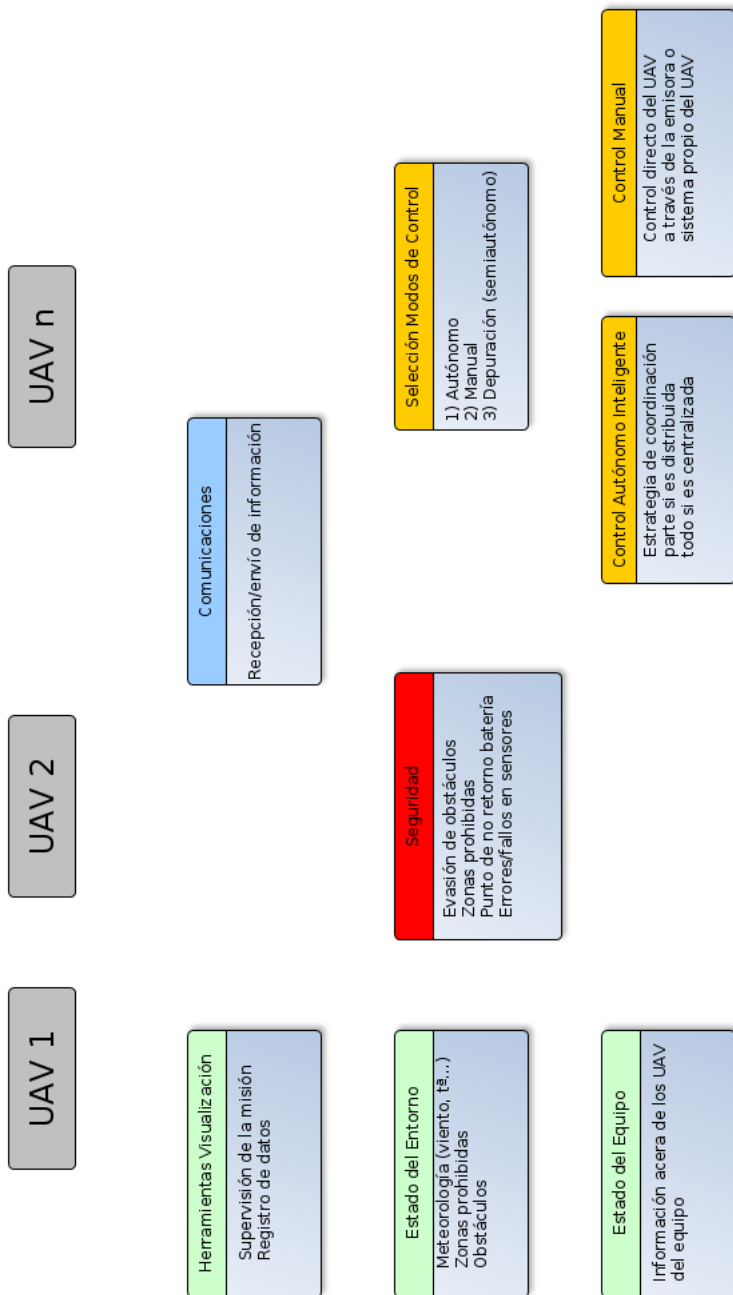


Figura 4.3: Bloques funcionales propuestos para la estación en tierra

La información sobre el estado del equipo de UAV en la que se recoge toda la información sobre los UAV de la que se dispone, incluyendo la del UAV propio. Resulta importante disponer de información como la posición y objetivos del resto de UAV para que la estrategia generada evite los solapamientos y seleccione los objetivos para el UAV de la forma más eficiente posible. La mayor parte de esta información proviene de la comunicación con el resto de UAV o de la estación en tierra.

Finalmente, las herramientas de visualización y registro de datos, se encargan de facilitar al operador el control de todos los datos relevantes para la misión, tanto del entorno como del equipo de UAV. Además se lleva el registro de los datos de telemetría en un fichero de la estación en tierra, con la finalidad de analizar posteriormente los datos y realizar correcciones o mejoras sobre determinados funcionamientos de la plataforma.

Seguridad Los bloques funcionales de seguridad son los mismos que los ya descritos en el apartado 4.2.1 ya que se concibe la posibilidad de implementarlos indistintamente en la estación en tierra o en los propios UAV, aunque se muestran en un único bloque por simplicidad. Teniendo en cuenta que se pueden producir fallos en el enlace de comunicaciones, que habría que implementar todos los bloques para cada uno de los UAV y que el tiempo de respuesta se reduce al estar implementado sobre el propio UAV, esta parece la opción más recomendable. Además de que se reducen los datos que se han de transmitir a la estación en tierra y se alivia la potencia computacional requerida, algo no despreciable en el caso de que el número de UAV sea elevado.

Se pueden implementar en este bloque comportamientos de seguridad de los UAV en grupo, tales como una vuelta a casa en caso de empeoramiento de las condiciones meteorológicas.

Comunicación La comunicación, al igual que en el caso de los UAV, se incluye como un único bloque funcional que se encarga de proveer al resto de bloques de la posibilidad de intercambiar información, abstrayendo al resto de bloques funcionales de las problemáticas o limitaciones del funcionamiento concreto de la tecnologías de comunicación existente para la transmisión de datos.

4.3. Independencia del hardware

El actual auge de los UAV impulsa la investigación y desarrollo de novedades de forma continua en todos los ámbitos relacionados con estos. Estas novedades cobran forma en el caso de las aeronaves con nuevas configuraciones,

el empleo de materiales diferentes para su fabricación, etc. Existen multitud de configuraciones de aeronaves, en algunos casos tratan de simplificar maniobras complejas como el aterrizaje y el despegue en UAV de ala fija mediante el empleo de aeronaves de despegue vertical (VTOL) o buscan formas diferentes de generar sustentación, como en el caso de los ornitópteros. En lo referente a los materiales se busca aligerar lo máximo posible la estructura para aumentar la eficiencia en vuelo y está cobrando mucha fuerza en los últimos años la impresión 3D de piezas, o incluso de UAV completos [Southampton, 2016] debido a su bajo coste y la facilidad de crear piezas personalizadas de forma rápida.

En lo que se refiere a la electrónica embarcada, se debe sumar a las novedades la tendencia de multitud de fabricantes a generar soluciones o ecosistemas cerrados en los que se incluyen únicamente los controladores, sensores y actuadores más comunes y que emplean protocolos de comunicación propietarios entre los mismos. Esto dificulta en gran medida la mezcla de productos de más de un fabricante y limita las capacidades de expansión del sistema al complicar la incorporación de nuevos sensores o actuadores. Frente a estas soluciones cerradas han aparecido alternativas abiertas muy prometedoras como Ardupilot [Ardupilot, 2016], PX4 [Px4, 2016], etc que incluso permiten el acceso al código de forma libre y gratuita.

Teniendo en cuenta la variedad en el hardware y la programación del mismo resulta prioritario establecer mecanismos que permitan facilitar el desarrollo de estrategias de coordinación sobre la diversidad y heterogeneidad existente actualmente. Con el objetivo de aliviar esa carga para los desarrolladores en este trabajo se propone la definición de una interfaz común para los UAV, basada en la inclusión de un middleware, de forma que se homogeneice la comunicación con los mismos y se minimicen los problemas asociados a la renovación o sustitución de hardware concreto.

Esta homogeneización resulta más interesante si cabe en el caso que se plantea en este trabajo, ya que cuando al tratarse UAV heterogéneos la variedad es constante y no es una adaptación puntual debida a un cambio. No obstante, que la comunicación con los UAV se pueda realizar de forma homogénea no implica en ningún momento que no se deban tener en cuenta las características diferenciadoras de cada aeronave a la hora de diseñar y desarrollar la estrategia de coordinación. Las estrategias más eficientes son aquellas capaces de extraer el máximo provecho de los recursos disponibles, que en este caso vienen determinados por los UAV y su electrónica embarcada.

Con la finalidad de maximizar la utilidad del middleware y fijar los principios básicos que faciliten su implementación se han establecido una serie de premisas u objetivos que se persiguen con el uso del mismo:

- La simplicidad debe ser uno de los principios clave que guíen el diseño: al igual que en el caso de la estructura para el software se pretende que el middleware sea lo más sencillo posible y su inclusión no provoque un aumento del trabajo significativo a la hora de implementarlo.
- La flexibilidad: resulta una característica clave en este componente. El middleware ha de permitir adaptarse al mayor número posible de sistemas existentes en la actualidad. Esto supone hacer frente a la enorme heterogeneidad existente en los sensores, así como en los controladores y pilotos automáticos.
- Abstracción del protocolo de comunicación. Dada la heterogeneidad que existe en la electrónica embarcada en los UAV, especialmente en lo referente a los sistemas de control y sus protocolos de comunicación, acotar el hardware bajo este criterio puede resultar limitante a la hora de seleccionar el más adecuado. Debido a esto se busca que el middleware funcione con el mayor número posible de protocolos, de forma que en los criterios de selección primen el resto de especificaciones, más que la forma de transmitir la información.
- Permitir la conexión de hardware simulado, dado que algunos elementos como los sensores o las propias aeronaves se pueden simular en base a modelos computacionales, permitir la comunicación de este tipo de modelos con el sistema a través del middleware puede ampliar las funcionalidades del sistema final para aplicaciones como las simulaciones de fallos en la instrumentación.
- Minimizar el efecto de cambios de hardware sobre la programación. Cuando el acceso a los recursos hardware se produce de forma directa y se sustituye algún elemento hardware se deben modificar todas las interacciones del programa con el mismo. En el caso de que se incluya un middleware, únicamente es necesario modificar el middleware, ya que la interfaz para el resto del código se mantiene inalterada.

4.3.1. Niveles de software

Conviene llegados a este punto aclarar que en muchas ocasiones al referirnos al hardware no lo se hace exclusivamente a la concepción clásica en la que el hardware abarca únicamente los elementos tangibles, que en el caso de los UAV esta compuesto por la aeronave física, con las superficies de control, motores, etc y los componentes físicos de la electrónica embarcada. Actualmente lo más frecuente es trabajar con sensores inteligentes, que contienen uno o más sensores básicos y la circuitería de soporte. La electrónica complementaria,

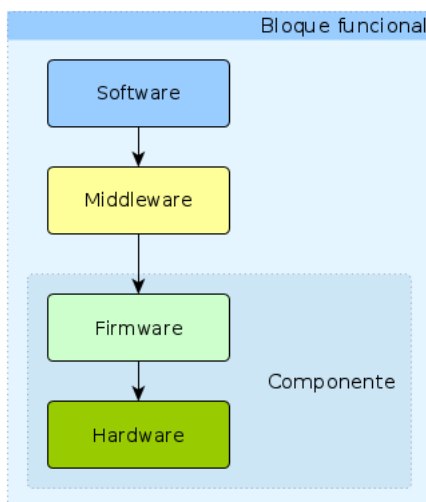


Figura 4.4: Niveles de software propuestos

que suele contener un firmware aporta capacidades como el almacenamiento de los parámetros de configuración o calibración, además de transformar la señal analógica a señales digitales e implementa protocolos de comunicación.

Así, en este trabajo se hace referencia frecuentemente a la abstracción de hardware, aunque realmente la capa de abstracción se sitúa sobre el firmware o software de más bajo nivel. Así, lo que se pretende es que se puedan sustituir los componentes electrónicos del UAV por otras opciones disponibles, sin que el resto de programación, salvo la parte correspondiente del middleware, deba cambiar.

Teniendo en cuenta esto, los bloques funcionales definidos en el apartado 4.2.1 que se comunican con el hardware se estructuran en tres niveles, de forma análoga a como se muestra en la Figura 4.4. Este esquema constituye la base general que facilita la comprensión de la estructura desde el punto de vista de los niveles del código:

- En un nivel superior está el código escrito por el desarrollador, esto es la estrategia de coordinación y planificación de la misión, la parte del código de más alto nivel y que aporta la inteligencia al conjunto de UAV y la encargada de gestionar las acciones de cada uno de ellos (control, comunicación, seguridad y sensorización) en cada momento.
- El middleware o código intermedio, se ubica entre el software de bajo nivel y la estrategia de coordinación y que se encarga de ofrecer una interfaz común a esta última. De esta forma, el middleware permite controlar con

los mismos comandos UAV diferentes y realizar cambios en el hardware sin afectar a la estrategia de coordinación, modificando únicamente el middleware.

- El tercer nivel lo constituye el firmware o software de bajo nivel. Se trata de la parte del código más cercana al hardware y la encargada de realizar las medidas de los sensores, actuar sobre el control de potencia de los motores y las superficies de control.

4.3.2. Implementación del middleware

Entrando con mayor profundidad a las características que debe presentar este middleware con el objetivo de satisfacer los principios establecidas al principio de esta sección, se clasifica el hardware que puede formar parte de la instrumentación de un UAV en tres grandes grupos:

- El primero de los grupos lo forman los sensores y transductores y constituye la fuente de información al sistema, el puente de entrada entre la realidad física y el sistema del UAV. Estos dispositivos detectan las diferentes magnitudes físicas o químicas relevantes para los UAV y la misión a realizar con ellos, las transforman en variables eléctricas que se pueden medir como capacidad, resistencia, tensión, etc. y que son traducidas a valores digitales e integradas en el sistema.
- Los actuadores realizan la función inversa de los anteriores, conformando el puente de salida entre el sistema de control de los UAV y la realidad física. Un actuador es un dispositivo que transforma energía hidráulica, neumática o eléctrica en energía mecánica de movimiento para generar un efecto sobre el sistema que se pretende controlar. Los principales actuadores presentes en los UAV son los servomotores, que se encargan de variar las posiciones de las superficies de control y los motores de los rotores o hélices, que generan sustentación o desplazamiento de la aeronave.
- Finalmente la unidad de control, que es la encargada de coordinar y dirigir el sistema. Esta formada por una unidad de procesamiento que recoge e interpreta la información proveniente de los sensores y transductores, realiza los cálculos programados y genera las salidas para los actuadores. La intención última de la unidad de control es, como su propio nombre indica, controlar el sistema y conseguir que funcione siguiendo unos comportamientos definidos por el programador.

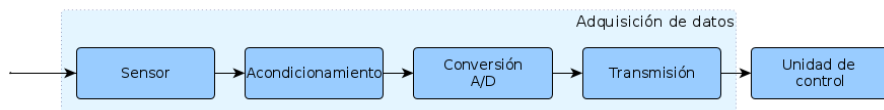


Figura 4.5: Esquema del proceso de adquisición de datos

4.3.2.1. Middleware para sensores

Los sensores y transductores, tal y como se ha explicado en la sección anterior constituyen el nexo de unión entre el mundo físico, inherentemente analógico y el sistema de control, que actualmente consiste en la enorme mayoría de los casos en una unidad de procesamiento digital.

Este proceso se realiza en varios pasos y su visión general aporta perspectiva sobre la lógica del middleware, ya que desde un punto de vista del funcionamiento de la estrategia de coordinación este proceso resulta transparente y únicamente el resultado, la medida de la magnitud es interesante. De ahí que tenga sentido la introducción de un middleware que aisle al desarrollador de estrategias de todo este proceso, lo que no debe en ningún caso disminuir la importancia del mismo, ya que de su buen funcionamiento depende todo el control del sistema.

Un esquema simplificado del funcionamiento del proceso es el que se muestra en la Figura 4.5:

El flujo de información procede del sensor, que como ya se ha explicado convierte la información física o química en variables eléctricas. En la fase de acondicionamiento de la señal estas variables eléctricas se amplifican, se filtran y se ajustan al rango de entrada para su conversión. A continuación, se produce la conversión de las variables analógicas a señales digitales, que se transmiten a la unidad de control, generalmente a través de un bus de comunicaciones y siguiendo un determinado protocolo.

Actualmente, la mayoría de sensores disponibles en el mercado ya incorporan el hardware y el firmware que realiza el acondicionamiento del mismo, la conversión A/D y suministran el dato medido a través de un bus de comunicación y con un protocolo. Este último paso es uno de los puntos en los que existe heterogeneidad, debido a que se emplean varios buses de comunicación y diversos protocolos. Para hacer frente a esto el middleware debe generar una capa de abstracción, que se encarga de ofrecer los datos provenientes de los sensores de una forma homogénea, sea cual sea el protocolo de comunicación empleado para su transmisión. De esta forma, el sistema de control no debe preocuparse de los protocolos empleados o su gestión, ya que la parte del middleware se encarga

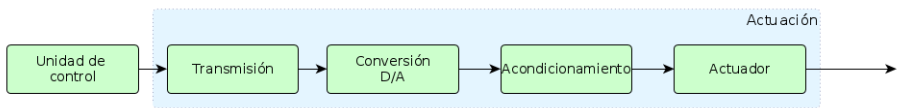


Figura 4.6: Esquema del proceso de actuación

de ello.

Una forma de conseguir el aislamiento es ofrecer a la unidad de control el valor de la magnitud medida en una variable, cuyo tipo varía según las capacidades del sistema empleado y el sensor. En el caso de sensores que únicamente tienen dos estados se puede emplear una variable binaria, mientras que en general se trata de variables enteras o flotantes, según las escalas de las magnitudes a medir. Dichas variables quedan accesibles mediante una lectura a una zona de memoria compartida o un método similar.

Asimismo, se debe implementar un método para lanzar los muestreos y adquisiciones de nuevos datos. En la mayoría de los casos se puede realizar un muestreo periódico, con una frecuencia determinada por el tipo de sensor y sobrescribir los valores existentes en la variable compartida. De esta forma, la unidad de control siempre dispone de la medida más reciente. En caso de que el tiempo de adquisición sea demasiado alto o interese medir la variable en un instante determinado se puede incorporar al middleware la capacidad de realizar un muestreo a petición.

El acceso a la variable y el control del muestreo, en caso de ser necesario, es la única interacción de la unidad de control con la parte de adquisición de datos y el resto de la configuración se realiza desde la parte del middleware.

4.3.2.2. Middleware para actuadores

Los actuadores constituyen la antítesis, desde un punto de vista funcional, a los sensores. Son los encargados de actuar sobre el mundo físico según los comandos recibidos de la unidad de control.

Al igual que en el caso de los sensores el proceso se compone de varios pasos y el poseer una visión general del funcionamiento ayuda a entender las funcionalidades del middleware. El proceso en este caso es el inverso, tal y como se puede ver en la Figura 4.6

Los comandos de la unidad de control son transmitidos al convertidor digital analógico. Este transforma estos comandos digitales en señales eléctricas analógicas que son amplificadas, filtradas y ajustadas al rango de funcionamiento del actuador (en el caso de actuadores digitales este paso no es necesario).

Finalmente el actuador transforma las señales recibidas en un movimiento, empleando para ello la energía suministrada por su alimentación, en el caso de los UAV generalmente eléctrica.

De nuevo el medio a través del cual se realiza la transmisión de los comandos es heterogéneo y existen diversos protocolos y buses de comunicación. El middleware ha de ocuparse de gestionar estos buses y transmitir el comando a través del mismo. De la misma forma que en el caso de los sensores, una zona de memoria compartida entre el middleware y la unidad de control parece la forma más sencilla de comunicar a ambos, abstrayendo la unidad de control de todo lo relacionado con el protocolo.

Al igual que en el caso de los sensores, el uso de un tipo de variable u otro depende principalmente del tipo de actuador. Así, en los actuadores de tipo todo o nada, lo óptimo es una variable binaria, mientras que en el resto se emplean enteros o flotantes según resulte más conveniente. En algunos casos el comando al actuador se envía de forma periódica, mientras que en otros puede requerirse un mayor control del momento preciso en el que se inicia el proceso de actuación. En estos casos el middleware proporciona un método que active o lance la ejecución del comando.

Nuevamente, de forma análoga a lo que sucedía con los sensores, el acceso a la variable para los comandos y el control de la actuación son la única interacción de la unidad de control con los actuadores. Del resto de tareas relacionados con los actuadores se ocupa el middleware.

4.3.2.3. Middleware para unidad de control

La unidad de control constituye la parte en la que resulta más complicado establecer el alcance del middleware, porque existe una gran variedad de unidades de control en el mercado actualmente y las funcionalidades que ofrece cada una de ellas difiere del resto.

El bloque funcional más estrechamente relacionado con la unidad de control es el de vuelo básico, en el se trata de definir una serie de comportamientos básicos para la aeronave: el retorno a casa, el mantenimiento de una posición, el cálculo de trayectorias desde el punto actual hasta un punto marcado como destino, así como otros que se consideren útiles como el mantenimiento de una altura o velocidad.

Partiendo de que los tres primeros comportamientos antes mencionados constituyen la base mínima se procede ahora a definirlos con mayor detalle para facilitar la comprensión del alcance del firmware y el middleware en este caso:

- En primer lugar se trata el retorno a casa, que constituye una característica

que incluyen multitud de unidades de control comerciales. Se trata de un sistema que, bajo un determinado criterio de activación, toma el control de la aeronave y la dirige de vuelta al punto de despegue, llegando en algunos casos incluso a aterrizar de forma autónoma. Se considera básico para una plataforma como la que se propone en este trabajo, dado que aporta seguridad frente a posibles fallos en el enlace de comunicaciones, así como problemas con la estrategia de coordinación que alejen las aeronaves más allá del alcance de los pilotos. Los criterios de activación suelen estar relacionados con un debilitamiento o pérdida de la señal de comunicaciones o una activación manual por parte del operador.

- El mantenimiento de una posición fija no es posible para todas las aeronaves, si bien se pueden realizar maniobras similares en todas ellas. Para los helicópteros o multihélices no presenta mayor problema, ya que pueden generar sustentación sin movimiento. Por el contrario, los UAV de ala fija necesitan mantener una velocidad con respecto al viento para generar sustentación. Para solucionar este problema con estos últimos se pueden establecer trayectorias circulares que mantengan una pequeña distancia alrededor del punto designado. Esta maniobra resulta útil, por ejemplo, para realizar la supervisión o vigilancia de un objetivo, una vez localizado.
- Finalmente, para conseguir que el UAV se desplace a los puntos designados resulta necesario realizar el cálculo de trayectorias desde la posición actual del UAV hasta el punto designado o destino. Estas trayectorias han de ser volables, esto es, realizables por el UAV con sus capacidades y limitaciones de forma estable y controlada. Además, han de cumplir los criterios que le impone el entorno, es decir, debe evitar tanto los obstáculos como las zonas donde el vuelo no esté permitido en su camino al punto de destino. Esto constituye la base, que se puede optimizar según diversos criterios de eficiencia en vuelo para obtener el menor consumo de energía y así maximizar la autonomía y el tiempo en vuelo.

Teniendo esto en cuenta, estas tres funciones básicas se incluye en el esquema de la Figura 4.4 dentro del apartado del firmware o software de bajo nivel, sobre estos esta el middleware que debe ofrecer al código de nivel superior el método para activar cada uno de ellos. Se puede entender cada uno de los comportamientos anteriormente descritos como modos de vuelo, dentro del bloque de vuelo básico y seleccionarlos mediante una variable. En lo que se refiere a la posición de destino se debe definir un formato para la posición y orientación deseadas y realizar en el middleware la traducción de la información del modo de vuelo deseado y el punto de destino al firmware de la unidad de control.

4.4. Incorporación iterativa y progresiva de la complejidad real

Los simuladores se han situado como una de las herramientas indispensables en la investigación a la hora de realizar nuevos desarrollos en numerosos campos, entre los que el trabajo con UAV no constituye una excepción. Mediante la simulación se accede a un número mayor de recursos a un coste mucho menor, ya que el coste de un ordenador capaz de ejecutar una simulación de un sistema es despreciable frente al coste de construir y desplegar el mismo sistema en la realidad. A esto se debe sumar el hecho de que el sistema simulado no tiene asociado los riesgos de rotura o daño que tiene el sistema real, ni las dificultades logísticas, de variabilidad en las condiciones del entorno, etc. Sin embargo, de la misma forma que han acelerado el desarrollo de sistemas de control, estrategias de coordinación y el diseño de nuevos productos los simuladores también acarrear un problema intrínseco: el reality gap.

El reality gap se define como la diferencia entre el simulador y la realidad. En el caso de las simulaciones de sistemas robóticos, como con los UAV, esta diferencia provoca, a su vez, una variación en la respuesta prevista del sistema. Así, el sistema de control diseñado y probado en el entorno de simulación se encuentra en la realidad con que el sistema responde de forma diferente, llevando el control a una situación diferente a la de las simulaciones y potencialmente no prevista en el diseño del mismo. Resulta complicado conocer a priori, el efecto de estas situaciones, lo que introduce una incertidumbre importante a la hora de realizar el paso del simulador a la realidad.

Cuanto más preciso sea el simulador, especialmente en los aspectos más relevantes para el sistema de control desarrollado, menor es la diferencia de comportamiento y, de forma análoga, menor el impacto del reality gap. Sin embargo, por muy preciso que sea el simulador, siempre existe una diferencia debido a la inmensa complejidad existente en la realidad. Además, definir la relevancia no siempre resulta sencillo y suele ser común el descubrimiento de problemas graves generados por parámetros que a priori se pueden considerar poco relevantes. En el caso concreto de los UAV, a diferencia de otros robots que desarrollan su labor en interiores, hay que tener en cuenta también la variabilidad del entorno exterior. Las condiciones de temperatura, humedad, la dirección e intensidad del viento, así como otras variables meteorológicas afectan al comportamiento de la aeronave y a sus capacidades.

Todas estas diferencias repercuten en la respuesta de la aeronave, mientras el sistema de control está ajustado para la respuesta producida en el simulador. Esto puede llevar a situaciones peligrosas como la pérdida de sustentación en el caso de UAV de ala fija. Este fenómeno se produce al aumentar el ángulo de

ataque del UAV y reducir su velocidad con respecto al viento por debajo de la velocidad de pérdida, a partir de ese momento se desequilibran las fuerzas sobre el UAV haciéndolo caer bruscamente. Este tipo de situaciones suelen requerir además de una lógica de control diferente a la habitual, con una respuesta más rápida, capaz de recuperar el vuelo estable y controlado. De la misma forma se pueden producir desvíos de la trayectoria prevista, aunque el UAV se mantenga en vuelo estable y controlado. Estos desvíos generan el riesgo de invadir zonas prohibidas o colisionar con otros vehículos u obstáculos presentes en el entorno del vuelo.

Todos los problemas anteriores suceden cuando se trabaja con un único UAV. Si, como en el caso del presente trabajo, se plantea hacerlo con un equipo de múltiples UAV se debe tener en cuenta varios problemas más. Uno de los más complejos es el efecto de los acoplamientos o interacciones existentes entre los UAV en el control del equipo. En el caso de un único UAV la diferencia en la respuesta del UAV real frente al simulado únicamente afecta a ese UAV, si se tienen múltiples UAV el efecto de la perturbación en uno de ellos puede generar respuestas sobre el resto, arrastrando todo el sistema a una situación no prevista y potencialmente peligrosa. Los problemas en las comunicaciones (retardos, mensajes perdidos, ruidos, etc.) así como en la sensorización también tiene el potencial de perturbar el funcionamiento del sistema. En general, al contar con varios UAV se aumenta la complejidad del sistema, introduciendo por tanto un mayor número de elementos hardware y software sometidos al reality gap, lo que lleva a un aumento en las probabilidades de respuestas no previstas en el sistema.

4.4.1. La realidad mixta

Para hacer frente a los problemas enumerados anteriormente y facilitar el paso del simulador a la realidad en este trabajo se propone la utilización de la realidad mixta, de forma que se pueda incorporar la complejidad de la realidad de forma progresiva. La realidad mixta consiste en combinar el mundo virtual con el mundo real, en tiempo real. Esta combinación permite crear nuevos espacios de trabajo, en los que interactúan tanto objetos y/o personas reales como virtuales.

Mediante la creación de una plataforma que emplee la realidad mixta, se propone la generación de un entorno virtual en el que coexistan los UAV reales y virtuales. Esta plataforma se desarrolla teniendo en cuenta una serie de premisas básicas:

- Facilitar el paso del simulador a la realidad, haciéndolo progresivo y suave: frente a la visión del salto del simulador a la realidad en un único paso

se pretende que las herramientas permitan realizar un acercamiento entre ambos entornos (real y virtual) mediante pequeños pasos incrementales en los que se va aumentando la complejidad realizados iterativamente sobre la plataforma de realidad mixta.

- Evitar la reescritura de código en el paso del simulador a la realidad: en muchos casos, aunque se mantenga la lógica de control no se implementa esta en el mismo lenguaje de programación en el simulador y en el sistema real. Este cambio de lenguaje genera un esfuerzo extra y una posible fuente de errores y discrepancias entre el control realizado en el simulador y el real. Para evitar esto el código desarrollado en el simulador debe poder trasladarse completamente al sistema real sin realizar modificaciones.
- Simplicidad y realismo: para minimizar el reality gap los modelos empleados han de ser precisos y se ha de tener especial cuidado en aquellos aspectos que más importancia tengan, evitando generar modelos complejos para fenómenos o efectos poco relevantes. Se debe tener siempre en cuenta que el fin último de los desarrollos realizados sobre la plataforma es la aplicación real de los mismos.
- Equivalencia entre UAV reales y virtuales. Se puede considerar un caso más de hardware heterogéneo, la plataforma debe tratar de forma indistinta a los UAV, sean estos reales o virtuales. Todos ellos deben tener presencia en el entorno virtual al mismo tiempo, colaborando para la realización de las misiones bajo la misma estrategia de coordinación.
- Flexibilidad y reutilización. Se pretende escapar del paradigma de estructura rígida sujeta a una serie de herramientas fijas, permitiendo y facilitando el uso de diferentes simuladores, diferentes UAV y favoreciendo la realización de diferentes misiones con el mayor reaprovechamiento posible de código entre ellas, reduciendo así el tiempo y esfuerzo de desarrollo.

El empleo de la realidad mixta trae consigo una serie de ventajas e inconvenientes. Dentro del primer grupo destaca el tratamiento homogéneo de los objetos, sean estos reales o virtuales y que encaja perfectamente con la visión de este proyecto. Si bien se han de tener en cuenta las diferentes capacidades de cada UAV, poder enviar comandos a todos ellos con una estructura común facilita el desarrollo de estrategias de coordinación. De esta forma se evita que el desarrollador deba tener en cuenta diversos protocolos para la comunicación con los UAV.

Otra ventaja es la reducción de los costes y los problemas logísticos unidos a la realización de pruebas reales con varios UAV. Fabricar, instrumentar y

realizar vuelos con múltiples UAV reales de forma simultánea tiene unos grandes requerimientos tanto materiales como humanos. Frente a esto, la realidad mixta permite generar fácilmente varios UAV virtuales, hasta donde alcance la potencia computacional del equipo o equipos empleados.

A diferencia de la simulación tradicional, permite la inclusión de UAV reales y su interacción con los virtuales, generando un paso intermedio antes de la prueba con todos los UAV reales. Esto permite una reducción del riesgo asociado al reality gap, especialmente en las primeras fases del paso del simulador a la realidad, evitando poner en peligro los equipos y las personas, a la vez que mantiene la estrategia lo más cercana posible a la realidad.

El principal inconveniente del uso de la realidad mixta es el desarrollo de la plataforma que permita trabajar con ella y el esfuerzo requerido para adaptarse a esta plataforma frente el uso de soluciones ad-hoc. Estos requerimientos se extienden tanto a la parte software como a la parte hardware, que también ha de cumplir una serie de requisitos para permitir el empleo de la misma según las premisas marcadas anteriormente. Por eso en el presente trabajo se mantiene el foco en la simplicidad y flexibilidad, ofreciendo una serie de principios básicos válidos para los trabajos con UAV. Dentro de esta idea, se prima la creación de un entorno de desarrollo que maximice la productividad a medio plazo mediante el reaprovechamiento del trabajo, la facilidad para la utilización de diferentes simuladores y UAV y la posibilidad de ampliación de la funcionalidades de forma sencilla.

En resumen, se pretende que la generación de la plataforma aporte las suficientes ventajas para superar los inconvenientes de realizar el desarrollo sujeto a las premisas indicadas y suponga una mejora tanto en la seguridad como en los tiempos empleados para el desarrollo de estrategias de coordinación con UAV.

4.4.2. Estrategia para la superación del reality gap

A la hora de realizar el paso de la simulación a la realidad se sigue la misma premisa, incrementando progresivamente la complejidad. Para seguir esa misma línea, en primer lugar se trabaja con un único UAV para generar el modelo y el control de vuelo básico. Este proceso ha de realizarse para los diferentes modelos de UAV. Una vez validados los modelos de UAV y completado el controlador de vuelo básico de todos ellos, se pasa a trabajar sobre varios UAV y a desarrollar la estrategia de coordinación. De forma más detallada:

Pruebas con un UAV El esquema del proceso que se debe realizar se muestra en la Figura 4.7 y comienza con el desarrollo de un modelo del UAV. Para ello se procede a recopilar datos característicos de la aeronave que se va a em-

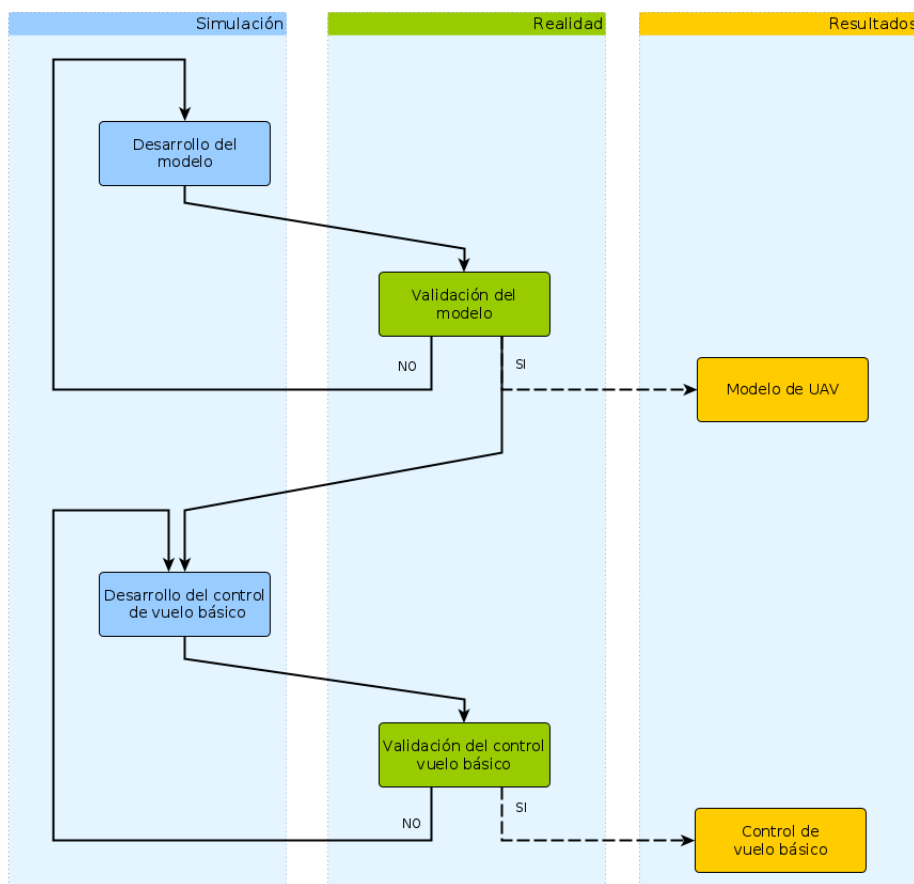


Figura 4.7: Esquema del proceso de desarrollo del modelo y el control de vuelo básico

plear e introducirlos en el simulador, que es el encargado de reproducir el modelo dinámico a partir de ellos.

Llegado a este punto se diseñan una serie de pruebas (vuelo sostenido, subida a plena potencia, descenso sin motor...) que se realizan con el modelo del simulador y con el UAV real. Dichos resultados se comparan y se subsanan las discrepancias detectadas, modificando el modelo en los aspectos necesarios para reducir esta diferencia al mínimo y ajustar así el modelo del UAV para que sea un fiel reflejo del real.

Una vez finalizado este proceso, el resultado es un modelo de UAV que, en las condiciones probadas, se comporta de forma muy cercana al UAV real. Este resultado constituye la base para el desarrollo del controlador de vuelo básico de la aeronave y que ha de realizarse de una forma similar:

En primera instancia se programa el controlador de vuelo básico y se prueba sobre el simulador, una vez que el desarrollo alcanza un nivel de estabilidad suficiente se realizan pruebas del controlador transferido al UAV real. Se analizan las diferencias entre ambos procediendo a mejorar y perfeccionar el control de vuelo básico hasta que funcione de forma semejante en el entorno real y en el simulado.

Todo el proceso anterior debe repetirse con cada uno de los tipos de UAV que se desee emplear en las misiones y se puede emplear para cada uno de ellos el simulador que se considere más adecuado.

Pruebas con varios UAV Este proceso parte del resultado generado en el anterior, así se supone que existe un controlador de vuelo básico para todos los UAV y un modelo simulado que se comporta de forma cercana al real. A partir de esta base se debe desarrollar el sistema de comunicaciones entre los UAV, de forma que se pueda intercambiar información entre los UAV. Esta comunicación debe extenderse también a los UAV reales, de forma que la información se transmita entre UAV reales y virtuales de forma indistinta.

De la misma forma se debe implementar también todo lo necesario para la aplicación que se quiere realizar sobre la plataforma. Esto incluye simular la sensorización y los fenómenos que se desea estudiar, de forma que el entorno simulado constituya una representación cercana al entorno en el que desarrollan su labor los UAV reales. Una vez alcanzado este punto se comienza a desarrollar y depurar la estrategia de coordinación sobre los UAV virtuales.

Cuando la estrategia desarrollada alcanza un nivel de estabilidad y madurez suficiente se procede a introducir un UAV real en el entorno de realidad mixta. Este UAV real envía su información en tiempo real al entorno virtual, compartido con los UAV virtuales. A diferencia de lo que pasaría si se trasladase la estrategia a los UAV reales en una única etapa, de esta forma exclusivamente es necesario fijarnos en un UAV real, lo que aumenta la seguridad frente a la idea de lanzar varios UAV con una estrategia que únicamente ha sido probada en el simulador. En base a los resultados de las pruebas se procede a reducir las diferencias que surjan entre el UAV real y los UAV virtuales.

Basándose en la plataforma de realidad mixta se procede a repetir las pruebas añadiendo progresivamente un mayor número de UAV reales al equipo, depurando y ajustando las discrepancias surgidas. Como se ha probado con anterioridad un UAV real con la estrategia se reduce considerablemente el riesgo de fallos en el proceso. En cualquier caso, resulta indispensable en estas fases de pruebas contar con un piloto por cada aeronave en vuelo, de forma que si se produce algún problema pueda tomar el control de la misma y minimizar el riesgo de accidente.

A priori puede considerarse que el final de este proceso se alcanza cuando todos los UAV del equipo son UAV reales y en ocasiones sucede así; sin embargo en determinadas ocasiones puede interesar mantener UAV virtuales en la aplicación final. Esto se explica porque mediante el uso de UAV virtuales se puede aportar información a la estrategia de coordinación basada en modelos de los fenómenos que se desea estudiar o en datos recopilados en pruebas anteriores, lo que contribuye de nuevo a potenciar los beneficios del uso de la plataforma propuesta en el presente trabajo.

4.5. Conclusiones

En este capítulo se ha realizado una recopilación de los principios básicos de la metodología en los que se basa la aproximación para el desarrollo de estrategias de coordinación que se propone en el presente trabajo. Así, se pretende que la plataforma sea sencilla y flexible, aportando una estructura modular que ayude al desarrollo del software de forma estructurada y que permita la reutilización. De la misma forma, se propone el empleo de un middleware que abstraiga a la estrategia de coordinación la heterogeneidad del hardware, facilitando así el desarrollo de la misma. Finalmente y con la finalidad de aumentar la seguridad y facilitar la implementación de la estrategia en la realidad se propone el empleo de la realidad mixta y la introducción de la complejidad de forma progresiva e iterativa.

Todos estos esfuerzos van encaminados a facilitar el desarrollo de estrategias de coordinación de forma más rápida y segura. Para demostrar el potencial de las propuestas realizadas y completar la aproximación central de este trabajo se ha desarrollado una plataforma software compatible con esta metodología y que se describe con detalle en el Capítulo 5, así como varios ejemplos pequeños, obtenidos de la experiencia en el uso de la plataforma, en los que se pueden ver las ventajas aportadas por esta aproximación.

Capítulo 5

Diseño y desarrollo de la plataforma software

5.1. Introducción

En el capítulo anterior se aportó una visión general de la metodología propuesta en el presente trabajo, justificando los aspectos principales de la misma y cómo ayudan a afrontar los principales retos presentes a la hora de desarrollar estrategias de coordinación. Para completar la aproximación propuesta en el presente trabajo es necesario el desarrollo de una plataforma software que permita aplicar los principios básicos descritos en la metodología. Este capítulo se centra en la plataforma que se ha desarrollado para ello, describiendo con mayor detalle sus aportaciones y las principales decisiones de diseño adoptadas para su desarrollo. También se incluye un ejemplo obtenido para cada uno de los aspectos de la metodología, obtenido del uso de la plataforma para las aplicaciones desarrolladas con la misma.

Para comenzar, resulta imprescindible aclarar el concepto de la plataforma. En este trabajo se considera la plataforma el conjunto de las herramientas y software desarrollado que permite la aplicación de la metodología propuesta. Tratando de realizar una analogía con la edificación, se trata de los cimientos de un edificio, que constituyen la base necesaria para construirlo y se pueden realizar con diferentes materiales. De la misma forma, para el desarrollo de la plataforma se pueden seguir diversos caminos y primar algunos aspectos frente a otros. En este sentido se ha optado por mantener varias de las premisas comunes ya descritas en el capítulo anterior.

Dado que, para hablar de la plataforma y sus características resulta nece-

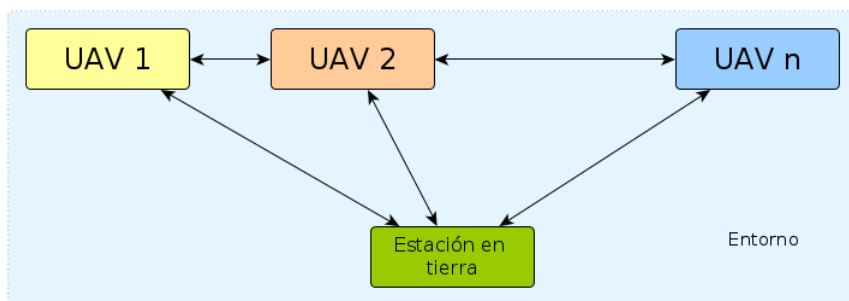


Figura 5.1: Paradigma de uso de la plataforma

sario concretar los principios generales propuestos en la metodología, conviene aclarar el paradigma de uso de las estrategias de coordinación, que se muestra en la Figura 5.1. En ella se representan también los flujos de información y los tres elementos fundamentales en el sistema:

- Los UAV: conforman la principal fuente de información al sistema y simultáneamente la principal actuación del mismo. Dentro del marco de este trabajo se considera siempre al UAV como una entidad con autonomía, esto es, no se trata únicamente de las aeronaves, sino también el sistema de control capaz de mantenerlas en vuelo y de la inteligencia artificial en el caso de estrategias distribuidas. Siguiendo la metodología propuesta, estos UAV pueden ser heterogéneos, llegando al punto de tratarse de UAV completamente simulados.
- La estación de tierra: el nexo de unión entre los UAV y la interfaz con la que se comunican los operadores con el sistema. En el caso de estrategias centralizadas es también la encargada de la estrategia de coordinación y, en todos ellos, la parte que realiza la recopilación de datos y la que permite la visualización y supervisión de la evolución del sistema.
- El entorno o medio en el que se desarrolla la misión: a diferencia de las misiones en las que no se trabaja con realidad mixta, hay dos entornos: Un entorno físico real, sobre el que no se puede actuar y un entorno virtual dinámico y que se ajusta según lo requerido en las misiones.

Así, la plataforma se conforma tanto por el software desarrollado en la estación de tierra como en los UAV y que se realiza siguiendo la estructura propuesta en la metodología y descrita en la sección 4.2.1. Este software incluye también la generación del entorno de realidad mixta sobre el que se desplazan todos los UAV.

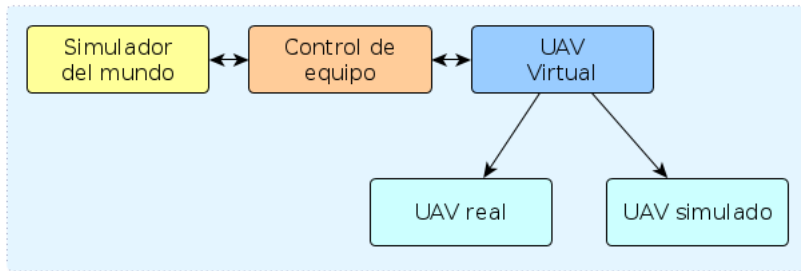


Figura 5.2: Estructura de la plataforma

5.1.1. Visión general

Teniendo en cuenta todo lo anterior, se ha diseñado la plataforma como un sistema multiagente, en el que cada elemento es un agente que se puede ejecutar en diferentes ordenadores, situados incluso en diferentes lugares. Además, cada elemento proporciona sus propios puntos de entrada para los datos que vienen del mundo real y que permiten ajustar su comportamiento a las condiciones y limitaciones de la realidad.

Así, desde el punto de vista del control se puede dividir el sistema en tres elementos principales o tipos de agentes: el control de equipo, el simulador del mundo, y los UAV virtuales. El resultado es el que se muestra en la Figura 5.2. Todos los elementos o agentes pueden ser distribuidos, y se comunican entre ellos utilizando un protocolo binario desarrollado sobre UDP.

De forma general se puede realizar una analogía entre los elementos de la Figura 5.1 y los propuestos en la plataforma:

- El UAV virtual: se corresponde con el conjunto de una aeronave y la electrónica asociada al mismo. El UAV virtual puede corresponderse indistintamente con UAV simulado o UAV real, así como otras combinaciones de ambos tales como pruebas «Hardware in the Loop» (HIL).
- El control del equipo: resulta casi análogo a la estación en tierra, está formado por las herramientas de visualización y la parte centralizada de la estrategia de coordinación.
- El simulador del mundo: se corresponde con el entorno virtual, es dinámico y permite la realimentación (en tiempo real o en diferido) de datos obtenidos en vuelos reales para ajustarse de forma precisa al entorno real.

A lo largo de este capítulo se ilustra de forma progresiva cómo se construye la plataforma que constituye la base de la aproximación propuesta:

- En primer lugar, en la sección 5.2 se describe la implementación de los bloques funcionales de los elementos que componen la plataforma, siguiendo la metodología propuesta en la sección 4.2.1. También se incluye un ejemplo que muestra las potenciales ventajas del uso de la estructura normalizada y modular.
- En la sección 5.3 se profundiza en los aspectos relativos a la abstracción e independencia de la estrategia del hardware concreto y se incorpora también un ejemplo demostrativo que pone de manifiesto las posibilidades aportadas por el uso de la metodología.
- Finalmente la sección 5.4 se centra en la incorporación iterativa de la complejidad real, describiendo la forma en la que se combinan los elementos reales y simulados mediante el uso de la realidad mixta. Dentro de la misma también se detalla un ejemplo de aplicación en el que se mejora con datos reales el modelado de uno de los componentes.

5.2. Estructura normalizada y modular para el software

El código de la plataforma se estructura en bloques funcionales, siguiendo lo establecido en la sección 4.2.1 tanto para la estación de tierra como para los UAV. Además, para permitir el uso de la realidad mixta se incluye la generación del entorno virtual mediante el simulador del mundo, que se trata con mayor detalle en la sección 5.4.

Se detalla la estructura de los tres elementos por separado para aclarar el alcance de cada uno de ellos:

5.2.1. UAV virtuales

Los UAV virtuales representan los UAV desplegados en el sistema, cada UAV virtual se corresponde con un UAV real, simulado o un estado intermedio entre ambos como el que se produce cuando se realizan simulaciones en las que se incluye el hardware real (HIL), se simulan sensores, etc.

Dado que uno de los principios de diseño de la plataforma es el de tratar de forma homogénea a los UAV, sin tener en cuenta sus características físicas, el desarrollo de los bloques funcionales para ambos es prácticamente el mismo, siguiendo lo descrito en la Figura 5.3. La principal diferencia entre los UAV virtuales radica en la parte del hardware, es decir, en la aeronave y su comportamiento dinámico. En la sección 5.3 se trata con mayor detalle esta diferencia,

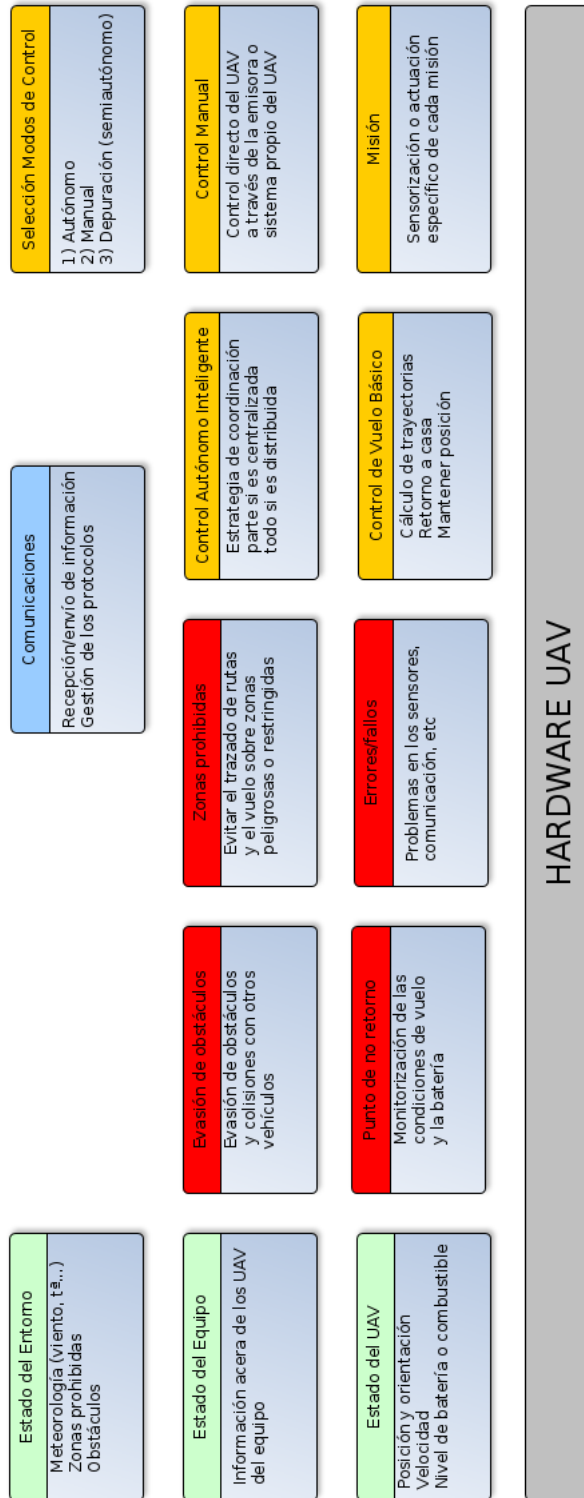


Figura 5.3: Bloques funcionales para el software de los UAV

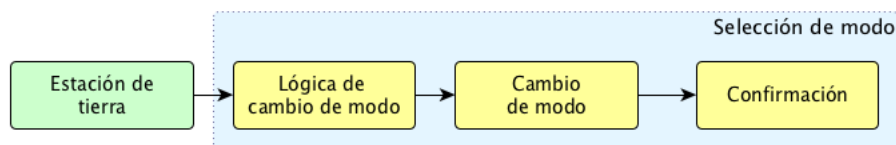


Figura 5.4: Proceso de selección del modo de control

mientras esta sección se centra en el software.

5.2.1.1. Control

Desde el punto de vista del control, hay cinco bloques funcionales, tal y como se puede ver en la Figura 5.3.

Selección de modo En primer lugar existe un bloque encargado de la selección del modo de control del UAV, que es controlado desde la estación de tierra. Los diferentes modos y submodos de control se definen aquí y las condiciones de activación de cada uno de ellos en la estación de tierra. Inicialmente, deberían definirse al menos dos modos, uno completamente manual en el que el operador toma el mando directo de la aeronave y otro completamente autónomo en el que el control se realiza mediante las órdenes del control autónomo inteligente, que se calculan en función de los objetivos de la misión. Además, para la realización de las pruebas iniciales y el ajuste de los controladores de vuelo resulta recomendable definir un estado intermedio o de depuración, en el que se maximice la toma de datos y se controle el UAV dentro de unos límites más estrictos para garantizar la seguridad. También se pueden definir modos manuales asistidos para determinadas maniobras complicadas.

Se trata de un bloque que no alberga demasiada complejidad, aunque resulta crítico para el correcto funcionamiento del sistema. Consta de tres partes básicas, tal y como puede verse en la Figura 5.4:

- La lógica de cambio de modo, es la encargada de interpretar los mensajes de cambio de modo provenientes de la estación en tierra y de comprobar si es posible realizar el cambio solicitado. En caso de que no sea el caso se puede establecer el protocolo a seguir, generalmente responder con un mensaje de error y su causa a la estación en tierra. En caso afirmativo se continúa el proceso.
- La parte en la que se pasa de forma efectiva de un modo de control a otro. Una vez asegurado en el paso anterior que el nuevo modo solicitado es

válido se toman las medidas necesarias para poder realizar el cambio con seguridad. En general se requiere una inicialización del sistema a valores adecuados para que la transición sea lo más suave posible.

- Una vez finalizada la transición resulta recomendable generar un método para confirmar el éxito del cambio de modo al origen de la solicitud. Esta realimentación se puede realizar de diversas formas, ya sea mediante un mensaje de confirmación, mediante indicaciones visuales sobre el propio UAV, etc.

Control autónomo inteligente El comportamiento autónomo de los UAV virtuales reside en el control autónomo inteligente, que es el encargado de decidir cual es la acción siguiente a realizar.

De forma general se compone de tres partes, que no necesariamente deben implementarse en todos los casos:

- La toma de datos, en la que se recopila toda la información relevante para la estrategia de coordinación. Dependiendo de la estrategia empleada se toman datos del estado del UAV, de otros UAV, del entorno, etc. Si la misión requiere el uso de sensorización específica, el valor, ya procesado por el bloque de misión, también se incorpora en este punto.
- El proceso de decisión, que constituye la inteligencia de la estrategia de coordinación. A partir de todos los datos tomados en el paso anterior y según los criterios definidos por el desarrollador de estrategias se calculan las actuaciones o salidas del sistema.
- Finalmente existe un proceso de asignación y transmisión de las salidas, que constituyen las consignas para los controladores de los UAV. Resulta imprescindible para el funcionamiento óptimo del sistema la transmisión efectiva de las decisiones para permitir su implementación.

La complejidad de cada una de las partes viene determinada principalmente por el tipo de estrategia de coordinación: En el caso de estrategias completamente centralizadas únicamente resulta necesario implementar la última parte, ya que el controlador central se encarga de la toma de datos y la toma de decisiones. Por contra, si se trabaja con estrategias distribuidas las dos primeras partes reciben la mayor parte del contenido, ya que en este caso cada uno de los UAV calcula sus consignas.

En este bloque y en su homólogo de la estación en tierra es donde se centra la mayor parte de los esfuerzos y la aportación del desarrollador de estrategias de coordinación.

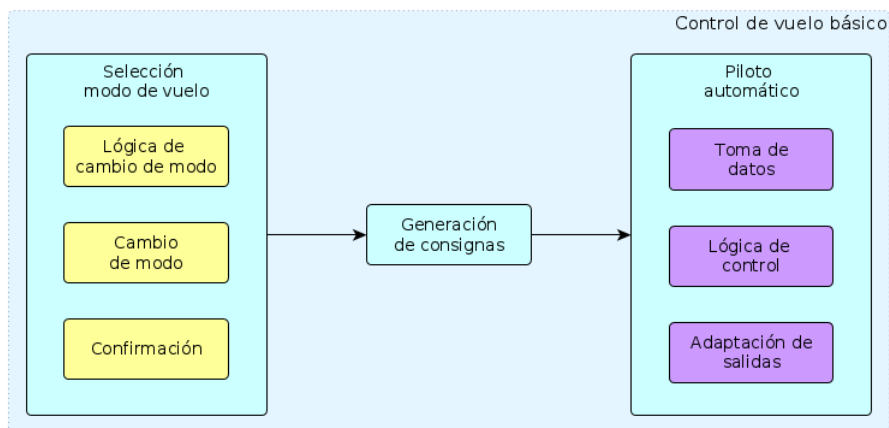


Figura 5.5: Estructura del control de vuelo básico

Control de vuelo básico Constituye la parte de más bajo nivel del control autónomo, y es una extensión del piloto automático para permitir funciones más avanzadas o modos de vuelo tales como el cálculo de trayectorias, el retorno autónomo a la zona de despegue o “vuelta a casa” y el mantenimiento de la posición. Su principal función es traducir los comandos del control autónomo inteligente a posiciones de las superficies de control del vehículo y a la regulación de la potencia de los motores, manteniendo la aeronave en vuelo estable y controlado.

Se puede dividir en tres partes:

- La selección del modo de vuelo: se puede subdividir de la misma forma que el bloque de selección de modo de control, ya que la lógica del proceso es muy similar:
 - En primer lugar se analiza si es posible realizar el cambio al modo deseado, en caso negativo se informa al sistema del error y en caso afirmativo se continúa el proceso.
 - A continuación se inicializa todo lo necesario y se procede a realizar la transición al nuevo modo.
 - Una vez finalizado el paso al nuevo modo de vuelo, se confirma el resultado satisfactorio.
- La generación y adaptación de las consignas para el piloto automático: estas consignas vienen determinadas por el modo de vuelo en el que se encuentre el UAV, los parámetros recibidos del control autónomo inteligente

y, en algunos casos, el estado del UAV. Las consignas se calculan en este paso y se limitan, según las capacidades del UAV y ajustan a la escala adecuada para su correcta interpretación por parte del piloto automático.

- El piloto automático: que a su vez puede subdividirse en tres partes, de forma análoga a la estrategia de coordinación:
 - La toma de datos del UAV: en este caso únicamente se centra en los datos de estado del propio UAV.
 - La lógica de control: es la parte del software encargada de mantener al UAV lo más cerca posible de las consignas obtenidas en el paso anterior. Para ello, se basa en tomar decisiones en función de las consignas y el estado actual del UAV.
 - La adaptación de las salidas: se ajustan a la escala adecuada para los actuadores y se transmiten a los controladores de los mismos para conseguir el efecto deseado.

Misión El bloque de misión condensa todo aquello relacionado con la aplicación concreta que se pretende realizar con el equipo de UAV y su complejidad depende en gran medida del hardware que se emplee en ella. Así, si se pretende realizar captura y procesamiento de imágenes, el código para realizar todo el proceso se ubicaría dentro de este bloque. Comprende tanto la lectura como el filtrado de la sensorización específica o, en caso de misiones en las que se requiera, también la actuación. Se puede definir una estructura general con las siguientes partes:

- Sensorización: en la que se realizan todas las acciones necesarias para la toma de información con los sensores de misión. Puede subdividirse en dos apartados:
 - Por una parte la gestión de los buses de comunicación y protocolos empleados con los sensores.
 - Por otra, el procesamiento, filtrado y adaptación de los datos, dando lugar al resultado que se emplea en el control autónomo inteligente.
- Actuación: obtiene los comandos provenientes del control autónomo inteligente y realiza las gestiones pertinentes para que se cumplan. Se puede subdividir en dos apartados al igual que en el caso anterior:
 - Por una parte existe una lógica que compruebe que la operación solicitada del actuador se puede realizar de forma segura, así como un filtrado de los parámetros de entrada a valores válidos.

- Finalmente, a partir del resultado del apartado anterior, se lleva a cabo la actuación solicitada.

Control manual Se trata de uno de los bloques funcionales de más bajo nivel y se encarga de recibir y transmitir a los elementos de control la señal proveniente del operador. Se trata de un modo que busca garantizar la seguridad en caso de que se produzca algún problema con el control autónomo o se esté depurando y trabajando todavía en el ajuste del control de vuelo básico.

Dada su aspiración de modo de control seguro y fiable suele tratarse de un bloque en el que el software interfiere muy poco, de forma que un error en el código no provoque fallos en el mismo. Generalmente, una vez activado el modo manual únicamente se leen los datos provenientes de la estación de tierra y se envían directamente a los elementos de control correspondientes.

5.2.1.2. Estado

En lo referente a la información de estado, se cuenta con tres bloques funcionales diferentes, tal y como se muestra en la Figura 5.3:

Estado del entorno El estado del entorno o ambiente en el que se desarrolla la misión recoge todos los datos referentes a las características relevantes del mismo para la actividad a realizar. Comprende dos apartados:

- La adquisición de los datos, ya sea a través de sensores incorporados en el propio UAV o a través de la transmisión de datos provenientes de otros UAV o de la estación de tierra. Debe gestionar el refresco de los mismos de forma periódica para disponer de información actualizada.
- El procesamiento, filtrado y almacenamiento de los datos tomados en el apartado anterior que así lo requieran. El resultado de todo este proceso queda a disposición del resto de bloques funcionales.

Dada la importancia de los cambios que provocan en el medio aéreo las condiciones meteorológicas, estas constituyen un porcentaje importante de la información a tener en cuenta en este bloque, ya que las variaciones afectan directamente al desempeño y comportamiento de los UAV. Asimismo, en este apartado se definen las zonas prohibidas o peligrosas, así como aquellos obstáculos de los que se tenga constancia para evitarlos.

Estado del equipo En este bloque se condensa toda la información sobre los UAV de la que se dispone, incluyendo la del propio UAV. Resulta importante

disponer de información como la posición y objetivos del resto de UAV para que la estrategia generada evite los solapamientos y asigne los nuevos objetivos a los UAV de la forma más eficiente posible.

Su estructura es prácticamente la misma que la del apartado anterior, por una parte se ocupa de la adquisición de los datos y por otra de su procesamiento y filtrado, dejando el resultado a disposición de los bloques funcionales que lo requieran.

Generalmente el control autónomo inteligente es el principal candidato a requerir esta información. Así, los parámetros con mayor importancia son los valores de los sensores de misión, la posición y la orientación de los UAV, ya que son los requeridos para este bloque.

Estado del UAV A diferencia de los dos bloques anteriores, cuya información suele ser una combinación de información del propio UAV e información externa, en la mayor parte de los casos el estado del propio UAV se obtiene de la lectura de los sensores embarcados (IMU, pitot, altímetro...).

En lo referente a la estructura, sin embargo, mantiene el mismo esquema que en los dos bloques anteriores, con una parte que se encarga de la toma de datos y gestión de los buses de comunicación y otra que procesa y filtra los datos para que el resto de bloques pueda utilizarla.

La información proveniente de este proceso se emplea también para el control de vuelo básico así como para su transmisión a través de la red de comunicaciones, con la finalidad de informar al resto de UAV del equipo y proceder de forma informada a la toma de decisiones.

Además de los datos de posición y orientación, resulta especialmente importante para el caso de misiones reales el nivel de batería o combustible, tanto para tomar decisiones estratégicas al respecto, como para garantizar la capacidad de los UAV de alcanzar el punto de aterrizaje previsto.

5.2.1.3. Seguridad

La seguridad comprende cuatro bloques funcionales, tal y como se puede ver en la Figura 5.3. Dado que en se puede dar el caso de que varias situaciones de peligro se produzcan simultáneamente, es necesario establecer niveles de prioridad para estas situaciones. De esta forma se pueda atender con mayor presteza las situaciones más urgentes.

Evasión de colisiones Se trata de un bloque funcional que pretende evitar colisiones con obstáculos tanto de obstáculos físicos estáticos (edificios, árboles,

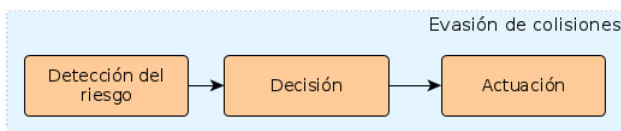


Figura 5.6: Estructura del bloque de evasión de colisiones

cables de tendido eléctrico...), como obstáculos móviles (tráfico rodado) o de otros vehículos aéreos en movimiento en la misma zona. Para ello consta de tres partes:

- En primer lugar, la detección: el primer paso para conseguir evitar las colisiones consiste en ser capaz de detectar el peligro con anticipación suficiente. Para esto, es necesario equipar al UAV con la sensorización necesaria y monitorizarla conjuntamente con la información sobre obstáculos que pueda obtenerse del resto de UAV del equipo.
- En el proceso de decisión: una vez detectado un riesgo de colisión en el primer apartado, se ha de proceder a decidir las actuaciones más apropiadas disponibles. Para esto se ha de tener en cuenta el estado actual del UAV y la información de que se dispone acerca del obstáculo. El resultado de este apartado es una serie de actuaciones sobre el UAV con la intención de minimizar el riesgo y, en última instancia, los daños sobre el UAV.
- Finalmente, la actuación: una vez determinadas las acciones a emprender, se envían los comandos correspondientes al control de vuelo básico para que sean ejecutados.

Evasión de zonas prohibidas Otro aspecto a tener en cuenta de cara a la seguridad es el mantenimiento de las aeronaves en la zona de vuelo designada, evitando la entrada en zonas prohibidas o peligrosas, con los riesgos que ello acarrea, tanto para los propios UAV como para otros vehículos aéreos. La subdivisión de este bloque es similar a la anterior:

- La monitorización y detección del riesgo de entrada en una zona prohibida es la primera parte del proceso. Para ello se debe disponer de información acerca de las zonas sobre las que no se permite el vuelo y se puede calcular la trayectoria del UAV, en función de su posición y orientación actual. En el caso de que se detecte un riesgo de entrada en una de estas zonas se pasa al siguiente paso del proceso.
- La toma de decisiones: para definir la actuación óptima que mantenga al

UAV en la zona de vuelo designada. Generalmente consiste en un cambio en la trayectoria prevista que tenga en cuenta la zona restringida.

- La última parte del proceso consiste en la transmisión de la trayectoria corregida al control de vuelo básico, para que la ejecute.

Punto de no retorno El cálculo de la energía necesaria para realizar el retorno a la zona prevista para el aterrizaje constituye también una parte importante dentro de la seguridad en la operación de las aeronaves. Resulta primordial que los UAV se mantengan en un radio que permita el retorno seguro y controlado de los mismos. Para ello se sigue el siguiente proceso:

- En primer lugar, de forma similar a los casos anteriores, se ha de vigilar la autonomía en vuelo. Para ello, si se emplea la propulsión eléctrica, se comprueba la diferencia de tensión de la batería y, si se emplea otro combustible, el nivel del mismo en el depósito.
- En base a los datos obtenidos en el apartado anterior se calcula la autonomía disponible y, teniendo en cuenta la distancia a la zona de aterrizaje y las condiciones actuales del vuelo, se estima la necesaria para alcanzar la zona de aterrizaje.
- En caso de detectar el acercamiento al punto de no retorno se generan las acciones correctoras más adecuadas, que pueden ir desde advertir al operador de la situación a volar automáticamente al punto de aterrizaje designado.

Fallos o averías Se trata de un bloque funcional cuya implementación depende en gran medida del tipo de UAV empleado y de la instrumentación de que se disponga a bordo del mismo. Se centra en los casos de mal funcionamiento de alguno de los componentes del UAV. De forma general se puede realizar una división en:

- La vigilancia de los componentes: dado que en la mayoría de los casos no existe un sensor que detecte las averías en este paso generalmente se comprueba que el sistema funciona de forma correcta siguiendo unas determinadas reglas básicas. En caso de que se detecte un riesgo de fallo se pasa a la siguiente etapa.
- La toma de decisiones: en la que en función de los datos tomados en la primera etapa se decide cual es la actuación más conveniente. Hay averías de mayor o menor gravedad o incluso falsas alarmas sobre las mismas, por lo que la actuación óptima para todas ellas dista de ser la misma.

- La actuación: una vez dirimida la actuación considerada óptima se procede con la ejecución de la misma. Las actuaciones previstas pueden ser alarmas al operador del riesgo detectado u acciones correctivas sobre el sistema que permitan mantener la seguridad en la operación de los UAV.

Dado que los UAV y la instrumentación son sistemas complejos existen multitud de subsistemas implicados en el funcionamiento y no resulta posible evaluar en tiempo real el buen funcionamiento de todos ellos. Sin embargo, es muy recomendable la implementación de un sistema de detección de fallos en los sistemas críticos para la seguridad de la aeronave así como las actuaciones previstas en caso de producirse. Dentro de estos sistemas se pueden incluir los motores, las comunicaciones y los sensores que informan del estado del UAV.

Emprender acciones preventivas y correctivas ante fallos de estos subsistemas puede suponer la diferencia entre un grave riesgo tanto para el propio UAV como el resto de elementos del entorno a una operación segura o una minimización del mismo.

5.2.1.4. Comunicación

La comunicación se incluye como un único bloque funcional que se encarga de proveer al resto de bloques de la posibilidad de intercambiar información, abstrayéndolos de las problemáticas o limitaciones del funcionamiento concreto de la tecnologías de comunicación existente para la transmisión de datos. El funcionamiento básico de la plataforma requiere de la existencia de al menos un enlace de comunicaciones inalámbrico entre los UAV y de estos con la estación en tierra. Se puede añadir más enlaces para redundancia o para transmisión de datos específicos, en cualquier caso este bloque se puede dividir en dos partes:

- Por un lado está la gestión de los enlaces de comunicación existentes. Dado que se supone un enlace bidireccional se ha de ocupar de seguir el procedimiento necesario para llevar a cabo la transmisión de los mensajes que se le soliciten como de la lectura de los mensajes entrantes, incluyendo la comprobación de la integridad de los mismos mediante el método definido en el protocolo de comunicaciones.
- Por el otro está la gestión de los mensajes, que a su vez se puede dividir en dos, la entrada y la salida:
 - En lo referente a los mensajes salientes, es frecuente encontrarse con varios mensajes pendientes de envío, ya que la limitación del ancho de banda hace imposible enviar los datos de forma instantánea. El bloque de comunicaciones ha de almacenar los mensajes pendientes

de envío, gestionando la cola de envíos pendientes, incluyendo incluso sistemas de prioridad para los mensajes.

- Con respecto a la entrada, se ha de ocupar del direccionamiento de los mensajes recibidos al bloque funcional correspondiente. Este direccionamiento ha de definirse mediante algún parámetro o código desde el emisor como puede ser el tipo de mensaje.

5.2.2. Estación de tierra

La estación de tierra es el punto central para las tareas de control, monitorización y visualización del equipo de UAV. Es la principal herramienta del operador para interactuar con los UAV, supervisar el correcto desarrollo de las misiones y actuar en caso de que surja algún problema. El software de la estación de tierra se desarrolla siguiendo el esquema modular mostrado en la Figura 5.7

5.2.2.1. Control

Comenzando por la parte de control, hay en este caso tres bloques funcionales, tal y como se puede ver en la Figura 5.7.

Selección del modo de control Se trata del bloque complementario del programado en los UAV. Se puede dividir en tres partes:

- Por un lado está la monitorización del modo deseado, que puede venir dado por una introducción de datos al sistema (mediante palancas, botones, etc) o por el propio programa.
- Cuando se detecta que se ha de cambiar el modo de control se procede a transmitir el mensaje al UAV para que realice el cambio.
- Finalmente se espera la confirmación del nuevo modo de control por parte del UAV con el fin de prevenir que el operador considere el UAV en un modo de control diferente al designado.

Control autónomo inteligente Un bloque homónimo al del UAV, con la misma función y estructura y cuya complejidad se fija según el tipo de estrategia de coordinación, de forma inversa. Así, en el caso de estrategias centralizadas, este bloque es el encargado de traducir los objetivos de la misión a comandos para los UAV y si se trata de una estrategia distribuida, la inteligencia se implementa sobre los UAV, dejando este bloque vacío.

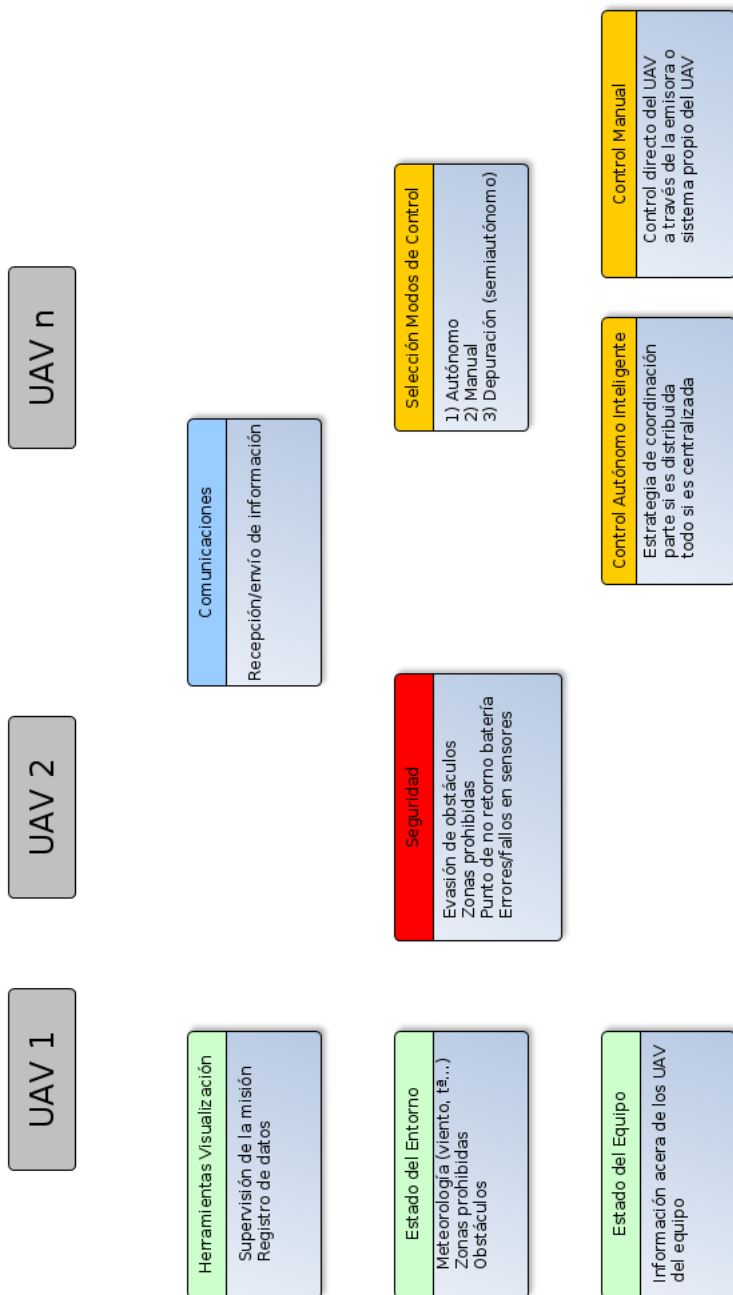


Figura 5.7: Bloques funcionales para el software de la estación de tierra

Control manual Es el encargado de transmitir a los UAV la señal de control proveniente de los joysticks, palancas o similar que emplee el operador para controlarlos. Se trata de un modo que busca garantizar la seguridad en caso de que se produzca algún problema con el control autónomo y no suele contener un desarrollo software demasiado complejo para evitar la aparición de fallos en el mismo.

5.2.2.2. Estado

En lo referente a la información de estado, se cuenta con tres bloques funcionales, tal y como se muestra en la Figura 5.7:

Tanto el estado del entorno con el del equipo de UAV mantienen la estructura y funcionalidad descrita anteriormente para el UAV, por lo que se omite para evitar alargar innecesariamente el presente documento.

Herramientas de visualización En este bloque se combinan las herramientas de visualización y registro de datos. Se puede dividir en dos partes:

- En primer lugar la visualización de los datos: se encarga de tomar los datos relevantes de la información disponible, tanto del estado de los UAV como del entorno y facilitárselos al operador, favoreciendo la comprensión por parte de este. Por ejemplo, en la plataforma se ha incluido una visualización generada en tiempo real de la posición de los UAV del equipo y sus trayectorias empleando Google Maps.
- En segundo lugar, el archivo de información o datos significativos: la generación de este tipo de registros sirve para: permitir la revisión de misiones realizadas con la intención de depurar fallos en el sistema, procesar los datos para la obtención estadísticas sobre los UAV y la misión, etc.

5.2.2.3. Seguridad

Los bloques funcionales de seguridad serían los mismos que los ya descritos para el UAV. No se recomienda hacerlo de esta forma ya que se pueden producir fallos en el enlace de comunicaciones, además habría que hacerlo para todos los bloques y cada uno de los UAV, con lo que el tiempo de respuesta y el coste computacional para la estación en tierra son mucho mayores frente a la opción de estar implementado sobre el propio UAV.

En este bloque se pueden incluir comportamientos de seguridad de los UAV en grupo, tales como una vuelta a casa en caso de empeoramiento de las condiciones meteorológicas.

5.2.2.4. Comunicación

La comunicación se incluye como un único bloque funcional que se encarga de proveer al resto de bloques de la posibilidad de intercambiar información, abstrayendo al resto de bloques funcionales de las problemáticas o limitaciones del funcionamiento concreto de la tecnologías de comunicación existente para la transmisión de datos. Tiene la misma estructura que en el caso del UAV.

5.2.3. Simulador del mundo

Es el componente encargado de proporcionar a la estación de tierra y los UAV virtuales la información acerca de la situación del entorno donde operan los UAV virtuales. Aunque aquí se da una visión general del mismo se trata con más detalle en la sección 5.4, ya que constituye la base para el empleo de la realidad mixta. Tiene dos funciones principales y su implementación depende de la tarea a realizar con los equipos de UAV:

- Por una parte, debe simular en entorno donde vuelan los UAV, sea este interior o exterior. En el caso de los entornos exteriores, se ha confiado gran parte de este trabajo al simulador de vuelo FlightGear, que tiene un modelo del terreno que recrea de forma veraz la morfología del entorno real usando ortofotos provenientes de World Scenery e información sobre la elevación obtenida de Shuttle Radar Topography Mission (SRTM). Además, el simulador del mundo está conectado a cada instancia del FlightGear e introduce los datos meteorológicos capturados por los UAV reales o los provenientes de fuentes en línea (como NOAA), para simular de forma precisa las condiciones atmosféricas en las que se está llevando a cabo la misión.
- Por otra parte, el simulador del mundo también se encarga de generar los valores de los sensores para la instrumentación de los UAV simulados. Esta sensorización se genera de forma dinámica según la tarea a realizar, las condiciones atmosféricas... Además, se puede tomar la información de los sensores de UAV reales e incorporarla para adaptar y corregir iterativamente las diferencias entre los sensores virtuales y los reales.

5.2.4. Ejemplo de las ventajas de la estructura modular

El desarrollo de la plataforma del presente trabajo se ha extendido durante un periodo prolongado de tiempo, a lo largo del cual se han realizado modificaciones y mejoras a diversas partes de la misma. Una de esas mejoras

incluyó un cambio en el simulador empleado, que gracias al empleo del software estructurado y modular no supuso cambios sobre el resto de la plataforma.

5.2.4.1. JMonkey

En una primera aproximación se optó por el empleo de un simulador de desarrollo propio implementado usando JAVA y el motor de juegos de OpenGL de jMonkey 3 (jME3). Además de emplearlo para simular los UAV y el entorno en el que se desplazaban servía como herramienta de visualización, al contar con una representación en 3D de todos los UAV del equipo.

jME3 es un motor de juegos de código abierto y gratuito, creado para desarrolladores de Java que quieren crear juegos 3D. Se trata de un software con muy buena documentación y sobre el que resulta sencillo comenzar a generar modelos y entornos de forma rápida.

El proceso de desarrollo comenzó con la creación de un entorno exterior, cercano al entorno real en el que se desarrollarían las misiones con los UAV. A continuación se realizó un modelo de los UAV, incorporando para ello las características más relevantes de los mismos. Finalmente, se incorporaron varios UAV al entorno de forma simultánea y se añadió una representación visual de la localización de los objetivos de la misión de prueba de la plataforma.

Empleando jME3 se llevaron a cabo varias simulaciones, algunas de las cuales se muestran en la Figura 5.8 y que forman parte del trabajo publicado en [Varela et al., 2011]. La misión en estas simulaciones consistía en la localización de una o varias fuentes de polución con un equipo de UAV. En la Figura 5.8 se puede ver los UAV y la fuente de polución, representada por un cilindro de color rojo.

De la misma forma que el jME3 ofrecía ventajas, como el despliegue rápido y la posibilidad de visualizar el proceso en tres dimensiones, también acarreaba una serie de limitaciones. Dentro de ellas estaba la dificultad de añadir fenómenos meteorológicos, especialmente tratándose de fenómenos dinámicos. De la misma forma la excesiva simplicidad del modelo dinámico implementado para los UAV llevaba a una diferencia de comportamiento demasiado grande entre los UAV simulados y los UAV reales. Una vez avanzado el proyecto se llegó a un punto en el que estas limitaciones del jME3 unidas a diversas ventajas que presentaban otras alternativas llevaron a su sustitución.

5.2.4.2. Flightgear

Se realizó una selección entre los simuladores existentes en la que se valoró especialmente la fidelidad a la realidad, la facilidad para desarrollar progra-

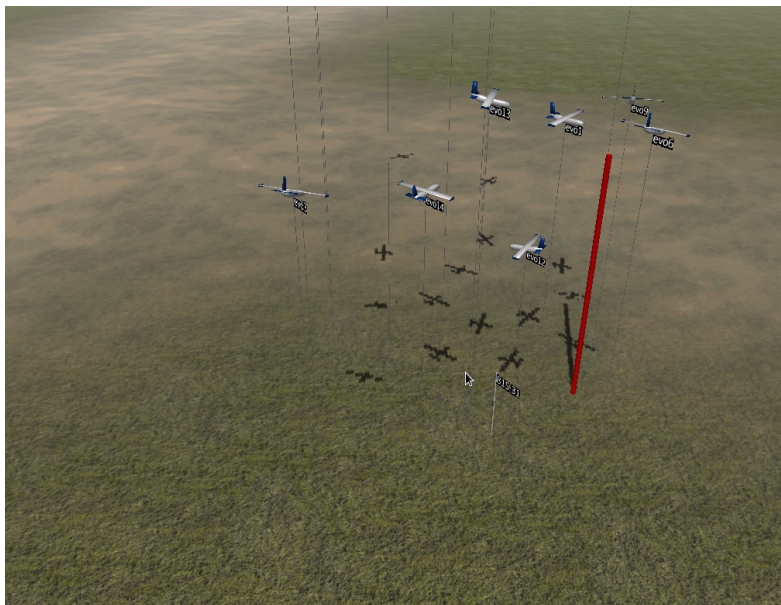


Figura 5.8: Simulación de varios UAV con jME3

mas conectados en tiempo real al simulador y la posibilidad de realizar nuevos modelos de aeronaves con facilidad.

Tras una revisión de las opciones, tanto libres como comerciales, se seleccionó el FlightGear [Flightgear, 2016]. Este simulador destaca, en primer lugar, por su modelo de dinámica de vuelo, el JSBSim [JSBSim, 2016]. Se trata de un programa de código abierto que modela la dinámica de vuelo de un vehículo aeroespacial con precisión. Además, dispone de la posibilidad de variar dinámicamente las condiciones atmosféricas y, al tratarse de software libre, existe una comunidad de desarrolladores y usuarios con abundante documentación lo que facilita y agiliza la tarea a realizar.

Todo esto ha propiciado que el FlightGear se use en investigaciones realizadas por diversas universidades alrededor de todo el mundo, como por ejemplo el simulador de base móvil de la Universidad de Nápoles, Italia [Coiro et al., 2007] o en la Universidad de Michigan, Estados Unidos [Miller, 2011] e incluso en empresas privadas.

El resto del proyecto se ha desarrollado empleando el Flightgear tanto en las simulaciones del trabajo publicado en [Varela et al., 2014a] como en [Lopez Peña et al., 2016]. La apariencia actual del entorno de simulación puede verse en la Figura 5.9. En la parte izquierda se muestra la interfaz de Google Maps con las trayectorias de todos los UAV y en la derecha una ventana por cada instancia del Flightgear que se encuentra en ejecución.

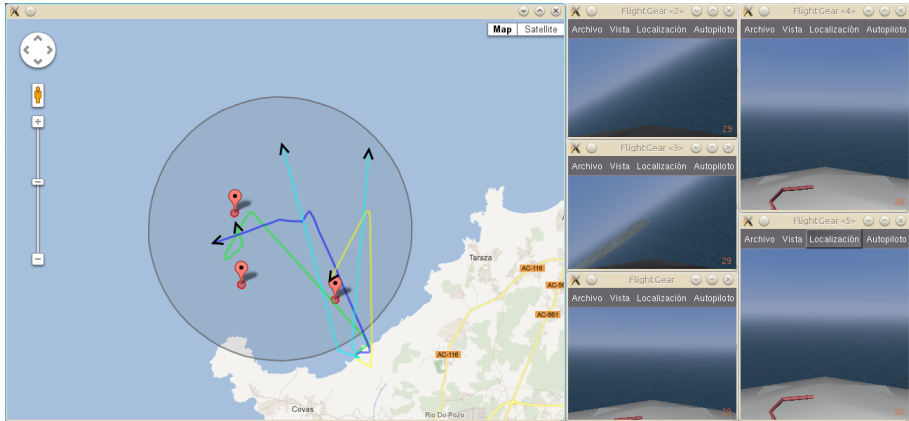


Figura 5.9: Simulación de varios UAV con FlightGear

5.2.4.3. Conclusiones

Con este ejemplo se pretende poner de manifiesto una de las principales ventajas que tiene el empleo de la estructura modular propuesta. Al dividir el programa en módulos es posible realizar cambios sobre alguno de los bloques funcionales sin tener que afectar el resto, ya que cada uno posee un alto grado de independencia.

En este caso, debido a que el jME3 se empleaba también como herramienta de visualización, el cambio de simulador obligó a cambiar también el bloque de visualización, pasando a emplearse un programa desarrollado sobre la API de Google Maps. Sin embargo, la estrategia de coordinación, así como el resto de bloques funcionales implementados se mantuvieron inalterados en el proceso.

El uso de la estructura propuesta posee también ventajas a la hora de realizar el desarrollo de los bloques funcionales en paralelo por un equipo de trabajo, para la búsqueda de errores y para la reutilización del código en posteriores aplicaciones. En general, se trata de una inversión de tiempo inicial mayor, comparado con una aproximación ad-hoc, que se va amortizando en el largo plazo, a medida que se emplea la plataforma y disminuye el tiempo requerido para hacer cambios y mejoras sobre la estrategia de coordinación.

5.3. Independencia del hardware

Actualmente existe un importante esfuerzo de investigación y desarrollo acerca de los UAV, lo que genera novedades de forma continua en todos los ámbitos relacionados con estos. Estos cambios afectan a las propias aeronaves,

pero en lo referente a la electrónica embarcada y los sistemas de control el ritmo de cambio es mucho mayor.

Así, cada pocos meses aparecen nuevos productos que mejoran a sus predecesores o a la competencia en algún aspecto. Dado que existen multitud de fabricantes en el mercado, uno de los métodos más empleados por estos para evitar la entrada de competencia es la creación de ecosistemas cerrados. Dentro de estos ecosistemas se emplean en muchos casos protocolos de comunicación propios entre los componentes que los forman. Así, el usuario se ve forzado a elegir una opción y, en muchos casos, no le es posible mezclar componentes de diversos fabricantes. Frente a estas soluciones cerradas han aparecido alternativas abiertas muy prometedoras como Ardupilot o PX4 que incluso permiten el acceso al código de forma libre y gratuita.

Sumado a este cambio, la intención en el presente trabajo es la de emplear la plataforma con UAV heterogéneos, es decir, que pueden no presentar las mismas características tanto desde el punto de vista de la aeronave como de la electrónica embarcada.

La forma que se propone en este trabajo para afrontar toda esta heterogeneidad es la de abstraerse de ella, mediante la inclusión de un middleware, evitando así que el software desarrollado se vea condicionado por las características particulares del hardware al que está asociado. En la sección 5.2 se ha descrito detalladamente la estructura propuesta para el software de todos los componentes de la plataforma, mientras que en esta sección se describe con mayor detalle el firmware y el middleware, su estructura e implementación.

5.3.1. Estructura del firmware

Desde un punto de vista general, el middleware es el encargado de permitir la comunicación de forma homogénea entre el software de alto nivel y el firmware o software de más bajo nivel, que es el que maneja el hardware. Esta sección se centra en explicar las funciones y estructura del firmware, con la intención de proveer una visión completa del software que ayude a ubicar y acotar la extensión del middleware.

El firmware es el encargado de manejar el hardware, estando en contacto directo con el mismo. Su complejidad viene determinada en gran medida por el hardware a manejar y proporciona al software de control métodos o funciones que permitan controlar y explotar las capacidades de este hardware. Se puede dividir en dos partes:

- La inicialización o configuración inicial: con la finalidad de preparar el hardware para que funcione y se comunique de la forma apropiada con la

aplicación es común que se requiera definir una serie de parámetros. Los parámetros de configuración van desde características del bus de comunicaciones empleado como la velocidad de transmisión a características propias del hardware conectado el formato de los datos a transmitir. Esta parte, tal y como indica su nombre se suele ejecutar al comenzar la comunicación con el dispositivo hardware.

- El código de manejo del hardware: con funciones o métodos que se pueden ejecutar de forma puntual o periódica. Gran parte de este código se ejecuta bajo demanda del software de nivel superior, aunque en algunos casos es el hardware el que lance la ejecución de código para transmitir información sobre eventos o estados al software de control. En muchos casos la ejecución de código se realiza periódicamente, de forma que se puedan mantener actualizada la información sin intervención del software de alto nivel.

Muchas de las características del firmware se ven influenciadas por el tipo de hardware al que está asociado, por lo que para concretar en mayor medida las funcionalidades del firmware se procede a tratar por separado cada uno de los grandes grupos de componentes de los UAV: los sensores, los actuadores y las unidades de control.

5.3.1.1. Firmware para sensores y transductores

Los sensores son la fuente de información del sistema y la primera característica diferenciadora frente a otro hardware reside en los protocolos de comunicación empleados. Dado que la mayoría de sensores van embarcados en el UAV no es necesario un protocolo que permita largas distancias, sino que se busca maximizar la velocidad de transmisión de volúmenes pequeños de datos y minimizar el cableado entre ellos. Así generalmente se emplean puertos de comunicación de tipo serie. Los estándares de comunicaciones más empleados son SPI, I2C y UART.

Para la adquisición de los datos de un transductor es necesario realizar una secuencia de acciones que finaliza con un resultado en forma de información. Este proceso conlleva un determinado tiempo y la información obtenida, a diferencia de la que se pueda obtener en un modelo de simulación, presenta problemas como valores anómalos y ruidos, por lo que debe ser filtrada antes de emplearse en el control. Además, se debe acondicionar la escala del sensor, que en muchos casos es diferente de la escala empleada en el software de control. Todo este proceso provoca un retraso desde la petición de un nuevo dato hasta la obtención del mismo, lo que favorece la realización de adquisiciones de datos de forma periódica frente al funcionamiento bajo petición. De esta forma se evitan

retrasos en la lectura de datos debido al proceso de adquisición y adaptación y se cuenta siempre con información actualizada disponible para su empleo.

Los sensores más importantes para los UAV son aquellos que ofrecen información sobre la posición y la actitud de la aeronave. Además se suelen incluir sensores que monitorizan el estado de los componentes básicos (temperatura, nivel de combustible, consumo de corriente...) y aquellos propios de la misión a realizar con los UAV.

5.3.1.2. Firmware para actuadores

Al igual que los transductores, los actuadores suelen requerir el envío de un volumen pequeño de información a una distancia reducida y con el empleo de pocos hilos de comunicación. Además, se prima mucho la minimización de la latencia o retardo desde que se transmite la orden desde el código hasta que la orden efectiva llega al controlador del actuador y este comienza su función.

De forma análoga a lo que sucede con los sensores se requiere una adaptación de las señales provenientes del sistema de control al formato y la escala esperada por los actuadores. Además, se han de filtrar y limitar estas señales para evitar enviar órdenes que pongan en peligro la aeronave o los actuadores.

En el caso de los UAV de pequeño tamaño los principales actuadores son los servomotores y el grupo propulsor formado por los motores y las hélices o rotores. La gran mayoría de estos dispositivos se controla mediante una modulación por ancho de pulso (PWM) y requieren de un hilo para cada actuador. Para reducir la longitud y el número de estos hilos actualmente se emplean protocolos como el SPPM o el SBUS, analógico y digital respectivamente, que permiten enviar varias señales a través de un hilo. Esto simplifica el cableado y facilita la generación de las señales de control, al tratarse de una única señal para varios servomotores. En cualquier caso, al final de este hilo se ha de decodificar la señal y enviarse a cada dispositivo por separado la señal PWM correspondiente.

También es frecuente el control de actuadores que únicamente cuentan con dos estados como interruptores o relés mediante el nivel de tensión eléctrica. En el caso de UAV mayores o actuadores específicos de la misión a realizar puede existir una mayor variedad de protocolos y buses de transmisión.

5.3.1.3. Firmware para unidades de control

Se trata de una unidad de procesamiento que recoge e interpreta la información proveniente de los sensores y transductores, realiza los cálculos programados y genera las salidas para los actuadores. Los protocolos de comunicación con sensores y actuadores se han analizado en los apartados anteriores.

El bloque funcional que se ocupa del manejo de la unidad de control es el control de vuelo básico, concretamente la parte del piloto automático es la que realice la unidad de control. Así, según el modo de vuelo definido en el control de vuelo básico recibe unas consignas que ha de convertir en actuaciones. Para ello, se vale de mecanismos de control como los PID, que se basan en el error entre el estado actual y la consigna.

Cuando se adquiere una unidad de control comercial se suministra con un firmware preprogramado que incorpora una lógica de control más o menos compleja de los actuadores principales (servomotores y grupo propulsor). También existe la posibilidad de realizar la unidad de control desde cero a partir de un microcontrolador, de forma que se puede definir la lógica de control desde el nivel más básico. La elección de una u otra opción determina la cantidad de firmware a desarrollar para obtener la funcionalidad deseada, así como el formato y bus de comunicación empleado para transmitir la información.

Así, la funcionalidad básica que se pretende para el firmware de la unidad de control es el vuelo a una velocidad dada, hasta un punto de forma controlada y estable. La generación de la consigna de velocidad y del punto de destino vienen dadas por las características de la aeronave y por el cálculo de trayectorias del control de vuelo básico.

También pueden existir otras unidades de procesamiento, encargadas de ejecutar la parte del código relativa a la estrategia de coordinación con las que también se debe intercambiar información. Para la interconexión de unidades se suelen emplear USB, UART y, en el caso de que los volúmenes de información sean mayores, buses como CAN o Ethernet. La introducción de más unidades de procesamiento obliga a incluir un firmware que maneje las comunicaciones entre ambas.

5.3.2. Estructura del middleware

Se puede dividir el middleware en tres partes, tal y como se muestra en la Figura 5.10, que han de ser correctamente definidas para lograr que el middleware cumpla su función y permita el intercambio de información entre el software de alto nivel y el firmware:

Interfaz con el software de alto nivel La primera de las partes es la encargada de poner a disposición del código de alto nivel las funciones y métodos para permitir la comunicación con el firmware, habilitando así el acceso a las capacidades del hardware.

Esta interfaz puede ser definida según las preferencias del desarrollador

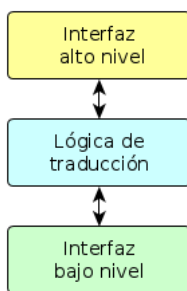


Figura 5.10: Esquema general de la estructura del middleware

del software. En este caso, siguiendo las premisas de diseño de simplicidad y flexibilidad se ha optado por una definición mínima con funciones para la lectura y escritura de los valores de sensores y actuadores, así como la configuración básica del firmware.

Interfaz de bajo nivel En segundo lugar, la interfaz de bajo nivel se encarga de realizar la misma función que la anterior, pero para el código de bajo nivel. De esta forma, se ofrece al código de bajo nivel la oportunidad de iniciar una comunicación y transmitir información al alto nivel cuando la situación lo requiere.

Se trata de un canal de comunicación que permite informar sobre eventos o situaciones relacionadas con el hardware, tales como posibles averías, realimentación de acciones críticas, etc. Así, por ejemplo, si hay un sensor de temperatura en los motores y se define un umbral de seguridad, se puede disparar una alarma de seguridad en el firmware cuando se sobrepase para que el software de alto nivel la reciba lo antes posible.

Esta interfaz se define por el firmware desarrollado y la inteligencia definida en este, así como por las capacidades del hardware al que se encuentre conectado.

Lógica de traducción Finalmente, para poder comunicar ambas interfaces debe existir una lógica encargada de traducir las funciones de entrada de uno de los niveles en funciones de salida, de forma que el código al otro lado pueda comprender la información transmitida. Esta parte constituye el núcleo del middleware.

La complejidad de esta lógica es variable, ya que depende de las definiciones de ambas interfaces, especialmente de las capacidades del firmware. En el caso de las unidades de control, si se opta por adquirir una unidad de control comercial y el firmware es cerrado se debe ajustar a los comandos disponibles, teniendo que

emplear varias funciones si se quieren realizar labores más complejas o incluso desarrollar partes del firmware que aporten las funcionalidades que no posee la unidad de control.

5.3.2.1. Middleware para sensores

De forma general se ha optado por simplificar al máximo la interfaz del alto nivel, de forma que el firmware se ocupe de la mayor parte de las tareas concretas como el manejo y gestión de los buses de comunicación y se ofrezcan una serie de funciones o métodos sencillos que permitan acceder a las capacidades del hardware, abstrayéndolo de las características concretas del mismo. En el caso de los sensores el middleware ofrece al software de alto nivel una función para la configuración inicial y varias para el manejo del dispositivo.

En el caso de la inicialización, no se trata de una definición de los parámetros concretos de la comunicación o el sensor, ya que esto se gestiona desde el firmware, sino de la definición de variables del firmware. Existen varias opciones a configurar: el número de muestras empleadas para el filtrado del sensor, si se desea que se realicen lecturas de los sensores de forma periódica, la frecuencia de muestreo, etc. Estos parámetros son filtrados para ajustarse a las capacidades del hardware conectado, que es el factor limitante.

De la misma forma, se pueden gestionar las alarmas que se hayan definido en el firmware, habilitándolas o no, según sea necesario para la aplicación a desarrollar. Las alarmas constituyen la principal forma de comunicación del firmware hacia el software de alto nivel, llamando la atención sobre eventos hardware para que se emprendan las acciones que se estimen oportunas.

Las funciones principales definidas para el acceso del software de alto nivel al hardware son dos: la función de muestreo y la de lectura. La primera de ellas se emplea para lanzar una adquisición de datos puntual y se puede combinar con la programación de una alarma cuando el dato esté disponible. De esta forma, se puede conocer el dato proporcionado por el sensor en un instante muy concreto de tiempo y mediante la definición de una máquina de estados, o similar, es posible tomar el nuevo dato cuando el firmware avise de que está disponible, sin mantener la unidad de procesamiento ocupada hasta que finalice el proceso de adquisición.

La función de lectura se emplea para la lectura de datos, ya sean provenientes de una adquisición realizada de forma periódica o de forma puntual. Ofrece como respuesta el valor del sensor filtrado y en unidades del sistema internacional. En el caso de que el sensor ofrezca más de un dato, como sucede en el GPS, la respuesta es una estructura de datos con todos ellos, en este caso con la posición, el rumbo y la velocidad del UAV.

5.3.2.2. Middleware para actuadores

Siguiendo la misma línea que para el caso de los sensores se propone una interfaz mínima para el alto nivel y la estructura sigue el mismo esquema con una función de configuración y otras para el manejo del hardware.

En lo referente a la función de inicialización, varios parámetros como la frecuencia de actualización para las señales vienen dados por las características del hardware. Se puede configurar si se desea realizar la actuación al escribir un valor nuevo o si se prefiere establecer algún método posterior para confirmar el valor y actuar.

Las principales funciones para el acceso al hardware son dos: la función de escritura y la de actuación. La primera de ellas se ocupa de escribir en los registros del firmware el valor de salida deseado para el actuador. Este valor de salida se convierte de las unidades de trabajo a la escala adecuada según el actuador a utilizar y se limita según las capacidades del hardware.

Además, en determinados actuadores interesa controlar el momento exacto en el que se ejecuta la misma con la finalidad de sincronizarla con otros eventos. También puede resultar útil para comprobar que el valor que el firmware es correcto antes de realizar la actuación. Para este tipo de funcionamiento, en el que la escritura y la actuación son dos eventos separados, se implementa la función de actuación, que se encarga de lanzar el proceso de cambio al valor que se ha escrito previamente.

Dependiendo de las preferencias del desarrollador, se puede definir el comportamiento del actuador en la configuración inicial, de forma que también es posible omitir el uso de la función de actuación realizarla al escribir un nuevo valor mediante la función de escritura.

5.3.2.3. Middleware para unidades de control o pilotos automáticos

La inicialización de las unidades de control sirve para gestionar las alarmas provenientes del firmware de la misma, así como para definir los parámetros de configuración disponibles que varían según las capacidades de la unidad de control y del firmware del que disponga. De forma general debe existir un método para ajustar la ganancia de los controladores del piloto automático así como la combinación de controles manuales y automáticos sobre los actuadores. La intención de esto es el aumento de la seguridad a la hora de realizar pruebas reales de los controladores y realizar un ajuste fino de los mismos. Una vez estos parámetros se encuentren definidos, se puede fijar su valor como parte de la inicialización.

En lo que se refiere al manejo únicamente se ha definido una función, que

transmite las consignas calculadas en el control de vuelo básico a la unidad de control. Estas consignas se calculan en función del modo de vuelo en el que se encuentre el UAV y se componen de tres grupos de datos principales: la posición, la dirección y la velocidad.

- La posición se define, en el caso de entornos exteriores, mediante las coordenadas geográficas (latitud y longitud) así como la altura. Esta última puede ser definida con respecto al nivel del mar o tomando como referencia la altura de la zona de despegue de la aeronave.
- Con respecto a la dirección o rumbo, se da en grados sexagesimales, con el Norte coincidiendo con el 0° y en sentido creciente hacia el oeste que correspondería con 90° . Se corresponde con la dirección deseada para finalizar el movimiento.
- Por último, la velocidad de la aeronave con respecto a tierra. Se emplea esta en vez de con la velocidad con respecto al viento incidente para garantizar que las trayectorias generadas son repetibles, aunque varíen las condiciones atmosféricas. En el caso de UAV de ala fija ha de tenerse en cuenta la velocidad de pérdida de la aeronave para evitar las pérdidas de sustentación.

Estos datos son traducidos en el middleware, de forma que se generan los comandos adecuados según el tipo de aeronave empleado. Así, en el caso de UAV de ala fija, generalmente se trabaja con rumbos, alturas y velocidades con respecto al viento. Por contra, en el caso de multirrotores es más común el empleo de velocidades de desplazamiento y de rotación con respecto a la posición actual. De igual forma las unidades también deben ser convertidas a las de trabajo del piloto automático.

En el caso de que la unidad de control disponga de funcionalidades más avanzadas, como el modo de vuelta a casa automático se define también el acceso a dichas funciones al seleccionar el modo de vuelo, omitiéndose entonces la conversión de las consignas.

5.3.3. Ejemplo de las ventajas de la independencia del hardware

Al igual que en el caso del software, en lo que respecta al hardware se han producido también múltiples variaciones a lo largo del desarrollo del presente proyecto. Se han realizado vuelos con siete aeronaves diferentes, cinco de ala fija y dos multirrotores, aunque no todas se han llegado a integrar en la plataforma. Se han empleado diferentes emisoras y enlaces de comunicación inalámbricos,

así como diferentes sensores. Sin embargo, uno de los cambios en los que mejor se refleja la importancia del middleware y la abstracción del hardware ha sido la variación de la unidad de control.

5.3.3.1. Primeros pasos, DMD y Eagle Tree

En primera instancia se seleccionó un piloto automático comercial de DMD, actualmente descatalogado, ya que ofrecía funcionalidades superiores a las alternativas del momento. Debido a las limitaciones del firmware, que no permitía la ampliación del sistema o la comunicación con otras placas de procesamiento o sensores diferentes de los distribuidos por el propio fabricante se decidió finalmente buscar una unidad de control alternativa.

La alternativa seleccionada fue la de Eagle Tree, que ofrecía una mayor versatilidad a la hora de conectar sensores externos o emplear sus sensores en otros controladores, así como un modo de vuelta a casa automático y otro con estabilización activa del UAV basado en una IMU. Además de servir de piloto automático, contaba con la capacidad de enviar datos a través del enlace de vídeo en tiempo real y más sensorización que el anterior.

Tras varios vuelos iniciales satisfactorios, en los que se probó la funcionalidad del estabilizador y la vuelta a casa, se comenzó a integrar el piloto automático con la unidad de control propia. Ésta, centrada en el control de vuelo básico, se ha desarrollado entorno a un microcontrolador PIC32. La lógica de control con la que se pretendía lograr integración de ambos sistemas es la que puede verse en la Figura 5.11:

Así, las consignas que recibía del control de vuelo básico se convertían en una altura con respecto al punto de despegue, una velocidad con respecto al viento y un rumbo para realizar la trayectoria hasta el punto de destino. Estas consignas se comparaban con los valores obtenidos de los sensores presentes en el UAV y, mediante varios controladores PID se obtenía la salida para las superficies de control del UAV.

Todas las salidas, excepto la de control del motor, pasaban antes por el estabilizador de Eagle Tree, denominado comercialmente Guardian. Se realizaron vuelos de prueba con esta configuración y el resultado fue bastante pobre, debido a una dificultad para ajustar los controladores PID desarrollados y sobre todo una inestabilidad en la velocidad del UAV. La estabilización, probada anteriormente con vuelos manuales y cuyo funcionamiento con los mismos era muy buena, no se ajustaba tan bien a los comandos generados desde el controlador automático. Además, la estabilización no era capaz de controlar la velocidad de la aeronave para mantenerla lejos de la velocidad de pérdida en condiciones de planeo o motor a baja potencia.

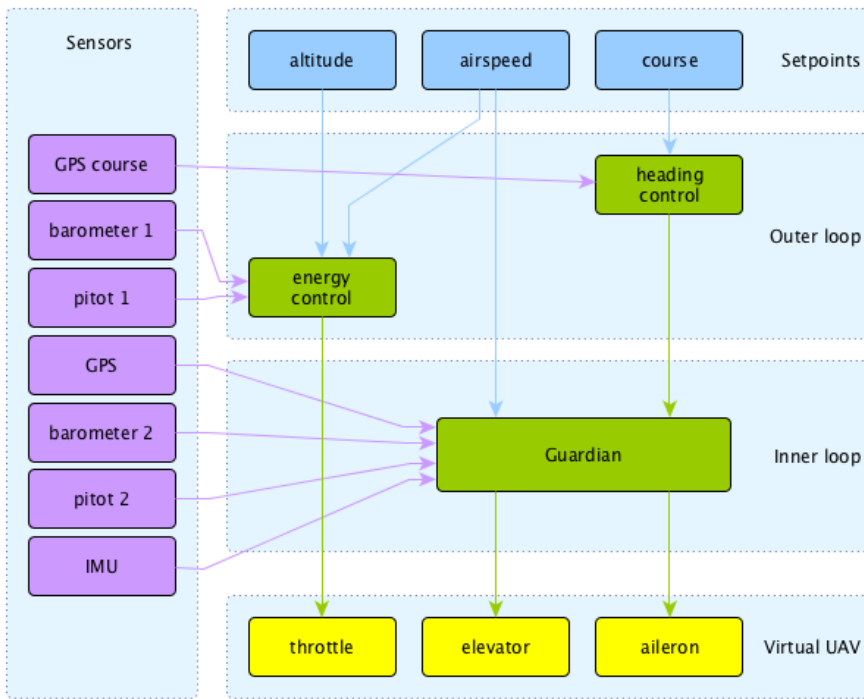


Figura 5.11: Lógica de control inicial del piloto automático con Eagle Tree

5.3.3.2. La solución, lógica de control propia

La solución a este problema pasó por retirar el Guardian de la lógica de control, conectando una IMU a la unidad de control desarrollada para esta aplicación, de forma que se dispusiese de información del cabeceo y alabeo, pasando a una lógica como la que se puede ver en la Figura 5.12:

El control del motor es idéntico al del caso anterior y se sustituye el Guardian por un control de velocidad y otro de alabeo. Finalmente, el control de velocidad se dividió en dos controladores individuales: el primero convierte la consigna de velocidad en una consigna de ángulo de cabeceo y el segundo toma esta consigna, actuando sobre la profundidad de la aeronave para mantenerla.

Así, mediante la limitación de los ángulos de cabeceo y alabeo se consiguió realizar un control del vuelo que garantizase la seguridad y que minimizase las entradas en velocidades de pérdida en situaciones de planeo o motor a baja potencia.

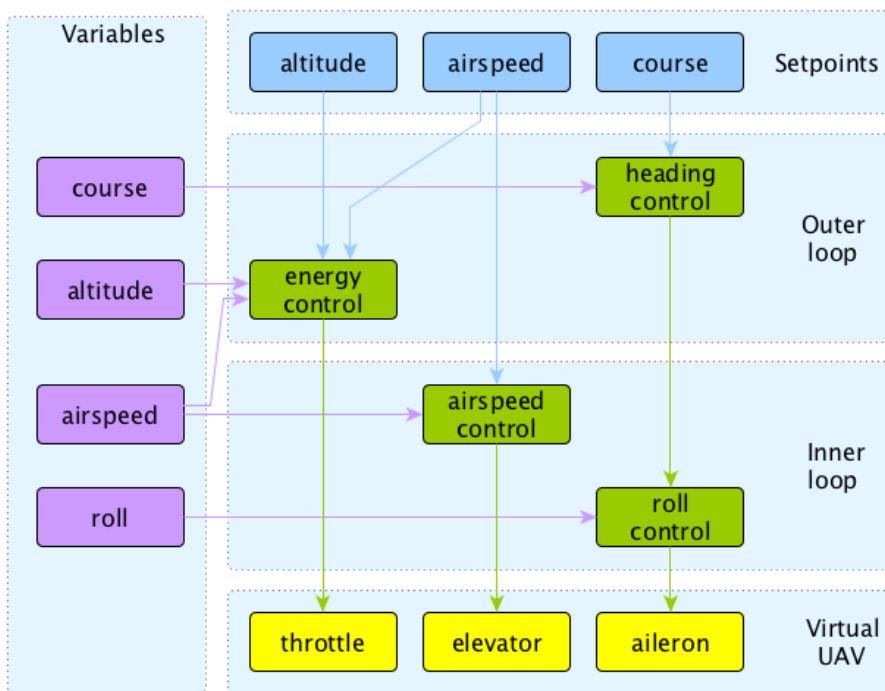


Figura 5.12: Lógica de control final del piloto automático

5.3.3.3. Conclusiones

El ejemplo planteado pone de manifiesto que el desarrollo de una plataforma en la que se incluyen elementos reales como los UAV es un proceso en el que se suceden cambios de hardware, muchas veces imprevistos. El impacto de dichos cambios sobre el software de la plataforma ha de minimizarse para reducir la carga de trabajo necesaria para mantener la plataforma.

Así, en este caso la estrategia de coordinación no sufrió ningún cambio durante la transición de los diferentes controladores y el control de vuelo básico únicamente se vio afectado en la parte del piloto automático, manteniendo los modos de vuelo y el cálculo de consignas inalterado. Esta capacidad de adaptar el software y afectar únicamente a partes concretas del código aporta flexibilidad a la plataforma, permitiendo actualizar los componentes cuando sea necesario sin necesidad de variar la estrategia de coordinación.

Cuando además se pretende emplear UAV heterogéneos esta capacidad adquiere una mayor importancia, ya que en este caso no se trata de un proceso de cambio puntual, sino de una diferencia permanente entre el hardware que ha de asumirse de alguna forma en la plataforma. Mediante el middleware y el

firmware propuestos se consigue abstraer la estrategia de coordinación de las diferencias presentes en el hardware, centrando los esfuerzos en el desarrollo de la misma.

5.4. Incorporación iterativa y progresiva de la complejidad real

Una de las herramientas que se han vuelto indispensables a la hora de realizar nuevos desarrollos en la investigación son los simuladores. Mediante la simulación se accede a un número mayor de recursos a un coste mucho menor, ya que el coste de un ordenador capaz de ejecutar una simulación de un sistema es despreciable frente al coste de construir y desplegar el mismo sistema en la realidad. El aumento de la capacidad computacional ha llevado aparejado una mejora en los simuladores, aunque se sigue manteniendo el problema del reality gap.

El reality gap se define como la diferencia entre el simulador y la realidad y se produce debido a la enorme complejidad existente en la realidad en la que multitud de sistemas y efectos se entrecruzan y que no se puede reproducir con fidelidad en un simulador.

Para minimizar el efecto del reality gap sobre el desarrollo de estrategias de coordinación en este trabajo se propone el empleo de la realidad mixta. La realidad mixta consiste en una combinación del mundo virtual con el mundo real, en tiempo real. Esta combinación permite crear nuevos espacios de trabajo, en los que interactúan tanto objetos y/o personas reales como virtuales.

Siguiendo esta línea, en la plataforma propuesta se incluye un entorno de realidad mixta, así como la figura de los UAV virtuales, que representan de forma indistinta a UAV reales, simulados o a una combinación de ambos. En esta sección se explican con detalle ambos conceptos y las ventajas potenciales de su uso.

5.4.1. UAV virtuales

Constituyen uno de los elementos centrales de la plataforma desarrollada. En la sección 5.2 ya se ha descrito con detalle la estructura del software de los mismos y ahora se centra en la parte del hardware.

La visión tradicional del hardware es la de los elementos físicos o materiales que constituyen un sistema informático. Sin embargo, en este trabajo, mediante el empleo de la realidad mixta se incluyen en el hardware también elementos

virtuales. La intención última es tratar a los UAV simulados y reales de forma equivalente, para lo cual se deben resolver una serie de cuestiones:

El comportamiento dinámico de los UAV simulados Se trata del principal requisito a la hora de simular el hardware, se debe contar con un modelo del comportamiento de la aeronave. Esto se compone de dos partes: la simulación del entorno en el que se desplaza y la simulación del propio UAV. Este apartado se centra en la segunda, dejando la parte del entorno para el simulador del mundo.

Para simular el UAV hay que desarrollar un modelo de comportamiento similar al de la aeronave real. Esto incluye tanto el comportamiento aerodinámico de la estructura o célula como el del sistema propulsor que emplee. Este proceso se realiza mediante la introducción de datos y parámetros característicos de la aeronave en un software de simulación, que es el encargado de calcular el comportamiento en base a estos.

La plataforma admite el empleo de cualquier simulador, únicamente es necesario establecer la comunicación necesaria con el resto de elementos de la plataforma mediante UDP. En las pruebas en entornos exteriores se ha utilizado el simulador de código abierto Flightgear. Este simulador destaca, en primer lugar, por su modelo de dinámica de vuelo, el JSBSim que modela la dinámica de vuelo de un vehículo aeroespacial con precisión. En segundo lugar, al tratarse de un programa de código abierto resulta más sencillo añadir o conectar nuevas funcionalidades que en el caso de simuladores cerrados. Además, este simulador dispone de una comunidad de desarrolladores y usuarios con abundante documentación lo que facilita y agiliza la tarea a realizar.

Con la finalidad de aportar un mayor realismo a las pruebas, resulta importante también modelar la autonomía en vuelo de la aeronave, que viene determinada por la energía almacenada y el consumo que se realice, principalmente en el motor. La energía se almacena en forma de baterías, en caso de que la propulsión sea eléctrica o en forma de depósitos de combustible en otros casos y para simular ambos basta calcular la energía almacenada. Se puede modelar el consumo de combustible usando información sobre el funcionamiento de los motores proporcionada por el simulador y basándonos en datos empíricos del consumo de la aeronave real.

La instrumentación de los UAV simulados Simular el comportamiento de la aeronave es el primer paso, pero resulta insuficiente para permitir la equivalencia de un UAV real y uno simulado. Actualmente los UAV cuentan con diversos sistemas electrónicos que también es necesario reproducir para conseguir dicha equivalencia.

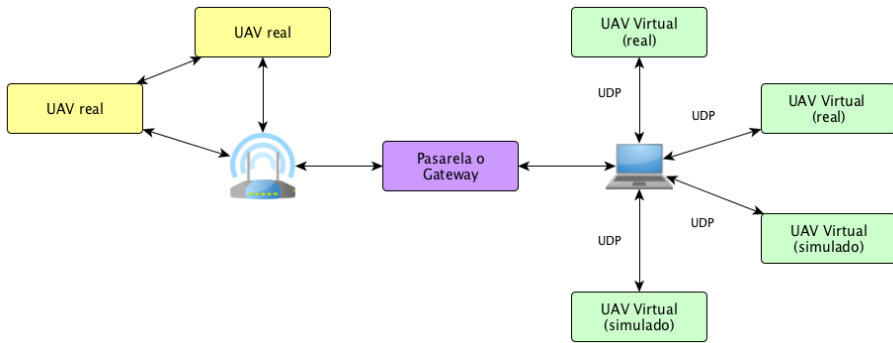


Figura 5.13: Estructura de la comunicación entre UAV reales y simulados

En primer lugar están los sensores, uno de los elementos más numerosos dentro de la instrumentación electrónica incluida en los UAV. Los sensores de posición y orientación, así como de las revoluciones de las hélices y el motor suelen venir suministrados por el software de simulación empleado para el comportamiento de la aeronave. Los demás sensores, incluidos aquellos empleados para determinadas misiones, requieren el modelado de la información que miden para permitir su simulación. En el caso de que se trate de información acerca de un UAV concreto, como puede ser: la posición de un actuador, la tensión presente en la batería o el nivel del depósito de combustible el modelo se incluye en la simulación de los UAV. Si, por el contrario, se trata de información para todos los UAV, como la información meteorológica o algún fenómeno a estudiar se incluyen los modelos en el simulador del mundo.

En segundo lugar están las comunicaciones, ya que resulta necesario establecer un modelo para el intercambio de información entre los UAV. Para garantizar el realismo en las simulaciones se ha de incluir un modelo de los módulos de comunicación, que en su forma más básica tenga en cuenta unas limitaciones de alcance para el intercambio de información entre UAV. Es recomendable incluir también un porcentaje de paquetes perdidos, que puede ser proporcional a la distancia.

Finalmente los actuadores y su efecto sobre el entorno. En el caso de los actuadores básicos para el control de la aeronave, su modelado viene dado por el simulador de la misma. El resto de actuadores provocan unos efectos sobre el entorno o la aeronave que se han de modelar para conseguir un comportamiento similar al real.

La comunicación entre UAV simulados y reales Al igual que entre los UAV de un equipo formado únicamente por UAV reales y aquellos constituidos

íntegramente por UAV simulados resulta necesario intercambiar información entre todos los UAV virtuales para realizar una coordinación efectiva. Así, además del modelo de comunicación comentado en el apartado anterior para los UAV simulados, resulta necesario el intercambio de información entre los UAV reales y los UAV simulados.

Para ello es necesario implementar una pasarela capaz de intercambiar información entre el simulador, que se comunica mediante paquetes UDP y el equipo real de aviones, que emplea una tecnología de comunicación inalámbrica. Esta pasarela es una combinación de hardware y software que se conecta simultáneamente al ordenador en el que se está ejecutando el simulador y a la red inalámbrica de los UAV reales, haciendo el trasvase de información entre ambas redes de forma transparente para el desarrollador.

Se puede emplear la estación de tierra para realizar esta función en caso de que exista una vía de comunicación entre esta y el ordenador sobre el que se ejecuta la simulación de los UAV simulados.

5.4.2. Simulador del mundo

Es el componente encargado de proporcionar a la estación de tierra y los UAV virtuales, la información acerca de la situación del entorno donde operan los UAV virtuales. Se trata de la base para la realización de simulaciones con realidad mixta, generando un entorno en el que se pueden mezclar los datos reales y virtuales. Tiene dos funciones principales: la simulación de un entorno realista y la de los sensores.

Simulación de un entorno realista Dado que los UAV se desplazan por el aire, toda variación en este medio provoca un cambio en el comportamiento y las capacidades de la aeronave. Uno de los aspectos que tiene mayor peso dentro de este aspecto es la simulación de las condiciones atmosféricas e interesa que sea lo más cercana posible a la realidad.

Aunque la plataforma desarrollada permite el uso de cualquier simulador, para el desarrollo de las pruebas en entornos exteriores se ha seleccionado el Flightgear. Una de las características más destacables de este simulador es que, además de modelar con precisión el comportamiento dinámico de las aeronaves permite la simulación de diferentes fenómenos meteorológicos como la lluvia o el viento. La información acerca de las condiciones meteorológicas reales puede ser capturada por sensores en los UAV reales o la estación de tierra y transmitida a través de estos a la plataforma. También existe la posibilidad de tomar datos meteorológicos provenientes de fuentes en línea como NOAA.

Otro aspecto importante en la simulación de los entornos son los obstáculos y el terreno. El Flightgear cuenta con un modelo del terreno que recrea de forma veraz la morfología del entorno real usando ortofotografías provenientes de World Scenery e información sobre la elevación obtenida de Shuttle Radar Topography Mission (SRTM). En lo referente a otros obstáculos, fijos o móviles, estos han de ser añadidos posteriormente a la plataforma.

Simulación de los sensores Al igual que en el caso del comportamiento dinámico y la instrumentación, el entorno no se compone únicamente de su parte mecánica y para conseguir la equivalencia con el entorno real también es necesario simular otros fenómenos y propiedades del entorno.

Así, se debe modelar e incluir en la plataforma todo aquello que se mida con los sensores, de forma que la instrumentación virtual de los UAV simulados cuente con una fuente de información equivalente a la de los UAV reales. Los modelos dependen de la sensorización requerida para la realización de las misiones. Así, si la misión consiste en la medición de los niveles de concentración de un contaminante es necesario modelar la dispersión en el medio de dicho contaminante, de forma que los UAV virtuales puedan tomar esa información y generar los valores de los sensores simulados.

Los modelos de los fenómenos o propiedades medidas deben generarse de forma dinámica, es decir, teniendo en cuenta las variaciones debidas a las condiciones del entorno que le afectan. Además resulta interesante que los modelos realizados se puedan corregir, ya sea en tiempo real o en diferido, con datos obtenidos de la sensorización real. De esta forma se permite una mayor cercanía en el funcionamiento entre los sensores reales y virtuales, haciendo posible la mejora iterativa de los modelos realizados así como su validación contra los datos reales.

5.4.3. La incorporación de la complejidad real

Una vez conocida la estructura de la plataforma, tanto en lo que se refiere al software como al hardware simulado, se han de definir también las posibles aplicaciones que tiene el uso de la misma. En este trabajo se ha optado por enfocar la plataforma como una herramienta que permita incorporar la complejidad de la realidad de forma progresiva e iterativa.

Uno de los mayores problemas asociados al reality gap es la forma de realizar el paso entre el simulador y la realidad. Así, de forma tradicional se ha de “saltar” a la realidad directamente, tomando los controladores desarrollados en el simulador y probándolos sobre el hardware real. Posteriormente se ha ido depurando este proceso y se han desarrollado etapas intermedias en el proceso

como el HIL (Hardware In the Loop), en el que el controlador se ejecuta sobre el hardware real pero el resto del hardware, el entorno y los sensores son simulados. La realidad mixta constituye una evolución más ambiciosa y flexible, que nos permite realizar el paso a la realidad de forma incremental.

5.4.3.1. Desarrollo e implementación de las estrategias de coordinación

En la plataforma de realidad mixta los UAV virtuales pueden representar indistintamente a un UAV real o simulado se pueden realizar las combinaciones de cada uno de ellos que se considere más adecuado. De esta forma, siguiendo la metodología propuesta en el capítulo anterior se plantea hacer uso de la realidad mixta para facilitar el paso del simulador a la realidad de los controladores realizados empleando la plataforma.

UAV autónomos El proceso comienza con las pruebas reales del control de vuelo básico desarrollado sobre el modelo del simulador. Para facilitar estas pruebas, en la plataforma se trabaja con el middleware que abstrae el hardware concreto, de forma que el controlador desarrollado sobre el simulador es exactamente el mismo que el que se prueba sobre el UAV real. Una vez realizadas estas pruebas se procede a analizar los datos obtenidos en las mismas, ajustando lo que sea necesario en el controlador y en el modelo hasta que se disponga de una aeronave capaz de volar de forma autónoma.

Este proceso se ha de repetir para los diversos tipos de aeronaves que se pretenden utilizar. Así, al finalizar el mismo todas las aeronaves del equipo han de ser autónomas, de forma que puedan mantenerse en el aire de forma estable y seguir los modos de vuelo definidos para el control de vuelo básico en la sección 5.2. Este control es capaz de manejar el UAV simulado y el real.

Implementación de la estrategia de coordinación El siguiente paso a realizar es el desarrollo de la estrategia de coordinación con los UAV simulados. Una vez alcanzado un resultado prometedor se pasa a la etapa de pruebas reales de la estrategia. Estas pruebas iniciales de una nueva estrategia de coordinación con un único UAV real y el resto simulados. Además, al tratarse de una plataforma en la que se ha abstraído el hardware, el software incluido en ambos UAV es el mismo. De esta forma se pretende facilitar el desarrollo de las estrategias de coordinación, aumentando la seguridad en las pruebas iniciales y permitiendo un despliegue más rápido y económico de las mismas en la realidad.

Así, una vez iniciado el proceso y verificado el buen funcionamiento de la estrategia de coordinación con un UAV real se puede proceder a incrementar

progresivamente el número de UAV reales frente a los UAV simulados. Se debe comprobar durante este proceso que la estrategia continúa funcionando correctamente y no surgen problemas por limitaciones reales que no se han tenido en cuenta en el simulador. Uno de los principales factores que pueden crear problemas son las comunicaciones, debido a los retrasos o latencias, las limitaciones en el ancho de banda, los paquetes perdidos o las reducciones de alcance debido a obstáculos o similares. Resulta crucial intercambiar información de forma rápida y efectiva entre los UAV y cuando este proceso falla, la coordinación del equipo se resiente.

Se debe tener siempre en cuenta que la intención final de este proceso es la implementación en la realidad de un equipo de UAV que, mediante una estrategia de coordinación, sean capaces de realizar una misión de forma colaborativa. Esta misión la realizan únicamente los UAV reales, siendo el resto una forma de aportar al sistema información de modelos o vuelos anteriores.

Eliminación o conservación de los UAV simulados A priori el final del proceso de paso de UAV simulados a reales se alcanza cuando todos los UAV virtuales se corresponden con UAV reales. Sin embargo durante el proceso es posible que se alcance otra solución si, por ejemplo, se ha sobrestimado el número de aeronaves necesarias para la realización de la misión. Inicialmente se toman un número de UAV que se considere adecuado, pero puede darse el caso de que la estimación no resulte exacta en la práctica. En ese caso, antes de que todos los UAV sean reales se llega a un número suficiente para realizar correctamente la misión.

Llegados a este punto se podrían eliminar los restantes UAV simulados, sin embargo, puede darse el caso de que los UAV simulados estén aportando información extra a los UAV reales a través de los modelos realizados para los sensores de misión.

Por ejemplo, si la misión consiste en la localización de unas fuentes emisoras de radio, han de simularse dichas fuentes para su uso en los sensores de los UAV simulados. En las pruebas de la estrategia de coordinación en las que se usa una combinación de UAV reales y UAV simulados, si se emplean modelos precisos para los sensores virtuales, las medidas virtuales aportadas por ellos son cercanas a las reales, ofreciendo mayor información de la que se dispondría si se trabajase únicamente con los UAV reales.

Teniendo esto en mente, los modelos realizados para los sensores han de ser precisos y adaptados a la realidad. Para ello se deben basar en los datos existentes, ya sea los recogidos por los sensores en vuelos anteriores u otros datos existentes referentes al fenómeno que se pretende estudiar. En lo referente a la adaptación a la realidad, los modelos han de ser dinámicos, permitiendo su

reajuste según las condiciones de los vuelos reales que le afecten (temperatura, viento, humedad...). De esta forma se aprovecha al máximo tanto la información de que se dispone acerca de los fenómenos o eventos a estudiar así como aquella proveniente de la experiencia y obtenida en la realización de pruebas o misiones anteriores.

5.4.4. Ejemplo de las ventajas de la incorporación de la información real

Una de las primeras aplicaciones que se planteó para demostrar el potencial de la plataforma desarrollada para el presente trabajo fue la localización de naufragos en un área amplia del mar [Varela et al., 2011]. En esta aplicación cada naufrago porta un chaleco con una radiobaliza de baja potencia y los UAV cuentan con un sensor capaz de captar y medir la intensidad de la señal emitida.

Además, para evitar el empleo de la direccionalidad de las ondas para facilitar la localización de los naufragos se realizó un montaje con tres receptores, de forma que se garantizase que el receptor era omnidireccional. Esto tiene sentido para simular un campo escalar, dado que en muchas aplicaciones de vigilancia ambiental como la localización de fuentes de polución no existe una direccionalidad de la que poder aprovecharse.

5.4.4.1. Montaje y modelo inicial de las radiobalizas

Para imitar la radiobaliza del naufrago, se realizó un montaje de un emisor de RF equipado con una antena SPW de 5.8 GHz y alimentado por una batería.

Como primera aproximación para el modelo de emisión de las radiobalizas se supuso que la antena equipada en los transmisores es omnidireccional, es decir, que emite con la misma intensidad en todas las direcciones. El patrón de radiación real es el que se muestra en la Figura 5.14, como puede verse es bastante homogéneo, aunque presenta una zona de menor emisión en la parte superior.

Las pruebas se realizaron en una zona despejada y libre de obstáculos o elementos que, a priori, distorsionasen demasiado la radiación emitida desde la antena. Por todo esto, se tomó un modelo de atenuación con el cuadrado de la distancia entre la antena del receptor y la antena de la fuente emisora. Además se supuso que el ruido tendría poca importancia, por lo que se omitió la inclusión del mismo en el modelo de la radiobaliza.

En lo que se refiere al receptor, se realizó un montaje con tres receptores para hacerlo lo más homogéneo posible y evitar que la intensidad de la señal se

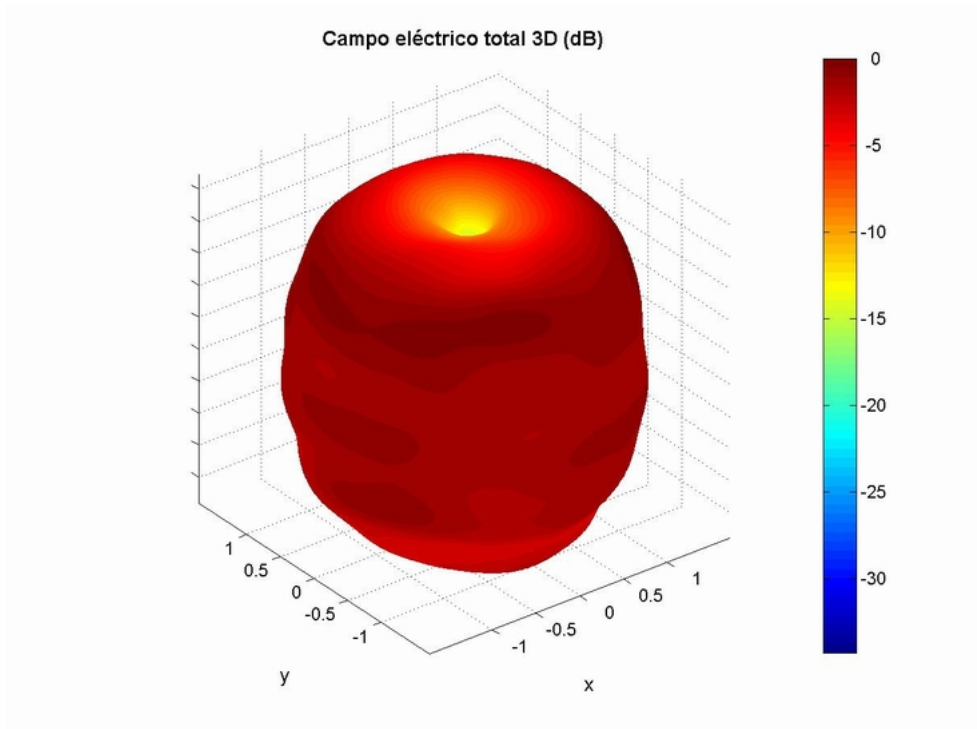


Figura 5.14: Patrón de radiación de las antenas de las radiobalizas

viere afectada por la orientación de la aeronave. El montaje realizado se muestra en la Figura 5.15.

Cuando se realizaron los primeros vuelos de prueba para validar este modelo resultó que el modelo omnidireccional de la antena no ofrecía suficiente precisión y los valores de intensidad real medidos diferían bastante de aquellos estimados a partir del simulador de las radiobalizas.

5.4.4.2. Modelo final de las radiobalizas

Para subsanar y minimizar las diferencias entre el modelo y los valores reales se optó por tomar los datos reales y calcular una función matemática que aproximara estos datos. Esto supuso integrar el patrón de radiación, el ruido y la atenuación de las antenas en una función matemática que no dependía únicamente de la distancia en 3d entre la fuente y el emisor como era el caso del modelo inicial.

En este caso, la función resultante depende de la distancia horizontal y de la altura a la que se encuentre el receptor, con respecto al emisor. Se continúa

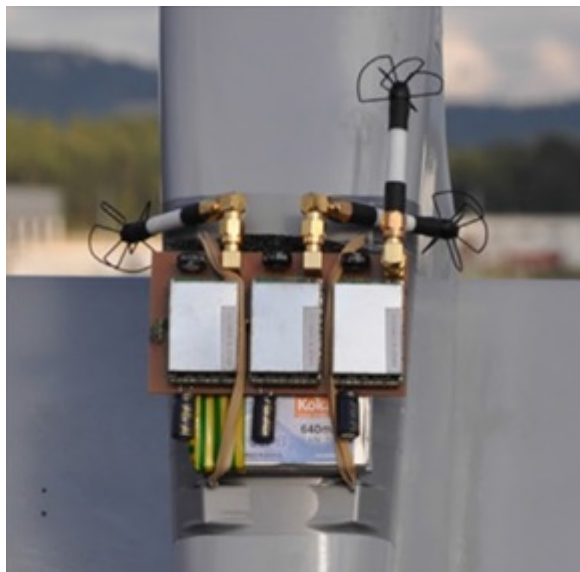


Figura 5.15: Montaje realizado para homogeneizar la recepción

suponiendo que la orientación del UAV no afecta, debido al montaje realizado para el receptor y se obtuvo un modelo que representaba finalmente de forma fiel la intensidad de la señal frente a la posición de los UAV tal y como se muestra en la Figura 5.16.

Para compensar las condiciones concretas del entorno durante la realización de la misión (ruido, humedad, temperatura, etc.) se introducen dos coeficientes que modifican el comportamiento del modelo. Dichos coeficientes se pueden ajustar dinámicamente en tiempo real, en función de los valores medidos por los UAV reales.

5.4.4.3. Conclusiones

El ejemplo presentado apoya la idea de la integración progresiva de la complejidad real, partiendo de un modelo simple para la realización de las pruebas y de su mejora en función de los datos recopilados en los primeros vuelos de prueba. Además, mediante el empleo de la plataforma de realidad mixta, este modelo se puede variar en tiempo real según las condiciones particulares existentes en el momento del vuelo, tomadas de los UAV reales.

Este ajuste dinámico genera un modelo virtual mucho más cercano a la realidad, permitiendo que las medidas de los UAV simulados aporten estimaciones más precisas. Esto contribuye a reducir el reality gap y además a la realización de la misión, ya que mediante las estimaciones de los UAV simulados

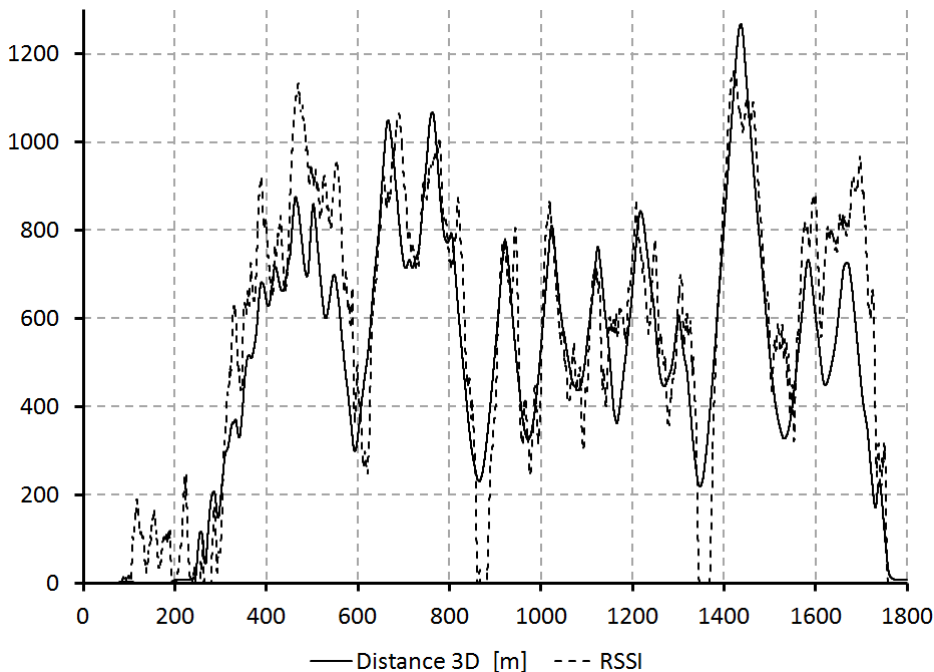


Figura 5.16: Comparación entre la intensidad del modelo y la radiobaliza real

se aporta información basada en los datos reales reales.

La mejora de los modelos de simulación a partir de los datos reales, incluso en tiempo real, contribuye a reducir el reality gap entre el simulador y la realidad, facilitando que el comportamiento de los UAV también sea más aproximado entre los UAV simulados y reales.

5.5. Conclusiones

En este capítulo se han descrito con detalle las características de la plataforma que permite aplicar los principios básicos descritos en la metodología. Así, basándose en la estructura modular, dividida en bloques funcionales, permite cambiar el simulador de los UAV sin que la estrategia de coordinación u otras partes del código se vean afectadas. Asimismo, se aplica el principio de abstraerse de la heterogeneidad del hardware, permitiendo realizar cambios o actualizaciones en el mismo de forma más sencilla y proporcionando la capacidad de tratar de forma homogénea UAV heterogéneos. Finalmente, se ha explicado con mayor profundidad el procedimiento de la introducción progresiva e iterativa de la complejidad real, ilustrando el mismo con un caso de ejemplo de una aplicación real en la que se mejora el modelo de unas radiobalizas basándose en

los datos reales.

En el capítulo siguiente se muestra una implementación concreta en la que se sigue la aproximación propuesta formada por la metodología y la plataforma de realidad mixta para un caso de aplicación concreto.

Capítulo 6

Implementación y pruebas

En este capítulo se presenta un caso de prueba consistente en una implementación concreta de la aproximación propuesta en este trabajo. Para ello se ha seleccionado una aplicación para que los UAV se coordinen y colaboren entre sí con un objetivo común. Para conseguir esta colaboración se ha llevado a cabo el desarrollo de una estrategia de coordinación siguiendo la metodología propuesta en el Capítulo 4 y se hace uso de la plataforma descrita en el Capítulo 5. Mediante este ejemplo se demuestra la utilidad de la aproximación planteada en este trabajo para el desarrollo de las estrategias de coordinación para equipos de UAV.

En primer lugar, en una primera sección se presenta una visión general de la aplicación propuesta, consistente en la localización de una fuente de polución, justificando su elección frente a otras alternativas posibles y describiendo las características generales y los objetivos de la misma.

A continuación, en la sección 6.2 se define el modelo de dispersión de contaminantes empleado y la parametrización utilizada en este caso concreto. En las secciones 6.3, 6.4 y 6.5 se describen respectivamente los UAV reales, los UAV simulados y la estrategia de coordinación planteada; justificándose la adecuación de todos ellos según los requisitos de la aplicación propuesta.

Finalmente en la sección 6.6 se recoge la información más destacada acerca del resultado de las pruebas realizadas con el equipo de UAV, tanto en simulación, como en la realidad.

6.1. Introducción

La aplicación propuesta para el presente trabajo es la localización y caracterización de una fuente de polución. Concretamente, la fuente de polución seleccionada es la chimenea de la central térmica de carbón que se encuentra situada en As Pontes de García Rodríguez y el contaminante a medir es el CO₂. La selección de esta aplicación concreta se ha debido principalmente a los siguientes motivos:

- Constituye una tarea representativa de los usos potenciales en vigilancia ambiental: la vigilancia ambiental es uno de los campos con mayor potencial para los UAV y esta aplicación reúne varias características representativas del mismo, comunes con otras aplicaciones como la vigilancia de incendios forestales, la monitorización del agua de embalses o la vigilancia de cultivos para la agricultura de precisión.
- El área de trabajo es extensa y el fenómeno se produce en el medio aéreo: la dispersión de contaminantes en la atmósfera se extiende a lo largo de varios kilómetros alrededor de la chimenea. Así, un equipo de UAV permite barrer rápidamente este espacio frente a la opción de un único UAV.
- Se trata de un fenómeno dinámico: resulta óptima la toma de datos simultáneos en más de un punto para caracterizar con garantías la fuente de polución y evitar cambios en las condiciones atmosféricas, como el viento, que puedan distorsionar la dispersión entre las medidas.
- Se dispone de sensorización con velocidad y precisión suficiente: la mayoría de los sensores actuales para contaminantes aéreos con un tamaño reducido son demasiado lentos o no presentan la precisión suficiente para la medida de la concentración presente de los contaminantes en el exterior.
- Repetibilidad y existencia de datos: frente a otros eventos cuya ocurrencia e intensidad es variable y difícilmente predecible como la aparición de focos de incendio o contaminación en embalses, la central térmica opera la mayor parte del año en unas condiciones de funcionamiento similares y existen datos acerca del proceso que facilitan la realización del modelo.
- Facilidad logística: la planta seleccionada se encuentra en As Pontes, en la zona norte de Galicia y a unos 40km de Ferrol, por lo que se facilita la logística de los desplazamientos y por tanto, la realización de las operaciones con los UAV reales frente a otras posibles aplicaciones.

Descripción de la central térmica La central en cuestión es una central térmica convencional situada en el municipio de As Pontes de García Rodríguez,



Figura 6.1: Vista de la central térmica de As Pontes

al noreste de la provincia de A Coruña. Cuenta con cuatro grupos de generación con una potencia total de 1400 MW. Diseñada originalmente para consumir el lignito extraído en la minería local, fue adaptada entre 1993 y 1996 para el consumo de una mezcla de éste con hulla importada en la misma proporción. Desde que se cerró la mina de lignito en el año 2007, tan solo consume hulla importada.

Su chimenea, con una altura de 356 m y un diámetro de 36,5 m en la base, es una de las más grandes del mundo y la más alta de Europa. Dentro de esta estructura de hormigón aloja cuatro conductos metálicos, uno para cada grupo generador.

Descripción del contaminante El contaminante seleccionado para esta aplicación es el dióxido de carbono, un gas incoloro e inodoro que está presente en la atmósfera y puede tener origen natural o industrial. Dentro de las fuentes naturales están todos los organismos aerobios ya que se trata de un producto de la respiración de éstos y además es liberado por rocas carbonatadas al diluirse en aguas y ácidos. En lo que respecta a las fuentes industriales, se produce por la combustión de combustibles fósiles como el carbón, el petróleo y el gas natural y se trata de un gas que contribuye de forma importante al efecto invernadero.

La quema de combustibles fósiles desde la Revolución Industrial ha aumentado rápidamente su concentración en la atmósfera hasta situarlo alrededor de

las 400 ppm, lo que ha contribuido al actual fenómeno de cambio climático y calentamiento global. Es además la principal causa de la acidificación del océano, ya que se disuelve en el agua para formar ácido carbónico.

6.2. Caracterización del modelo de pluma

La predicción de la concentración de una sustancia en la zona que rodea al punto de emisión es un tema de gran interés en contaminación atmosférica. Las chimeneas son los conductos para dar salida a la atmósfera a los gases resultantes de la combustión u otra reacción química. Su principal función es la de facilitar la dispersión de los contaminantes, evitando así las concentraciones puntuales en un área determinada.

La emanación de los gases de una chimenea recibe el nombre de pluma y su altura está determinada por la velocidad y empuje de los gases que salen por la chimenea. La dispersión posterior depende de diversos factores como la temperatura, la presión atmosférica, la presencia de precipitaciones y el viento. Un contaminante emitido a la atmósfera desde una chimenea es transportado en la dirección del viento predominante y dispersado por movimientos de aire perpendiculares al viento así como por turbulencia.

Altura efectiva Los gases emitidos por las chimeneas se mezclan con el aire del ambiente. Esta mezcla del aire ambiental en la pluma se denomina arrastre. Durante el arrastre en el aire, la pluma aumenta su diámetro mientras viaja a sotavento.

Al entrar en la atmósfera, estos gases tienen un *momentum*. Muchas veces se calientan y se vuelven más cálidos que el aire externo. En estos casos, los gases emitidos son menos densos que el aire exterior y, por lo tanto, flotantes. La combinación del momentum y la flotabilidad de los gases hacen que estos se eleven. Este fenómeno, conocido como elevación de la pluma, permite que los contaminantes emitidos al aire en esta corriente de gas se eleven a una altura mayor en la atmósfera.

Al estar en una capa atmosférica más alta y más alejada del suelo, la pluma experimenta una mayor dispersión antes de llegar a este. La altura final de la pluma, conocida como altura efectiva de chimenea H , es la suma de la altura física de la chimenea h y la elevación de la pluma Δh . En realidad, la elevación de la pluma se estima a partir de la distancia existente hasta la línea central imaginaria de la pluma y no hasta el borde superior o inferior de esta. La elevación de la pluma depende de las características físicas de la chimenea y del efluente, tal y como se muestra en la Figura 6.2.

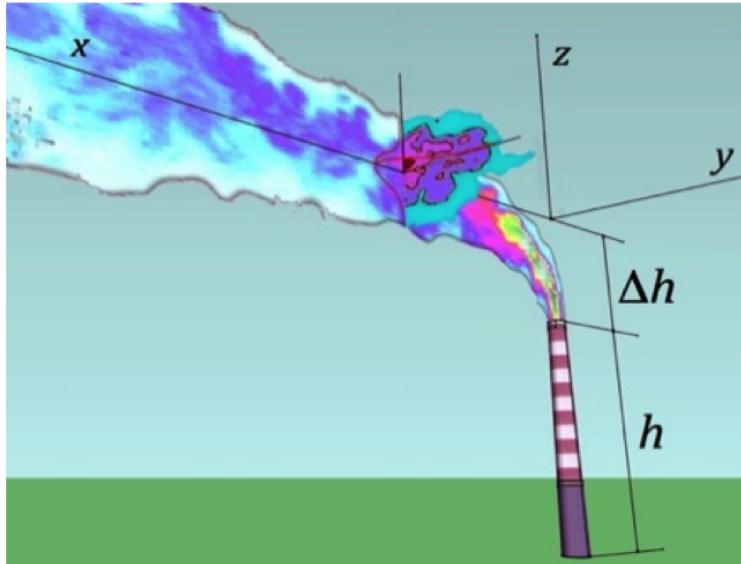


Figura 6.2: Disposición de la pluma

La diferencia de temperatura entre el gas de la chimenea y el aire ambiental determina la densidad de la pluma, que influye en su elevación. Además, la velocidad de los gases de la chimenea, que es una función del diámetro de la chimenea y de la tasa volumétrica del flujo de los gases de escape, determina el momentum de la pluma.

Momentum y flotabilidad La condición de la atmósfera, incluidos los vientos y el perfil de la temperatura a lo largo del recorrido de la pluma, determina en gran medida la elevación de la pluma. Dos características de ésta influyen en su elevación: el momentum y la flotabilidad.

La velocidad de salida de los gases de escape emitidos por la chimenea contribuyen a la elevación de la pluma en la atmósfera. Este momentum conduce el efluente hacia el exterior de la chimenea a un punto en el que las condiciones atmosféricas empiezan a afectar a la pluma. Una vez emitida, la velocidad vertical inicial de la pluma disminuye rápidamente debido al arrastre producido cuando adquiere un momentum horizontal. Este fenómeno hace que la pluma se incline. A mayor velocidad del viento, más horizontal es el momentum que adquiere la pluma.

Por lo general, dicha velocidad aumenta con la distancia sobre la superficie de la Tierra. A medida que la pluma continúa elevándose, los vientos más fuertes hacen que se incline aún más. Este proceso persiste hasta que la pluma parece horizontal al suelo. El punto donde la pluma parece llana puede ser una distancia

considerable de la chimenea a sotavento. La velocidad del viento es importante para impulsar la pluma. Cuanto más fuerte, más rápido es el serpenteo de la pluma.

El arrastre del aire ambiental hacia la pluma por acción del viento le resta flotabilidad muy rápidamente, de modo que durante los días con mucho viento la pluma no se eleva muy alta sobre la chimenea.

Los modelos de dispersión aportan estimaciones de la concentración de contaminantes basados en formulaciones matemáticas que consideran varios de los factores antes citados.

6.2.1. Descripción del modelo

Seleccionar un modelo adecuado de dispersión de contaminantes es de suma importancia para tener un mundo virtual capaz de representar con precisión las características relevantes del mundo real. En este sentido, la selección de un modelo de alta precisión puede parecer una elección adecuada. Sin embargo, esta característica entra en conflicto con uno de los requerimientos importantes de este modelo, ya que debe ser ligero, desde un punto de vista computacional, y fácilmente configurable. Esta última característica es muy importante no sólo para facilitar su implementación sino también para hacer posible cambiar el modelo en tiempo real para adaptarlo a las condiciones cambiantes en el mundo real.

Por estas razones, se ha recurrido a modelos paramétricos de dispersión de plumas en lugar de ir a modelos más sofisticados como los basados en la Dinámica de Fluidos Computacional (CFDs). En general, estos últimos tienen una mayor precisión que los paramétricos, pero su cálculo es demasiado lento para que sea apropiado para ser utilizado en un sistema en línea. La idea general es elegir el modelo más simple que sea capaz de dar la precisión requerida.

El modelo más común de contaminación atmosférica es el modelo de pluma gaussiano, que determina que, una vez que la pluma está completamente desarrollada y para una posición dada a lo largo del eje x , la dispersión de los contaminantes a lo largo de los ejes y y z sigue una distribución gaussiana.

Se trata de un modelo paramétrico en el que todo el proceso de dispersión de la pluma se rige por los siguientes parámetros:

- Flujo de masa del contaminante que sale de la chimenea: Q
- Diámetro de salida de la chimenea: D
- Altura de la chimenea: h

Surface wind speed (m/s)	Daytime isolation			Nighttime	
	Strong	Moderate	Slight	≥4/8 cloud	≤3/8 cloud
<2	A	A-B	B	-	-
2-3	A-B	B	C	E	F
3-5	B	B-C	C	D	E
5-6	C	C-D	D	D	D
>6	C	D	D	D	D

Tabla 6.1: Clases de estabilidad atmosférica según Pasquill

- Velocidad del viento a la altura efectiva de la pluma: u
- Parámetros de estabilidad de la atmósfera que tienen en cuenta la cantidad de turbulencia atmosférica presente

Comúnmente estos efectos atmosféricos se explican mediante el método desarrollado por Pasquill, que clasifica la estabilidad de la atmósfera en seis clases llamadas de A a F, siendo la clase A la más inestable y la clase F la más estable. Las condiciones meteorológicas que definen cada una de estas clases se proporcionan en la Tabla 6.1.

Se ha de incluir un modelo de elevación de la pluma como parte del modelo Gaussiano. Se ha recurrido al modelo de elevación de pluma de Briggs. Este modelo es simple y ampliamente utilizado. Establece que, para condiciones atmosféricas inestables o neutras (clases de Pasquill que van de A a D) la distancia de la subida final de la pluma es $x_f = 3,5x^*$, donde:

$$x^* = 14F^{5/8}, \text{ cuando } F < 55m^4s^{-3}$$

$$x^* = 34F^{2/5}, \text{ cuando } F \geq 55m^4s^{-3}$$

Siendo F el flujo de flotabilidad definido como:

$$F = gV_s r^2 (\rho_a - \rho / \rho_a)$$

Donde:

Stability class	Rural exponent	Urban exponent
A	0.07	0.15
B	0.07	0.15
C	0.10	0.20
D	0.15	0.30
E	0.35	0.30
F	0.55	0.30

Tabla 6.2: Perfil de viento en función de la clase de estabilidad atmosférica.

- g la aceleración gravitatoria.
- V_s la velocidad de salida del gas de la chimenea.
- r el radio de salida de la chimenea.
- ρ_a la densidad del aire ambiente.
- ρ la densidad del gas de la chimenea.

Para estas condiciones atmosféricas el modelo de Briggs da la subida final de la pluma como:

$$\Delta h = 1,6F^{1/3}(3,5x^*)^{2/3}u^{-1}$$

La velocidad del viento u , a la altura efectiva de la chimenea H , se puede ajustar con la siguiente ecuación:

$$u = u_0(H/z_0)^p$$

Donde:

- u_0 es la velocidad del viento a la altura z_0
- z_0 es la altura del anemómetro
- p se define en la Tabla 6.2 en función de la clase de estabilidad para zonas rurales y urbanas.

El modelo gaussiano para la pluma expresa la distribución de la concentración de contaminantes c en el campo de flujo del interior de la pluma como:

Pasquill type	σ_y (m)	σ_z (m)
A	$0.22x(1 + 0.0001x)^{-0.5}$	0.20x
B	$0.16x(1 + 0.0001x)^{-0.5}$	0.12x
C	$0.11x(1 + 0.0001x)^{-0.5}$	$0.08x(1 + 0.0002x)^{-0.5}$
D	$0.08x(1 + 0.0001x)^{-0.5}$	$0.06x(1 + 0.0015x)^{-0.5}$
E	$0.06x(1 + 0.0001x)^{-0.5}$	$0.03x(1 + 0.0003x)^{-0.5}$
F	$0.04x(1 + 0.0001x)^{-0.5}$	$0.01x(1 + 0.0003x)^{-0.5}$

Tabla 6.3: Sigma para un área rural

$$c(x, y, z) = \frac{Q}{2\pi u \sigma_y \sigma_z} \exp\left(-\frac{y^2}{2\sigma_y^2}\right) \exp\left(-\frac{z^2}{2\sigma_z^2}\right)$$

Donde σ_y y σ_z son las desviaciones estándar gaussianas a lo largo de los ejes y y z en una posición x dada. En la literatura se pueden encontrar diferentes maneras de obtener estos dos parámetros. La formulación semi empírica de Briggs es generalmente aceptada y es la que se utiliza en el presente trabajo. La Tabla 6.3 presenta las expresiones de estas funciones Briggs sigma para un área rural y para distancias entre 10^2 y 10^4 m de la fuente.

6.2.2. Parametrización para el caso de prueba

En esta sección se realizan los cálculos de los distintos coeficientes para el caso de la chimenea de la central térmica de As Pontes. Por conveniencia, se mide el CO_2 en el presente trabajo, aunque obviamente cualquier otro contaminante podría utilizarse en lugar de CO_2 simplemente sustituyendo los sensores a bordo.

Tal y como se definió en la introducción, actualmente la central eléctrica de carbón quema únicamente hulla importada. El factor de emisión de CO_2 EF de este tipo de carbón para la combustión estacionaria en la industria energética es de 96,1 kg de CO_2 por GJ sobre una base calorífica neta. La central cuenta con cuatro grupos independientes de generación de energía de 350 MW cada uno. Su consumo específico es de 2427 kcal / kWh, lo que equivale a una eficiencia neta η del 35,43 %.

Así, la emisión específica de la planta es $q = EF\eta^{-1} = 0,271$ kg / s de CO_2 por MW de potencia generada. Por lo tanto, la emisión de CO_2 a una potencia dada es $Q = qW$ y la máxima emisión se alcanza a plena potencia:

$$Q_{MAX} = qW_{MAX} = 379.8 \text{ kg/s de } \text{CO}_2$$

La chimenea tiene una altura de 356 m sobre el nivel del suelo y aloja cuatro conductos, uno para cada grupo generador, cada uno de ellos con un diámetro D_L de 6,34 m. La chimenea ha sido diseñada para tener una velocidad de escape de gases de combustión inferior a 30 m/s. Teniendo en cuenta la composición y el mayor valor de calentamiento del carbón importado, se supone una temperatura media de gas de emisión de 190°C. Para el propósito de este trabajo, se considera que la chimenea tiene una única salida con un diámetro equivalente D que es una función del número n de grupos que están funcionando:

$$D = \sqrt{n}D_L$$

El número de unidades de potencia de trabajo se puede detectar fácilmente observando el vapor que sale de las cuatro torres de refrigeración. Para el resto de parámetros se consideran las condiciones correspondientes al día en que se realizaron las pruebas, es decir:

- Temperatura ambiente: 20°C
- Velocidad y dirección del viento medida (a 10 m sobre el nivel del suelo) 7,8 m/s y 72°
- Condición atmosférica: aislamiento diurno moderado
- Clase Pasquill de estabilidad atmosférica: D

Además, se ha comprobado visualmente que la central estaba trabajando con los cuatro grupos. Se supone inicialmente que los grupos funcionan a una potencia nominal del 80% y que la velocidad de escape de los gases de combustión es de 20 m/s. Estas dos condiciones son aproximaciones cuyas inexactitudes se minimizan más adelante utilizando los datos medidos por los UAV reales.

Con estas condiciones, y de las ecuaciones anteriores se obtiene: $Q = 303,8$ kg/s, $D = 12,68$ m, $x^* = 712$ m y $\Delta h = 278$ m. Así, la altura efectiva estimada de la chimenea es de $H = 634$ m. La figura 6.3 muestra la configuración inicial del modelo según estos parámetros. Como se ha mencionado anteriormente, esta representación se realiza en Google Maps. Se puede ver la pluma emergiendo desde la posición de la chimenea y dispersándose en la dirección del viento hacia el oeste. El círculo representa el área de trabajo, donde los UAV virtuales pueden volar. El color de la pluma, en una gama del rojo al verde representa la concentración de CO₂ presente a la altura de la pluma.

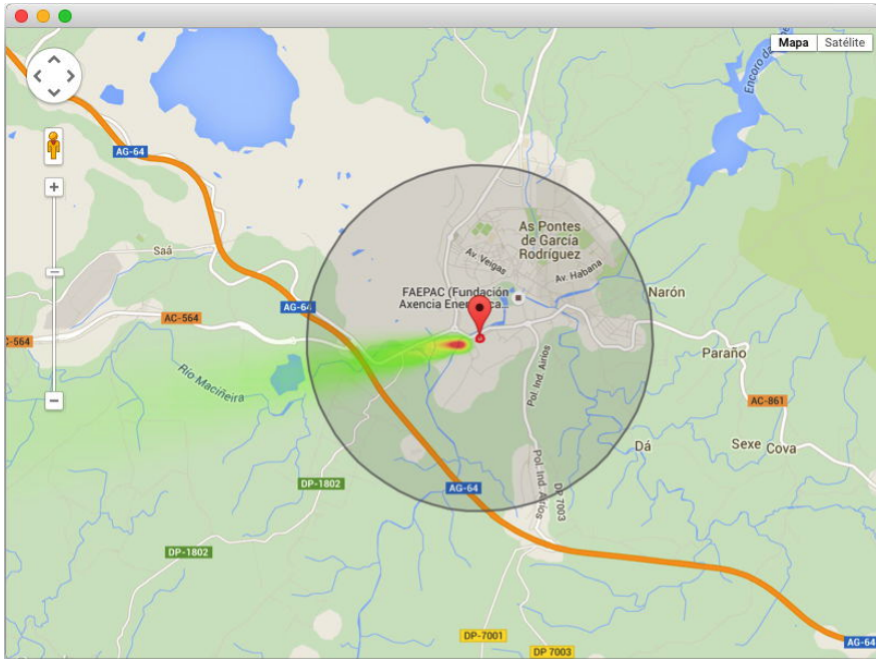


Figura 6.3: Visualización del modelo de la pluma sobre Google Maps

6.3. UAV reales

Como se ha dicho anteriormente, la plataforma de realidad mixta propuesta puede emplearse para diversas aplicaciones con escenarios y objetivos diferentes. Sin embargo, este trabajo se presenta una aplicación basada en la dispersión de contaminantes atmosféricos. La búsqueda se realiza mediante un equipo heterogéneo de UAV reales y simulados con sensores para ese contaminante específico. Con este objetivo en mente, se ha diseñado y construido un conjunto de UAV reales según los criterios de diseño siguientes:

- Seguridad primero: el diseño debe primar la seguridad sobre las características técnicas en todas las fases de diseño. Desde el control de vuelo seguro (seguridad activa o primaria), con el fin de evitar los problemas de seguridad, hasta los materiales y el factor de forma del diseño final (seguridad pasiva o secundaria), para hacer el UAV potencialmente menos dañino.
- Suficiente alcance y autonomía de vuelo.
- Suficiente capacidad de carga útil.
- Fácil de desplegar: teniendo en cuenta su tamaño y complejidad, los UAV y

los equipos terrestres asociados deben ser fáciles de desplegar en el campo.

- Fácil de volar en condiciones atmosféricas suaves o medias.
- Flexible para permitir una fácil adaptación a nuevos conjuntos de sensores, actuadores y unidades de control.
- Esquema de control enfocado a facilitar la integración en el mundo virtual, acercando el comportamiento de los UAV reales y simulados.

6.3.1. Selección de los UAV

A la hora de seleccionar los UAV que forman parte de la plataforma se han tenido en cuenta diversos aspectos que condicionan su desempeño tratando de maximizar las posibilidades de uso y la seguridad, a la vez que se minimiza el coste.

Uno de los aspectos fundamentales ha sido la eficiencia, se trata de maximizar tanto la autonomía como el alcance de cada uno de los UAV, por lo que han de tener una gran superficie alar y un perfil aerodinámico que permita el vuelo a bajas velocidades. Otro de los aspectos importantes a tener en cuenta es el material del que está hecho, por razones de seguridad y para minimizar las reparaciones el material debería ser flexible, capaz de absorber los impactos mediante la deformación y volver a su forma original, tal y como lo hacen las espumas plásticas como el polipropileno expandido (EPP).

Concretamente y, en base a los criterios de diseño enunciados en el apartado anterior, para la búsqueda de contaminantes, los UAVs necesitan recorrer distancias intermedias a grandes, en el rango de algunas decenas de kilómetros, y permanecer en vuelo durante más de 1 hora, mientras buscan la huella del contaminante. En la actualidad los multirrotores no resultan adecuados bajo este tipo de restricciones, por lo que se ha seleccionado un equipo de UAV de ala fija.

Con respecto a los requisitos de la carga útil, los sensores para contaminantes, así como su electrónica asociada, son pequeños por lo que no hay necesidad de un compartimento de carga grande en una estructura de fuselaje. Por ello, se ha optado por seleccionar alas volantes, con todos sus componentes integrados en el ala.

Como diseño básico, se utilizó un ala volante comercial *Hércules* y se modificó profundamente para satisfacer los criterios de diseño. Las alas están fabricadas en espuma EPP, un material ligero, duradero y sobre el que resulta fácil trabajar y menos peligroso en caso de un percance. La Figura 6.4 muestra algunos de los UAV utilizados actualmente para las pruebas.



Figura 6.4: Parte del equipo de UAV reales

6.3.2. Especificaciones de los UAV diseñados

El UAV diseñado tiene una envergadura de 1,98 m, y una cuerda alar media de 0,47 m. Se ha trabajado activamente sobre él para convertirlo en una plataforma estable: un ángulo de flecha de 25 grados ayuda en la estabilidad de la guiñada, un perfil aerodinámico con réflex en el cabeceo y un diedro de 3 grados en el alabeo. Además, el ala ha sido diseñada con una torsión alar de 4,5 grados para suavizar el comportamiento dinámico y convertirla en una plataforma de vuelo noble, que sea fácil de manejar en maniobras de alto cabeceo y/o de baja velocidad como pérdidas, despegues y aterrizajes.

Se ha añadido un pod, también de EPP, por debajo de la panza del UAV, de forma que sirva como un punto de montaje para sensores externos y eleve las puntas de las hélices del suelo. Esto, junto con unos winglets en los marginales, permite al UAV despegar y aterrizar en pequeñas superficies de hierba o tierra, y no necesita dispositivos de apoyo como catapultas o paracaídas para su operación. Aunque no es habitual en un ala volante de este tamaño, tiene dos motores en la parte delantera, y dos estabilizadores verticales con las derivas detrás de cada motor, en su flujo de empuje, en el borde de salida del ala.

Todos estos parámetros de diseño han sido seleccionados con el fin de hacer que el avión sea fácil de operar desde entornos hostiles como pequeñas superficies sin acondicionamiento. Este hecho resulta clave debido al complejo despliegue requerido para un equipo de UAV como este y su equipo terrestre asociado.

El ala tiene una superficie total de $0,93 \text{ m}^2$ y pesa $2,96 \text{ kg}$, listo para funcionar, por lo que tiene una carga alar de $3,18 \text{ kg/m}^2$. Cuenta con dos motores de imanes permanentes sin escobillas de 345 W cada uno, que tienen un factor de 1100 kV . Se alimenta a partir de dos baterías de polímero de litio de $11,1 \text{ V}$ unidas en paralelo, lo que suma 10 Ah o 400 kJ .

Con la carga útil utilizada en este trabajo, el avión mantiene una velocidad de crucero cercana a los 10 m/s con un consumo de corriente total de $9,3 \text{ A}$, incluyendo los motores eléctricos, la aviónica y los sensores. Esto hace que el vuelo pueda durar hasta 65 minutos y se pueda obtener un alcance cercano a los 40 km . Dado que la carga del ala se encuentra todavía baja se podrían incrementar fácilmente esas cifras aumentando el número de baterías embarcadas.

6.3.3. Electrónica de vuelo integrada

Para dotar de autonomía al UAV se ha integrado en el modelo antes descrito un volumen importante de electrónica para el control y las comunicaciones del mismo. Se ha realizado una selección y combinación de módulos comerciales con electrónica y programación desarrollada específicamente para este proyecto.

Visión general En la Figura 6.5 se muestra un esquema general de los diferentes componentes electrónicos integrados en cada uno de los UAV. En las secciones posteriores se describe con más detalle la función de cada uno de los elementos de este esquema.

6.3.3.1. Electrónica de control

El principal objetivo de la electrónica de control es dotar a los UAV de autonomía, es decir, que sean capaces de controlar el vuelo y trazar trayectorias de forma controlada sin intervención humana.

Para ello en primer lugar se debe seleccionar el hardware con el que se pretende trabajar, teniendo en cuenta que debe adaptarse a los requerimientos impuestos por la estructura de software a desarrollar sobre este. A continuación, trabajar sobre el software de más bajo nivel e ir desarrollando el de nivel superior sobre este.

Control de vuelo básico y unidad de control La selección de la unidad de control determina en gran medida el piloto automático, que es la parte del programa de control encargado de convertir las consignas provenientes del control de vuelo básico, en señales de control para la entrega de potencia del motor y para el movimiento de las superficies de control.

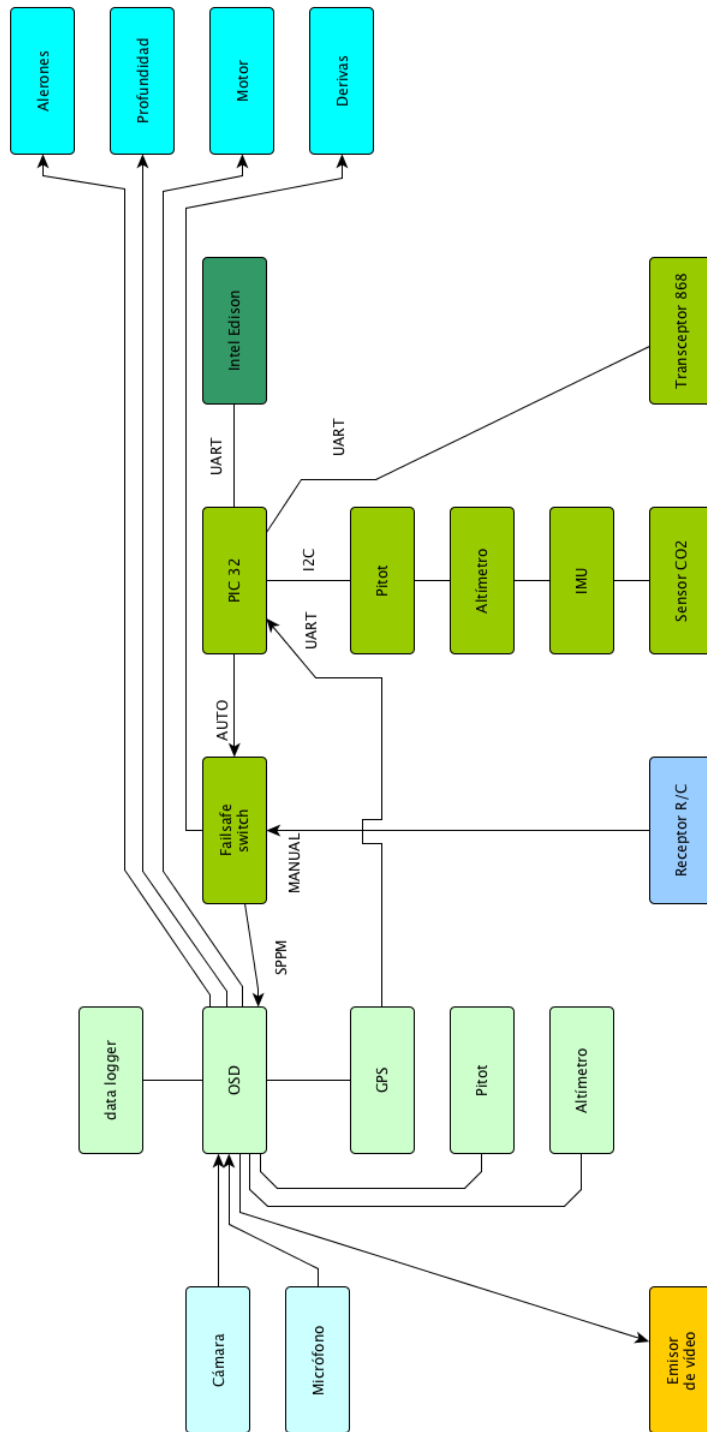


Figura 6.5: Esquema del conexionado de la electrónica del UAV

En este caso se ha optado por incluir una unidad de control comercial, aunque tras el proceso de mejora que se detalla en la sección 5.3.3 la mayor parte del control de vuelo básico recae sobre el microcontrolador PIC32 desarrollado para esta aplicación.

Se ha mantenido sin embargo, la unidad de control comercial debido a que aporta la función de vuelta a casa autónoma y la recopilación de los datos de telemetría. Para ello requiere conocer la posición del UAV, su altitud y velocidad con respecto al viento, para lo que cuenta con un GPS, un altímetro barométrico y un tubo pitot.

Tomando en consideración la seguridad como una de las prioridades, un piloto humano puede tomar el control del avión en todo momento, utilizando el enlace RC y un transmisor RC para cada avión. Los enlaces de vídeo y audio junto con la información de estado ayudan al piloto a controlar el UAV en primera persona en caso de que se encuentre fuera del alcance visual. La electrónica dedicada, diseñada alrededor de componentes analógicos discretos, vigila la integridad de los enlaces de radio y es capaz de transferir el control completo del UAV si es necesario.

En lo referente al microcontrolador PIC32, se lo ha instrumentado también con un GPS y un altímetro, para que conozca la posición del UAV; con una IMU, para la orientación y un tubo pitot para conocer la velocidad con respecto al viento. Además, se ha incorporado el sensor de CO₂ para la aplicación de medición de contaminantes.

El esquema del control del piloto automático se representa en la Figura 6.6. El control de vuelo básico proporciona las consignas, la información real se toma de los sensores integrados en el UAV y de la diferencia entre ambos se obtiene una señal, que una vez corregida y filtrada es la encargada de controlar las superficies de control y el motor.

Control autónomo inteligente Con la finalidad de disponer de suficiente potencia computacional para la ejecución de estrategias de coordinación complejas en tiempo real se ha integrado en los UAV reales un microprocesador Intel Edison Compute Module que cuenta con un procesador de dos núcleos Intel Atom a 500 MHz, 1 GB RAM DDR3 y 4 GB eMMC flash en un volumen de 127x72 mm.

Esta unidad de computación se comunica mediante una conexión UART con el microcontrolador donde se ejecuta el control de vuelo básico, intercambiando así la información requerida entre ellos.

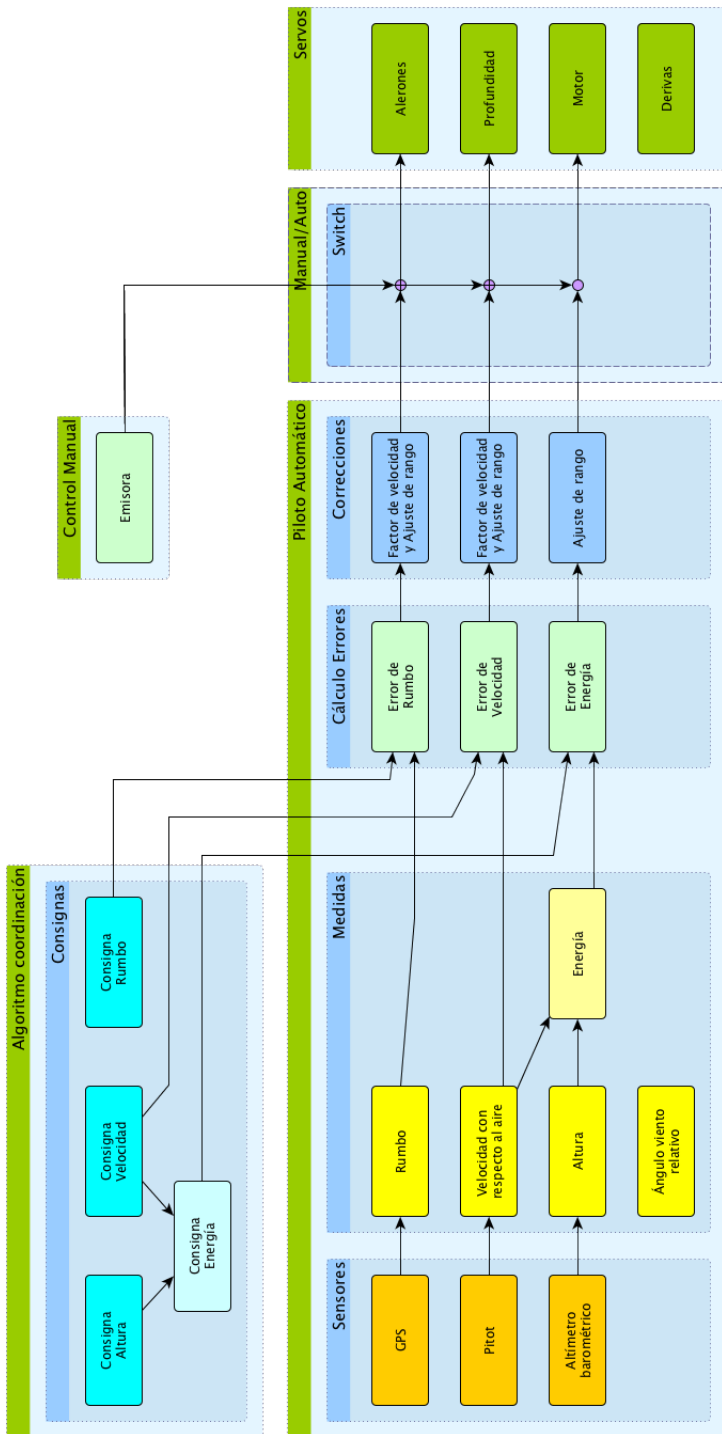


Figura 6.6: Esquema de la lógica de control del UAV

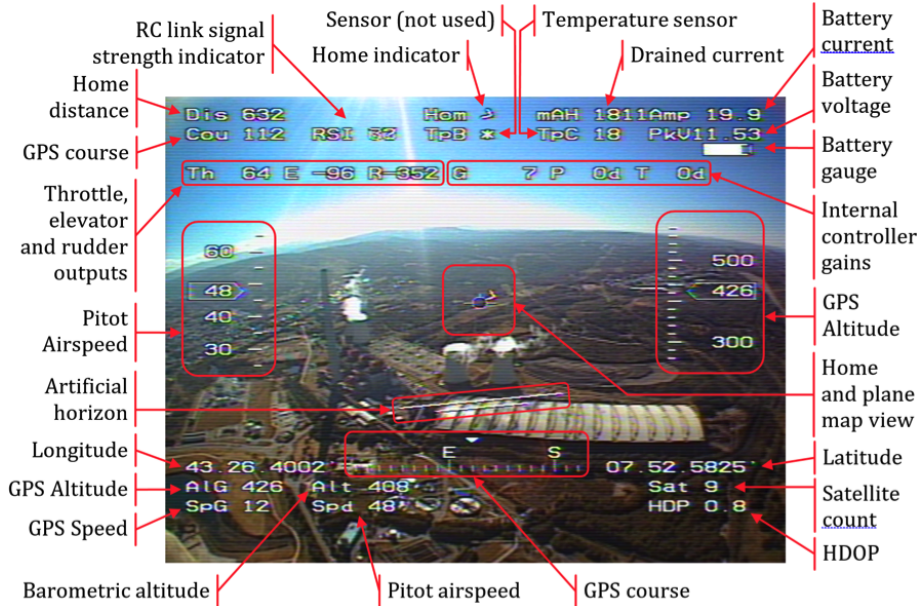


Figura 6.7: Explicación de los parámetros del OSD

6.3.3.2. Comunicaciones

Existen tres canales de comunicación inalámbricos entre cada uno de los UAV y la estación de tierra. El primero es el enlace de radio control, que permite que un piloto humano controle el avión desde un transmisor RC estándar. El lado RF del transmisor es una unidad programable de código abierto de largo alcance, con salto de frecuencia, que opera en la banda de 433 MHz. Este enlace es bidireccional, aunque en este trabajo únicamente se utiliza el enlace ascendente (tierra a UAV). El rango esperado está en las decenas de kilómetros.

El segundo enlace es el enlace de vídeo y sirve para dos propósitos. Por un lado, se utiliza para descargar vídeo y audio analógicos en tiempo real desde la cámara y el micrófono de a bordo. Se superpone a la imagen de la cámara una información exhaustiva de la electrónica integrada, como los datos del GPS, la altitud barométrica, la velocidad del Pitot, el estado de la batería, el consumo de corriente, la ubicación del plano geográfico, etc. Esto ayuda al operador humano a controlar que el UAV sigue las trayectorias previstas y visualizadas en la estación de tierra y le permite tomar el control del avión en caso de un percance.

Por otro lado, la electrónica de a bordo incorpora todos los datos recogidos disponibles en el espacio de teletexto del enlace de vídeo, de tal manera que el equipo de tierra pueda extraer y utilizar esta información en tiempo real.

Este enlace se encuentra en la banda de 1,2 GHz y utiliza antenas de polarización circular, tanto en el UAV como en tierra, para hacer que el enlace sea menos sensible a los cambios de orientación de los UAV y para ayudar a evitar artefactos de RF como el multipath. Las antenas seleccionadas son *skew planar wheel*, tanto en el UAV como en la estación terrestre. Estas antenas son omnidireccionales, con una baja directividad de 3 dB.

Por razones de seguridad, se coloca otra antena de 8,6 dB de polarización circular en la estación terrestre, dirigida permanentemente al avión mediante un mecanismo de seguimiento panorámico que utiliza la información GPS incrustada en la señal de vídeo. Un circuito de selección en función de la intensidad de señal ayuda a mantener el enlace de vídeo en caso de que el avión abandone el alcance de las antenas omnidireccionales, eligiendo el mejor receptor.

La tercera vía de comunicación es el enlace de datos. Duplica la descarga de datos del enlace de vídeo, por razones de seguridad y, al ser bidireccional, al mismo tiempo sirve como fuente principal de comunicación para las operaciones de UAV automatizadas. Este enlace se encuentra en la banda legal de 868 MHz y un largo alcance, de nuevo en las decenas de kilómetros.

6.3.3.3. Sensor de CO₂

Para la realización de esta aplicación resulta indispensable contar con sensores capaces de medir la intensidad del contaminante. Dado que en este caso se optó por la medida del CO₂ se realizó una búsqueda de las opciones disponibles comercialmente para ello. Finalmente se optó por un sensor iAQ-Core de la marca ams debido a su pequeño tamaño de 17 x 15mm y a su resolución, ya que es capaz de medir entre las 450 y las 2000 ppm de CO₂, lo que abarca el rango esperado en esta aplicación.

6.3.4. Seguridad y prevención de accidentes

Volar un UAV tiene inherentes problemas de seguridad, comunes a la mayoría de las aeronaves. Volar un equipo heterogéneo de UAV aumenta en gran medida la posibilidad de que ocurra este tipo de eventos. Se ha tomado especial cuidado para superar situaciones de seguridad, desde el momento del diseño.

Comenzando con el material seleccionado para construir los UAV, hasta el modo en que se ha implementado el control, como se ilustra en las siguientes secciones. Todas las superficies de control están duplicadas: dos elevones en cada lado del ala con servomotores separados, dos motores y dos derivas, con su propio servomotor independiente.

También la estructura del control del UAV está diseñada desde cero con

la seguridad en mente. Se han implementado diferentes medidas de seguridad dentro de los controles.

La primera de ellas es que el control puede ser asumido en cualquier momento por un piloto humano. Por esta razón, cualquiera de los UAV que están volando simultáneamente es supervisado continuamente por un operador en su estación de tierra, y un piloto debe estar siempre listo para tomar el control.

Una segunda medida de seguridad es un regreso seguro a casa que se implementa a un nivel más alto como se explica más adelante y que asegura que el UAV no salga de control en caso de una pérdida completa de comunicación con su base.

Está claro que la consecuencia de la implementación de todos estos requisitos de seguridad a nivel de diseño no es sólo que la complejidad del sistema aumenta sino también que reducen su operabilidad. Esto repercute negativamente en la logística de la operación, y principalmente al tratar de volar varios UAV simultáneamente. Se necesita una persona cualificada por cada UAV más algunos coordinadores para mantener estos niveles de seguridad.

Además, estos requisitos de seguridad también tienen como resultado una reducción en el rendimiento de los UAV. De hecho, como resultado de reforzar la seguridad en el nivel de diseño, se ha empleado un material estructural espumoso y una carga muy ligera del UAV. Esto ha reducido la velocidad máxima de la aeronave podría ser una limitación para algún tipo de operaciones.

Además de todas estas preocupaciones de seguridad, también deben tenerse en cuenta las normativa vigente. Aunque esta varía según el país, siempre se introducen límites en el uso de UAVs. Últimamente, la proliferación en el uso de drones combinada con algunos temores de seguridad ha impulsado la aplicación de regulaciones más restrictivas. Estos cambios en las regulaciones deben tenerse en cuenta en la planificación de cualquier campaña de vuelo UAV.

Como se ha dicho, el control autónomo inteligente conoce en tiempo real la posición, velocidad, rumbo y trayectoria planificada de cada UAV. Esto hace posible evitar eficazmente el vuelo de UAV reales en zonas restringidas, sean estas debidas a restricciones por la regulación o a restricciones físicas.

Ejemplos de estas restricciones son las áreas pobladas, las carreteras, el entorno de la chimenea o las zonas de condiciones de vuelo especialmente duras, tales como el lado de sotavento de las colinas en situaciones ventosas, con ráfagas de turbulencia fuertes. En cuanto al entorno de la chimenea, y siendo los UAV de espuma EPP, existe un riesgo real de afectar la integridad estructural del avión volando en la mezcla de gases a altas temperaturas, cuando está demasiado cerca de la salida de la chimenea.

Esta preocupación causó la exclusión de la zona de entorno de la chimenea

para los aviones reales. Sin embargo, contrariamente a los temores iniciales, las lecturas de la temperatura del aire de los sensores integrados en los UAV reales fueron bajas. En todos los experimentos realizados mostraron una variación menor de 4°C con respecto a la temperatura en tierra y en ningún caso se superaron los 25°C.

6.4. UAV simulados

Una vez diseñados los UAV reales se ha de modelar su contrapartida virtual para llevar a cabo las pruebas en la plataforma de realidad mixta. Para ello en primer lugar se selecciona el simulador a emplear, dado que la plataforma de realidad mixta permite el uso de cualquier simulador, siempre y cuando se pueda conectar con el resto del sistema mediante paquetes UDP. Posteriormente se introducen en el simulador los datos de los UAV reales diseñados para conseguir un funcionamiento similar entre ambos.

6.4.1. Elección del software de simulación

Se realizó una selección entre los simuladores existentes en la que se valoró especialmente la fidelidad a la realidad, la facilidad para integrar nuevos módulos conectados en tiempo real al simulador y la posibilidad de realizar nuevos modelos de aeronaves con facilidad.

Tal y como se ha explicado en la sección 5.2.4 inicialmente se desarrolló un simulador basado en JMonkey y posteriormente se decidió emplear el FlightGear [Flightgear, 2016]. Este simulador destaca, en primer lugar, por su modelo de dinámica de vuelo, el JSBSim [JSBSim, 2016]. Se trata de una aplicación de código abierto que modela la dinámica de vuelo de un vehículo aeroespacial con precisión. Además, dispone de la posibilidad de variar dinámicamente las condiciones atmosféricas y, al tratarse de software libre, existe una comunidad de desarrolladores y usuarios con abundante documentación lo que facilita y agiliza la realización de las tareas.

6.4.2. Modelado del UAV

El modelo debe representar lo más fielmente posible al avión real, para permitir que los resultados obtenidos en el simulador sean así lo más parecidos posibles a los que se obtienen en los vuelos reales.

Con esa finalidad se ha realizado una búsqueda de un modelo de FlightGear de características similares al de los UAV del equipo, para que sirviera

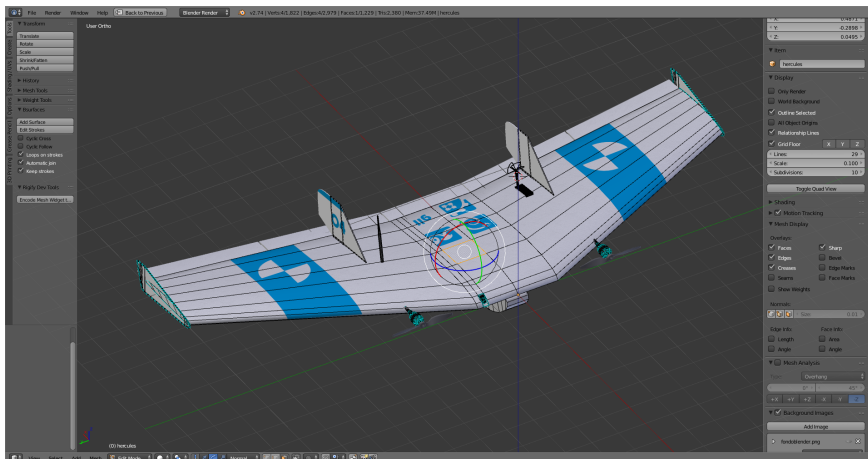


Figura 6.8: Captura del modelo gráfico del UAV en Blender [Blender, 2016]

de base y adaptarlo a nuestras características concretas. Finalmente, dado que la mayoría de modelos eran de aeronaves muy diferentes se optó por crear un modelo desde cero. Los modelos del FlightGear se componen en realidad de dos modelos claramente diferenciados. Por un lado está el modelo gráfico, que se usa en la visualización de las simulaciones y por el otro el modelo aerodinámico, que es el que emplea el JSBSim para los cálculos.

El modelo gráfico del avión consta a su vez de dos partes, una parte en la que se define la geometría en 3D del mismo y que se ha realizado usando el programa de modelado de código abierto Blender [Blender, 2016] y otra en la que se define la textura de las superficies mediante una imagen. En la Figura 6.8 se muestra el modelo gráfico del avión durante el proceso.

En lo referente al modelo aerodinámico, la parametrización para el JSBSim se realiza a partir de varios archivos xml, ya que JSBSim proporciona varios modelos físicos diferentes, que trabajan juntos para calcular la dinámica general de los vehículos. Hay que definir cuatro grupos de parámetros básicos para la aeronave:

- Los datos para el balance de masa: el simulador emplea la masa para proporcionar una fuerza debida a la gravedad y los momentos de inercia para amortiguar los cambios en la velocidad de rotación. Se especifican la masa en vacío, la inercia y masas puntuales (que puede tener una forma y tamaño).
- Los puntos de apoyo con el terreno para el aterrizaje y el despegue: cada punto proporciona una fuerza en el vehículo cuando está en contacto con el suelo. Si el contacto no se encuentra en el centro de gravedad también

proporciona un momento de rotación.

- La propulsión: existen diversos tipos de motores, todos ellos ofrecen una fuerza a lo largo de su eje X y crean momentos si esta fuerza no se aplica en el centro de gravedad. Los propulsores de hélice y rotor también crean momentos de rotación.
- Las características aerodinámicas: se permite definir las fórmulas que se consideren oportunas para calcular la fuerza aerodinámica, que se divide en sustentación, arrastre y fuerza lateral. También se pueden crear momentos de alabeo, cabeceo y guiñada.

Para la obtención de todos los datos necesarios se han realizado mediciones sobre el avión real tales como la envergadura, la cuerda alar, el diedro de las alas, etc. así como el peso en orden de vuelo, es decir, con las baterías, motores y sensorización colocados ya que al tratarse de un motor eléctrico no hay pesos variables como el del combustible.

En lo referente al motor y la hélice, se han introducido los datos en el programa JavaProp [JavaProp, 2016] para generar los coeficientes aerodinámicos para la hélice y se ha completado la información del motor con las especificaciones que se disponen.

Los coeficientes aerodinámicos del UAV, por su parte, constituyen la base para el correcto comportamiento del modelo. Algunos de los parámetros básicos se pueden obtener en base a datos del UAV y de aviones similares, mientras que otros más complejos requieren cálculos en programas en los que se ha de modelar el avión o ensayos en un túnel de viento. Dado que los ensayos suponían una inversión muy importante de tiempo y recursos, además de acceso a instalaciones complejas se optó por la primera opción. Así se realizó el modelado con el programa Athena Vortex Lattice [MIT, 2016] cuyos resultados se han ajustado bastante a la realidad, tal y como se puede comprobar en la sección 6.6 en la que se recogen y analizan los datos de las pruebas realizadas.

6.5. Estrategia de coordinación

En general, los algoritmos evolutivos suponen que se puede obtener información de cualquier punto del espacio de búsqueda en cualquier momento. Dado que en la realidad los UAV están sujetos a limitaciones en el movimiento se ha de buscar una forma de superar dicha limitación para aplicar este tipo de algoritmos.

En esta sección, en primer lugar se explica un método para superar dicha

limitación y posteriormente se describe en mayor profundidad el algoritmo evolutivo concreto empleado para este trabajo, basado en la evolución diferencial.

6.5.1. El muestreo restringido o Constrained Sampling

La estrategia evolutiva utiliza un enfoque híbrido donde la mayor parte del control está centralizado. El grado en que se puede distribuir la estrategia depende de los operadores particulares del algoritmo evolutivo específico utilizado en la implementación ya que algunos algoritmos, como CMA-ES, dependen del cálculo de una función de distribución que tenga en cuenta los valores de aptitud de todos los individuos de la población. Como se trata de un enfoque basado en la evolución es necesario definir un valor de aptitud para cada punto que se explora. En los problemas que implican agentes reales, este valor de aptitud viene determinado por el valor de un sensor o conjunto de sensores del agente.

En esta estrategia, el comportamiento del agente presenta dos fases de operación, tal y como se muestra en la Figura 6.9. Hay una etapa de exploración inicial que es necesaria para esparcir los agentes a través del ambiente antes de comenzar con la etapa evolutiva. De lo contrario, la búsqueda sería sesgada hacia las áreas que rodean los puntos iniciales. Durante esta etapa de exploración, los agentes envían la información detectada desde el entorno al controlador central, pero no reciben ninguna información del controlador y continúan con este comportamiento de exploración o dispersión hasta que se extienden por todo el entorno.

Cuando el controlador central comienza a recibir datos de detección, inicia el algoritmo evolutivo y emplea los datos recibidos por los agentes para evolucionarlo. Las entradas del algoritmo son las posiciones de los agentes y el valor de datos detectado por el sensor en esas posiciones. El algoritmo evolutivo almacena los datos de entrada en una cola hasta que tenga suficientes datos para ejecutar un paso evolutivo.

De esta manera, el número de agentes de búsqueda se desacopla de la población necesaria para ejecutar el algoritmo evolutivo. Una vez que el algoritmo tiene suficientes datos (es decir, una población suficientemente grande de valores detectados), genera una nueva población de posiciones que, según el proceso evolutivo, son más prometedoras que las actuales. Estos nuevos puntos son el resultado de aplicar los operadores evolutivos apropiados a la población de padres. Cada nueva población generada es almacenada temporalmente por el controlador, y se utiliza para seleccionar rutas prometedoras para los agentes.

Una vez que la etapa de exploración termina, cada agente comienza a tomar en cuenta los resultados del proceso evolutivo. Aquí el objetivo o la ruta de cada agente es definida por el controlador central.

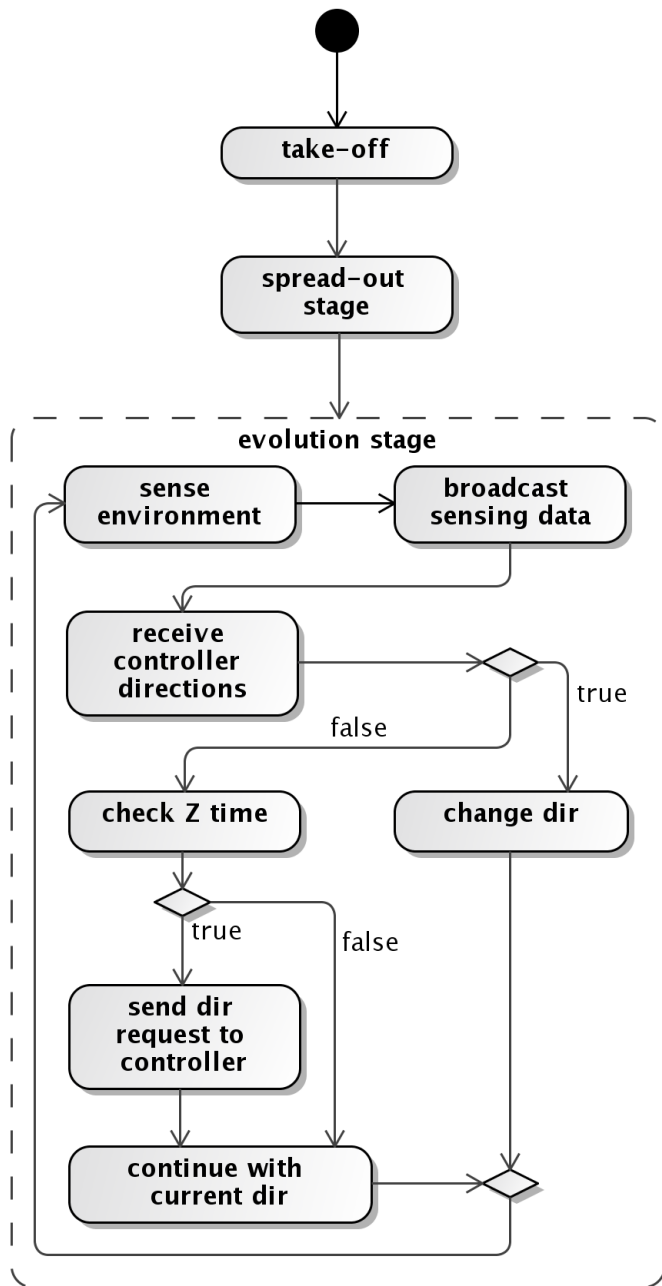


Figura 6.9: Diagrama de flujo del algoritmo evolutivo para el UAV

Durante la fase de evolución, cada Z segundos, cada agente envía un mensaje al controlador central solicitando un nuevo objetivo. Cuando el controlador recibe una solicitud, utiliza la población actualmente almacenada resultante del paso evolutivo anterior del algoritmo evolutivo para seleccionar una posición disponible para el agente solicitante. Luego lo elimina de las posiciones disponibles, para evitar que dos agentes reciban la misma ubicación de objetivo, aumentando así las capacidades de exploración del sistema, y envía la nueva posición objetivo al agente. El agente cambia su rumbo actual para tratar de alcanzar el objetivo recibido. Esta ruta se sigue hasta que han transcurrido los Z segundos y el proceso comienza de nuevo.

Como se indicó anteriormente, hay una etapa dentro del enfoque evolutivo en la que el controlador central obtiene una nueva población de posiciones prometedoras que debe explorarse. Sin embargo, la mayoría de los algoritmos evolutivos, cuando generan nuevas poblaciones, no toman en cuenta el hecho de que una entidad física debe preservar la continuidad en su trayectoria y tiene limitaciones y restricciones sobre el tipo de movimientos que puede lograr. Es decir, nuevas poblaciones de posiciones pueden implicar saltos discretos en el espacio o pueden requerir movimientos que no son factibles para el agente. Por otro lado, si los algoritmos están completamente limitados a producir sólo posiciones contiguas que respeten las restricciones de movimiento del agente, se pierde una gran parte de la capacidad de exploración.

Por ello, basado en el ciclo de ejecución común de un algoritmo evolutivo, se ha desarrollado una variante denominado Constrained Sampling - EA (CS-EA)[Varela et al., 2014a]. Este término tiene que ver con la imposibilidad de desplazar a los individuos que muestrean el espacio de búsqueda directamente a las posiciones generadas por el algoritmo evolutivo.

Para hacer frente a este problema se realizan varias modificaciones a un algoritmo evolutivo. En primer lugar, el algoritmo requiere dos poblaciones:

- La población evaluada, que representa la posición y los valores detectados en el tiempo de los agentes que evalúan el espacio de búsqueda durante su movimiento. Incluye un conjunto de posiciones puntuales y un valor para la aptitud (valor detectado) de estas posiciones.
- La población objetivo, generada por el algoritmo utilizando los operadores de reproducción y la información de aptitud proporcionada por la población evaluada. Corresponde a posiciones prometedoras del espacio de búsqueda.

Cada instante de tiempo, los agentes envían sus posiciones y los valores proporcionados por sus sensores al algoritmo evolutivo, estos valores son utilizados

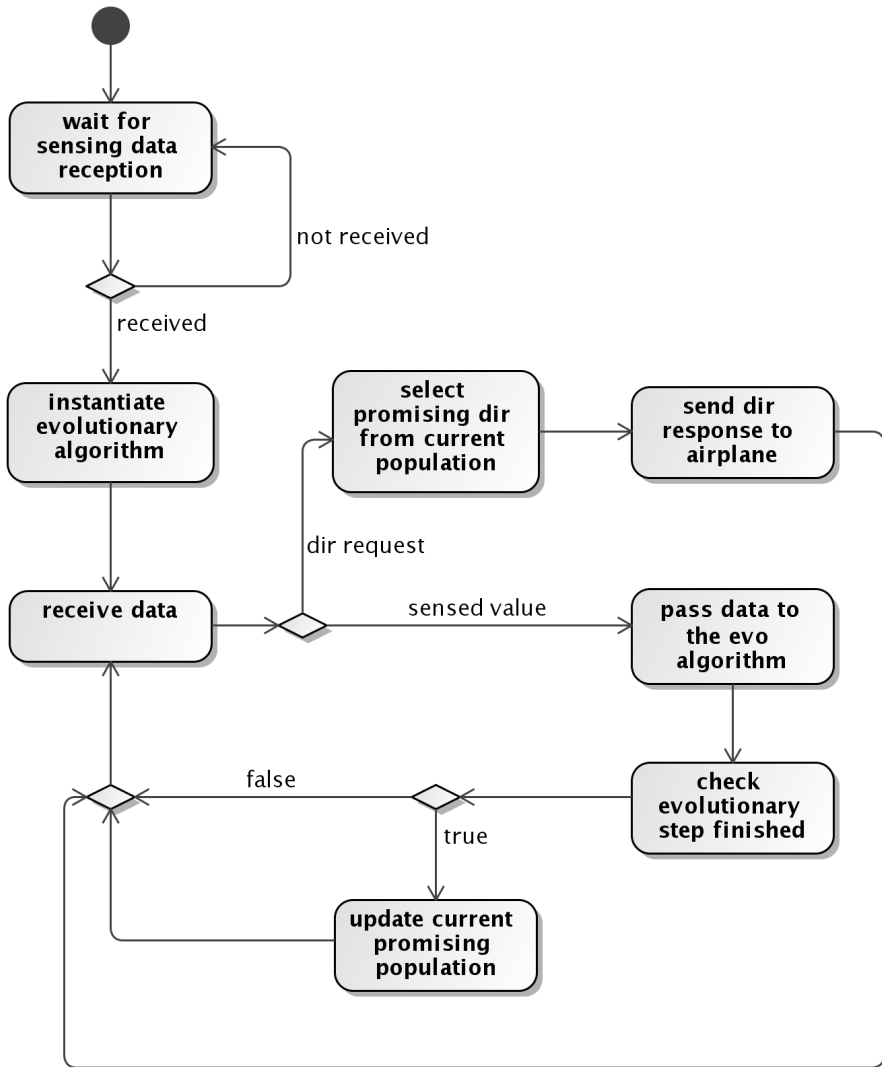


Figura 6.10: Diagrama de flujo del controlador central del algoritmo evolutivo

como los individuos evaluados. El algoritmo espera hasta que el tamaño de la población evaluada alcance el tamaño de población de CS-EA. En este punto, el CS-EA ejecuta una generación.

La población objetivo generada mediante los operadores CS-EA guía el movimiento de los agentes a lo largo del entorno hacia áreas que el algoritmo considera prometedoras, pero esta población no puede ser utilizada para guiar el proceso evolutivo debido a que, generalmente no hay un agente lo suficientemente cerca para tomar la medida en el tiempo requerido. Sin embargo, a medida que los agentes se mueven, están evaluando posiciones en el ambiente en la dirección de las posiciones objetivo y éstas pueden ser usadas como individuos evaluados en el CS-EA.

6.5.2. Constrained Sampling Differential Evolution (CS-DE)

En este trabajo, se utiliza un algoritmo de Evolución Diferencial (DE) como punto de partida debido a sus capacidades de exploración. El esquema básico del algoritmo DE itera sobre una población de soluciones candidatas aplicando una fórmula simple que combina información de individuos elegidos al azar de la población para generar una nueva población de soluciones potenciales. Cada solución candidata de la población es un vector n -dimensional, siendo n el número de parámetros de la función a optimizar.

En cada generación, durante la etapa de reproducción, el algoritmo DE aplica los operadores de mutación y cruce a cada individuo, también llamado vector objetivo, de la población. El resultado tras el proceso de mutación y cruce es otro vector denominado vector de prueba que se compara con el vector objetivo, conservando como miembro de la población de la siguiente generación el que tiene el mejor valor de aptitud.

Cuando existe la posibilidad de que el entorno tenga más de un óptimo se suele optar por el *niching*, que consiste en preservar la diversidad de la población a través de la formación de nichos o subpoblaciones que exploran diferentes áreas prometedoras del espacio de búsqueda. Esto permite que el algoritmo converja simultáneamente a varias soluciones de la función en la misma ejecución. Las técnicas de niching de EA que se pueden encontrar en la literatura incluyen *apiñamiento*, *intercambio de aptitud*, *limpieza*, *agrupamiento*, *paralelización* o *especialización*. En este trabajo se ha optado por aprovechar el comportamiento de agrupamiento de la estrategia de mutación aleatoria e incorporar el concepto de *vecindad* utilizando la información del individuo más cercano de la población para generar el individuo mutado.

De forma más específica, para cada individuo de la población $x_g^i; i =$

$1, 2, 3 \dots NP$, donde NP es el número total de individuos y g indica la generación, el vector de prueba se genera a partir del esquema DE/nrand/2/bin. En el cual $nrand$ es la estrategia de mutación, que genera el vector mutado (v_{g+1}^i) como:

$$v_{g+1}^i = x_g^{NNi} + F(x_g^{r_1} - x_g^{r_2}) + F(x_g^{r_3} - x_g^{r_4})$$

Donde:

- x_g^{NNi} es el individuo más cercano a x_g^i
- $r_1, r_2, r_3, r_4 \in \{1, \dots, NP\} / \{i\}$ son enteros aleatorios mutuamente diferentes, excluyendo el índice actual i
- F es un número real denominado factor de escala

El vector de prueba se obtiene aplicando el operador de cruce común binomial al vector mutado y objetivo. Para cada parámetro de los individuos j , donde $j = 1, 2, 3 \dots, n$, el vector de prueba (u_{g+1}^i) se genera como:

$$u_j^i = \begin{cases} v_j^i & \text{si } rand(0, 1) < CR \text{ o } j == k \\ x_j^i & \text{en cualquier otro caso} \end{cases}$$

Donde:

- CR es la tasa de cruce
- k es un entero entre $[0, n)$ lo que asegura que al menos se hereda un parámetro del vector mutado

6.6. Pruebas

En esta sección se presentan algunos de los resultados obtenidos durante las pruebas realizadas utilizando una mezcla de UAV reales coordinada con UAVs simulados con la misión de encontrar y caracterizar una fuente de contaminante atmosférico, que se ha expuesto en las secciones anteriores.

El objetivo principal de estas pruebas no es sólo mostrar cómo funciona el sistema propuesto, sino sobre todo utilizarlo para demostrar el potencial de la aproximación propuesta para reducir y facilitar el paso del *reality gap*. Con el fin de minimizar el reality gap, el objetivo es replicar en el mundo virtual todas

las características relevantes del mundo real que tengan una influencia efectiva, por un lado, en el comportamiento de los UAV, sus controles y su coordinación y; por otro lado, sobre el modelo del mundo implementado.

La fuente de datos para estas pruebas es la chimenea de la central térmica de As Pontes y las condiciones reales que caracterizan la prueba son las que se han utilizado en la sección 6.2 para configurar el modelo de la pluma.

En un primer paso se busca ajustar los controles de vuelo y los sensores de los UAV reales y simulados. El ajuste y configuración de los controles de vuelo de los UAV empleando la realidad mixta no es nada novedoso, pero es un paso preliminar necesario en el proceso. La sección 6.6.1 presenta con detalle cómo se han ajustado los controles de vuelo y la sección 6.6.2 muestra el ajuste del modelo de dispersión de contaminantes empleado.

El segundo paso es ajustar los parámetros para el entorno y el modelo de misión que se implementa en la plataforma. En este caso, el entorno corresponde a un vuelo al aire libre para el que la velocidad y dirección del viento es muy importante. Además, los valores iniciales de los parámetros del modelo de pluma tienen algunas incertidumbres. Así, se han realizado algunos vuelos específicos para ajustar tanto los parámetros del modelo meteorológico como del modelo de pluma. Estas pruebas sirven además para validar el modelo de pluma utilizado. Es decir, este modelo ha sido elegido por su simplicidad, pero podría resultar no suficientemente preciso para lo que se pretende.

El último paso se refiere al ajuste del algoritmo de coordinación que controla el equipo de UAV. Como en el caso anterior, este algoritmo necesita ser ajustado y validado. En este caso se emplea un ajuste incremental del número de UAV reales en la plataforma como un medio de reducir progresivamente el reality gap.

6.6.1. Vuelos para el ajuste del controlador

El primer uso de la plataforma es el ajuste de los controladores de vuelo de los UAV. Este uso es habitual en la mayoría de las aplicaciones de UAV que hacen uso de simuladores y resulta necesario en este caso para permitir que los UAV vuelen de forma autónoma. Los controladores son idénticos para los UAV reales o simulados, por lo que para que funcionen correctamente ambos deben comportarse de la forma más parecida posible.

Los controladores de ambos bucles son controladores PID típicos, por lo que los términos proporcionales, integrales y derivados de cada uno deben ajustarse al comportamiento dinámico de los UAV. Para afinar estos parámetros se adoptó un enfoque incremental. En primer lugar se utilizó la plataforma con un

único UAV simulado en el entorno virtual de FlightGear y Jsbsim para simular la dinámica del avión en diferentes condiciones de vuelo y meteorológicas. Un programa basado en Java permite implementar los controladores y ajustar sus parámetros. Otro programa, basado en el mismo lenguaje de programación, permite generar gráficas con los datos de vuelo UAV. Estos datos, una vez analizados y trazados, se han utilizado en el proceso de ajuste.

Los controladores han sido sintonizados en un orden específico: cabeceo, velocidad, energía, alabeo y rumbo. Los controladores de bucle interno en el entorno de UAV real se han implementado con la ayuda de una IMU, que presenta valores de ruido e imprecisiones mayores que las del simulador. Esto provocó la necesidad de realizar el ajuste fino de los controladores reales, aunque los valores ya habían sido extraídos del simulador. Las pruebas reales se llevaron a cabo bajo la supervisión de un piloto humano y con un modo de control del UAV específicamente diseñado.

Utilizando un programa desarrollado especialmente para este proceso, un asistente humano puede ajustar la ganancia general y cada una de las ganancias de todos los controladores PID durante el vuelo. Los parámetros se establecieron inicialmente después de los obtenidos en el proceso de ajuste del entorno virtual, y luego se hicieron algunos vuelos específicos para afinar estos parámetros en el entorno real. Primero los controladores de bucle interno y luego los de bucle externo. La Figura 6.11 muestra parte de la trayectoria de vuelo seguida en uno de estos vuelos. Como puede verse consiste principalmente en tramos rectos en los que se va ajustando la ganancia de los PID.

Los parámetros obtenidos en estas pruebas de entorno real se utilizaron tanto en controladores de los UAV reales como simulados, haciendo así que el comportamiento de los UAV real y simulado fuese lo más cercano posible, y ayudando a superar el reality gap. Como se puede ver en la Figura 6.11, también se han obtenido medidas de CO₂ en estas pruebas de vuelo. Estos datos se usaron para calibrar y asegurar el correcto funcionamiento de los sensores de CO₂, así como para establecer niveles básicos de CO₂ presentes en el ambiente (color azul en la trayectoria de vuelo).

6.6.2. Vuelos para el ajuste del entorno virtual

Los datos del entorno se obtienen de los UAV reales, se transmiten a la plataforma y se utilizan para ajustar el comportamiento del entorno virtual, aproximándolo al real y disminuyendo el reality gap.

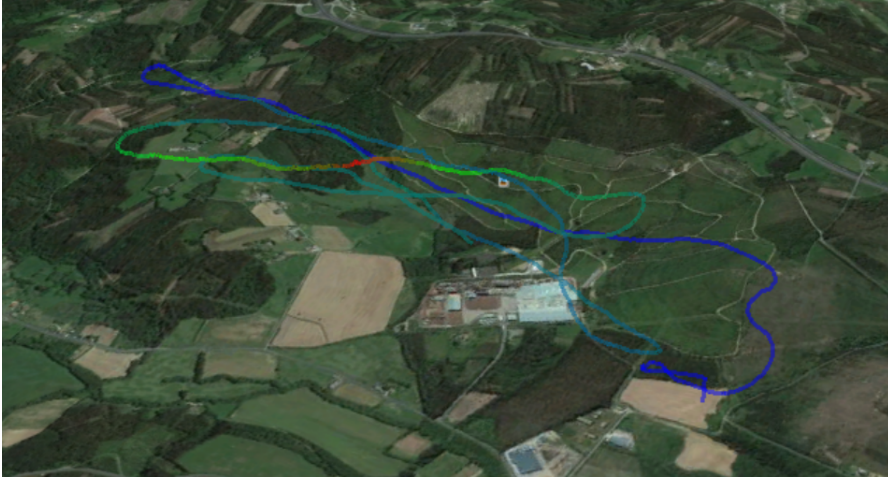


Figura 6.11: Trayectoria seguida por el UAV durante uno de los vuelos de ajuste del controlador

6.6.2.1. Ajuste del viento real

De forma más específica, las velocidades del GPS y del pitot se utilizan para estimar datos reales del viento en tiempo real, y georeferenciarlo. Esta información se utiliza para afinar el modelo meteorológico del mundo virtual, haciendo así que los UAV simulados se aproximen más a los reales.

El estimador es un algoritmo simple que calcula el rumbo bidimensional de la velocidad del viento y la magnitud de las velocidades GPS y pitot. Estos datos entrantes se acondicionan previamente con un filtro de paso bajo de un solo polo con una frecuencia de corte de 0,33 Hz, y luego se aplican relaciones trigonométricas simples para estimar la velocidad del viento entre dos pasos consecutivos de tiempo (k y $k - 1$):

$$\beta_k = \arccos \frac{v_p^2 - v_{pk}^2 - \|\Delta v_{ak}\|^2}{2v_p \|\Delta v_{ak}\|}$$

Donde v_p es el módulo de velocidad relativa del viento medido por Pitot y v_{ak} el vector de velocidad del plano absoluto (GPS). El ángulo calculado β_k es el que está entre la velocidad relativa del viento v_{rk} y $\Delta v_{ak} = v_{ak} - v_{a(k-1)}$ en el paso del tiempo k . A partir de este se calcula el ángulo de v_{rk} , α_k . A partir de aquí, la velocidad relativa del viento v_{rk} , se forma a partir de v_{pk} : $v_{rk} = v_{pk} \angle \alpha_k$. Por último, la velocidad del viento se obtiene componiendo velocidad relativa y absoluta:

$$v_{wk} = v_{ak} - v_{rk}$$

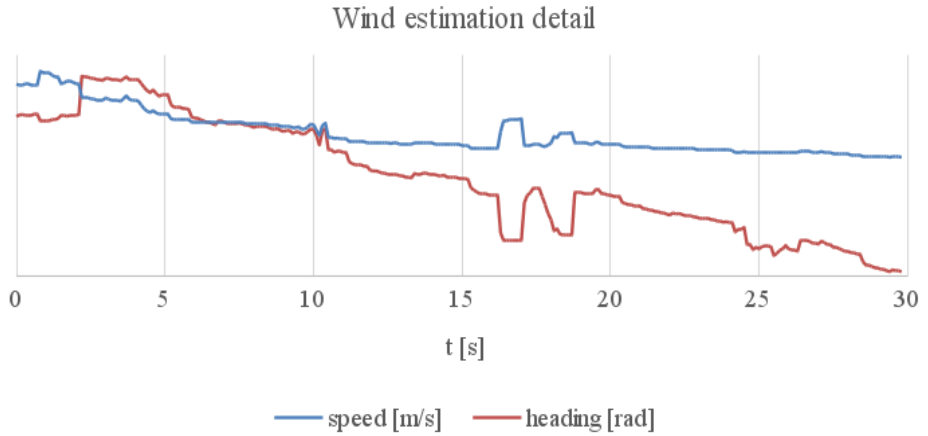


Figura 6.12: Estimación de la velocidad del viento

En la Figura 6.12 se muestra la estimación de la velocidad del viento con los valores de rumbo y magnitud obtenidos a partir de los datos de vuelo real utilizando el estimador simple que se ha expuesto.

6.6.2.2. Ajuste del modelo de la pluma

Un requisito para que cualquier modelo de dispersión de contaminación sea usado dentro de la plataforma es que pueda ser parametrizado de tal manera que se pueda implementar fácilmente. Adicionalmente, para adaptar el modelo a un caso de prueba dado, los valores de los parámetros de definición deben ser fáciles de modificar. En muchos casos, estos requisitos podrían conducir a modelos que van a depender de un conjunto limitado de variables. Además, estos modelos podrían presentar una alta sensibilidad a algunos de estos parámetros. En consecuencia, un correcto ajuste de los mismos es un problema crítico cuando se trata de obtener buenos resultados de estos modelos de dispersión de contaminación y, por tanto, del propio simulador. Además, la estimación de algunos de los parámetros que gobiernan es a veces inexacta.

Analizando el modelo de pluma gaussiano utilizado en el presente trabajo, se puede observar que la dirección y velocidad del viento son parámetros que pueden obtenerse con una buena exactitud utilizando datos meteorológicos en tiempo real o mediciones de campo. Sin embargo, la altura efectiva de la chimenea, el valor máximo y las desviaciones estándar de la concentración de contaminantes se evalúan dentro del modelo usando la correlación empírica proporcionada por Briggs, que se determinan usando la clasificación Pasquill de estabilidad atmosférica y por consiguiente podrían ser inexactas.

Además, la estimación de la emisión de contaminantes se realiza después de asumir que la central térmica está trabajando en condiciones estándar, lo que puede no ser el caso. Sin embargo, el valor máximo de concentración de contaminante y la altura en la que se produce son parámetros que podrían obtenerse sin demasiado esfuerzo a través de mediciones de los UAV. Por lo tanto, la idea es corregir estos valores para adaptar el modelo de dispersión de contaminantes mientras se ejecuta la prueba. De esta manera, la mayoría de las incertidumbres que surgen para la estimación del modelo que rige los parámetros se ve disminuida.

Como se ha mencionado anteriormente, el número de grupos generadores de la central térmica que se encuentran en funcionamiento se puede comprobar visualmente. Además, la velocidad y la dirección del viento son parámetros que pueden medirse en el campo u obtenerse de páginas meteorológicas que proporcionan información en tiempo real. Asimismo, las correlaciones para estimar la velocidad del viento a diferentes alturas son relativamente buenas, principalmente cuando se trabaja en pequeñas variaciones de altura, como es este caso.

También las variaciones en la velocidad y dirección del viento son relativamente pequeñas en distancias horizontales de hasta 10 km que caracterizan estas pruebas. Por lo tanto, se puede iniciar el modelo de dispersión de contaminantes basándose en la estimación de todos los parámetros de gobierno. Así, los vuelos de UAV reales proporcionan información sobre la altura a la cual se obtiene la concentración promedio máxima de contaminante. Por lo tanto, esta altitud es la altura efectiva de la chimenea y la concentración medida corresponde a la línea central de la pluma y por lo tanto es el máximo de la distribución gaussiana. Particularizando la expresión de la distribución de la concentración de contaminantes c a esta línea central:

$$c_{max}(x) = \frac{Q}{2\pi u \sigma_y \sigma_z}$$

A partir de esta expresión, las emisiones de caudal de contaminante Q pueden obtenerse después de dar los valores de las desviaciones estándar gaussianas y y z . Estos dos valores se pueden estimar utilizando los parámetros de dispersión de Briggs que se dan en la Tabla 6.3. En los casos en que varios UAV reales están volando, los valores de las desviaciones estándar pueden obtenerse estadísticamente a partir de los datos de concentración de contaminantes medidos por los UAV. En consecuencia, las mediciones reales de campo de UAV obtenidas durante el ensayo se utilizan para mejorar el modelo de dispersión aumentando la precisión de los valores inicialmente estimados para la altura efectiva de la chimenea y la emisión del flujo másico del contaminante.

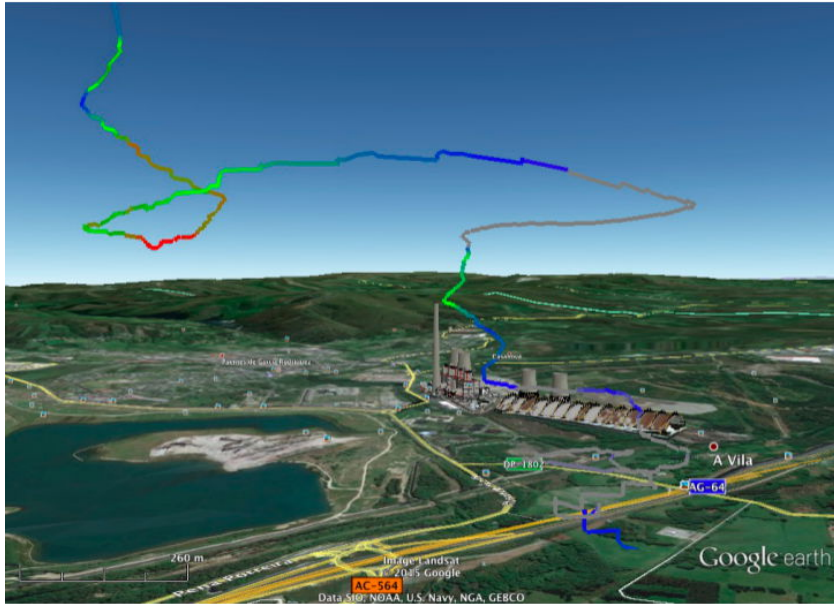


Figura 6.13: Concentraciones de CO₂ medidas durante un vuelo representado con KML de Google Earth

Introduciendo estos valores en el modelo de pluma, se proporciona una mejor representación de la pluma dentro del simulador. Se han realizado algunas pruebas en las que se ha coordinado un único UAV real con un equipo de UAVs simulados. La misión ha sido la de encontrar y caracterizar una fuente de contaminante atmosférico a partir de la salida de la chimenea de la central térmica de As Pontes. Siguiendo la estrategia planteada a lo largo de este trabajo, la información obtenida del UAV real se ha utilizado para ajustar los valores de los parámetros que rigen el modelo de dispersión de contaminantes de plumas gaussianos.

Para ilustrar la importancia de esto, la Figura 6.13 presenta uno de los vuelos de prueba reales de UAV. La figura representa por medio de un código de colores los valores de la concentración de CO₂ medida por el sensor de UAV en la trayectoria de vuelo. Estos valores de concentración se han registrado junto con los datos GPS en un archivo KML y luego se han representado en Google Earth. Los colores codificados van del gris al rojo. El gris corresponde a una concentración de 450 ppm o menos y el rojo corresponde a 840 ppm o más. La concentración máxima de CO₂ medida fue de 859 ppm a una distancia de 2,4 Km de la chimenea y el nivel de concentración de base medida en el aire libre fue de 449 ppm.

Uno de los problemas observados durante estos vuelos fue que la altura

efectiva de la pila era mucho menor de lo previsto. De hecho, en lugar de ser de 634 m era sólo 486 m. También el valor máximo de la concentración de contaminantes fue 3,9% mayor que el valor estimado. Esta última estimación se ha realizado utilizando la expresión anterior para la línea central y la tabla 6.3 y teniendo en cuenta el nivel de concentración de contaminantes base.

Las discrepancias entre las mediciones y las estimaciones surgen por una parte de la simplicidad del modelo paramétrico de pluma utilizado y por la falta de datos precisos disponibles para estimar los parámetros gobernantes del modelo. Los valores de medición correctos de estos parámetros se han introducido en el modelo. Después de esta corrección, se continúa con las pruebas de la estrategia de búsqueda del equipo coordinado de UAV.

6.6.3. Vuelos realidad mixta en As Pontes

Cuando se utiliza un algoritmo de control para el equipo, aparece uno de los problemas identificados al principio del desarrollo para cruzar el reality gap. Es decir, la intención es adaptar el entorno en línea y continuamente con información real. Así, inicialmente se dispone de un modelo de mundo estimado, y se actualiza con los valores real provenientes de los UAV.

Desde el punto de vista del controlador del equipo, es decir que las características del espacio de búsqueda se modifican continuamente durante el proceso de búsqueda o caracterización. Esta es una de las razones que apoyan la elección de un enfoque basado en algoritmos evolutivos, ya que este tipo de algoritmos han mostrado un gran rendimiento en la resolución de problemas en los que los espacios de búsqueda se modifican dinámicamente. Como se muestra en [Varela et al., 2014a].

Los algoritmos evolutivos generan, paso a paso, nuevas poblaciones de puntos prometedores donde se lee el valor máximo de CO_2 . Cada UAV utiliza una estrategia para seleccionar uno de esos puntos, e intenta alcanzar ese punto hasta que un nuevo paso del algoritmo evolutivo genere un nuevo conjunto de puntos prometedores. Los aviones están alimentando continuamente al algoritmo evolutivo con nuevos datos del campo, lo que utiliza el algoritmo evolutivo en cada paso para generar una nueva población de puntos hijos más prometedores que la anterior.

La prueba realizada dentro de este trabajo sostiene el enfoque de este trabajo para tratar con el problema del reality gap. Esta prueba se ha realizado utilizando cinco UAVs simulados y un UAV real. En cada instantánea de la Figura 6.14, los UAV están representados por flechas; Los negros son para los UAV simulados y los rojos para los UAVs reales. La trayectoria seguida por cada uno también está representada usando líneas coloreadas. Durante su vuelo, am-

Los tipos de vehículos aéreos no tripulados están tomando datos del ambiente, esta información se utiliza para alimentar el CS-DE. El conjunto de puntos prometedores propuesto por el algoritmo evolutivo está representado por estrellas amarillas.

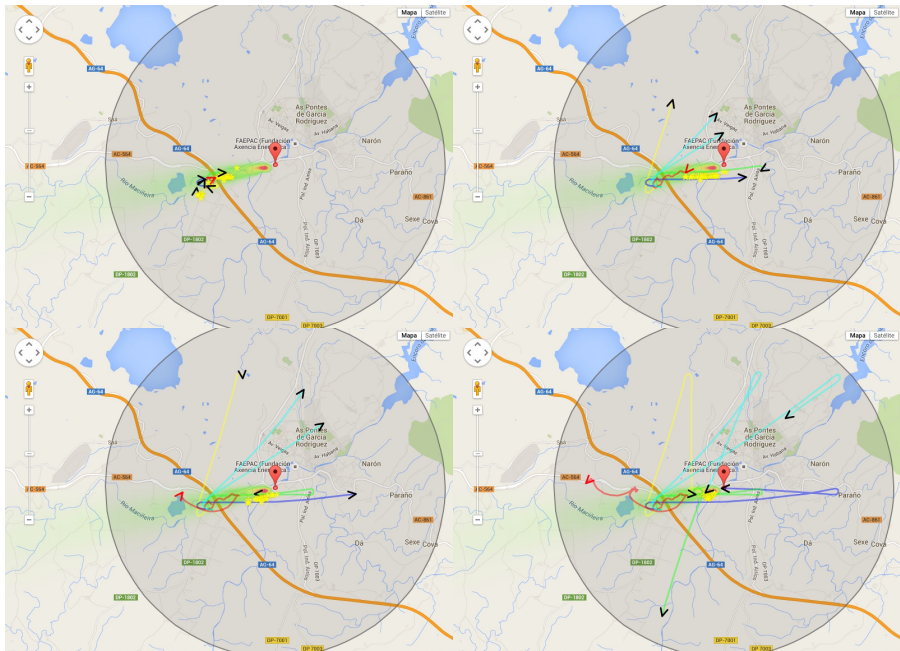


Figura 6.14: Secuencia de instantáneas del API de Google Maps obtenidas utilizando un algoritmo CS-DE para el controlador del equipo.

Como se muestra en la secuencia de instantáneas, a medida que progresa la ejecución y se dispone de más información del entorno, la población del algoritmo evolutivo se acerca al punto de concentración máxima de la pluma. Debe tenerse en cuenta que el UAV real está restringido por las cuestiones de seguridad y así como la legislación vigente en el momento del vuelo.

6.6.3.1. Finalización prematura de las pruebas por el cambio de legislación

Debido al creciente número de UAV y el miedo a un mal uso de los mismos, en el año 2014 se produjo la entrada en vigor de una nueva legislación en lo que se refiere al uso de UAV en España. Este cambio, producido durante el transcurso de este trabajo, supone la imposición de unas normas mucho más restrictivas que las que había hasta ese momento y no permite continuar con la realización de vuelos en los alrededores de la chimenea de As Pontes.

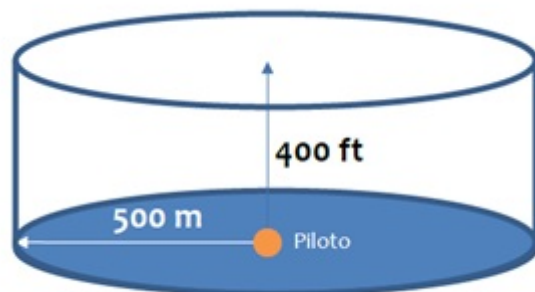


Figura 6.15: Zona de vuelo permitida actualmente para los UAV

Actualmente únicamente se puede volar hasta 400 pies (aproximadamente 120 metros) de altura desde la zona de despegue y se debe mantener en todo momento el UAV dentro del alcance visual del piloto, que puede alcanzar como máximo los 500 metros.

La figura 6.15 muestra el cilindro que constituye el espacio de vuelo permitido actualmente para los UAV. Tal y como se recoge en el presente documento, las alturas de la pluma son mucho mayores de 120m y las distancias requeridas para alcanzarla están fuera del rango determinado para el alcance visual. Así, no se pueden realizar más vuelos con UAV reales alrededor de la chimenea de As Pontes, aunque se espera que la nueva legislación añada excepciones como los vuelos para la investigación o establezca límites menos restrictivos.

6.6.4. Reajuste de las pruebas para la adaptación a la nueva legislación

Para hacer frente a este problema se ha tenido que adaptar la misión a las limitaciones existentes con la finalidad de ser capaces de demostrar el potencial de la aproximación, mientras se realizan las pruebas con los UAV reales dentro de los límites legales. En primer lugar, en vez de tratar de buscar una fuente de polución cuya altura efectiva se encuentre dentro de los 120 metros de altura se ha optado por hacer uso de la plataforma de realidad mixta para introducir sensores simulados en los UAV. En los vuelos de prueba anteriores se ha mejorado el modelo de chimenea paramétrico con datos medidos por los UAV reales, a partir de ahora, este modelo es el que se emplee para todos los UAV virtuales, sean estos reales o simulados. En vez de contar con un sensor, los UAV reales únicamente han de leer los datos correspondientes del modelo para la posición en la que se encuentran. De esta forma se cuenta con una chimenea virtual, que

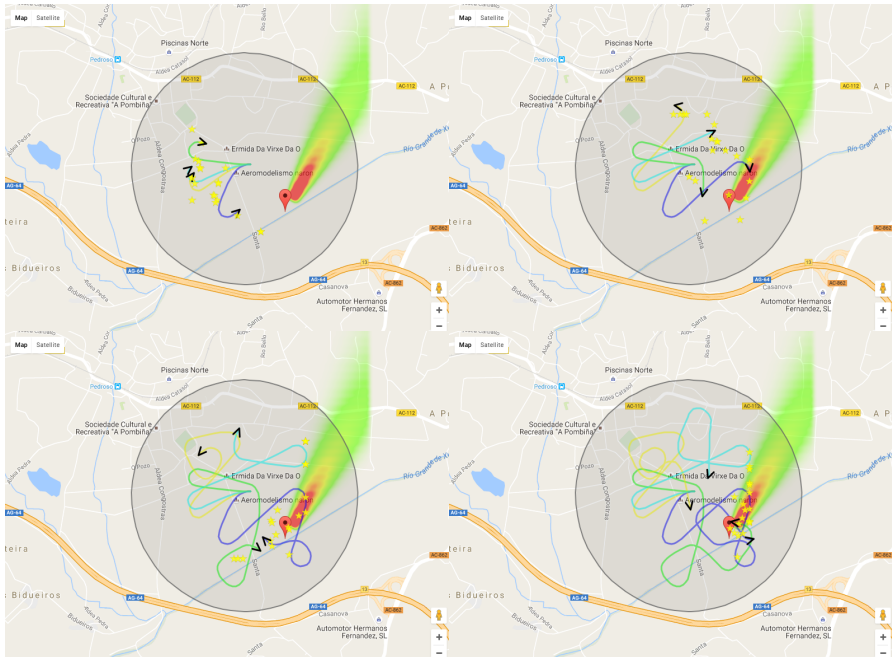


Figura 6.16: Secuencia de instantáneas con 4 UAV simulados tras el reajuste de la misión

se puede situar a cualquier altura y en cualquier lugar que se desee.

La segunda modificación que se ha realizado tiene que ver con el número de UAV y el espacio de búsqueda. Inicialmente se realizaron las pruebas virtuales con 6 UAV y el espacio de búsqueda estaba delimitado por un círculo de varios kilómetros. Si embargo la legislación actual obliga a reducir este círculo a un máximo de 500 metros. Dado que la extensión de la pluma de la chimenea de As Pontes ocuparía prácticamente todo el espacio en una misión como esta, se ha modificado la chimenea virtual, rebajando la cantidad de contaminantes emitidos para reducir la pluma. Además, se ha reducido el número de UAV del equipo de 6 a 4. De esta forma, empleando el modelo de pluma de la chimenea de As Pontes, validado con datos reales y modificándolo para reducir su área de influencia se puede reducir también el área de trabajo para todos los UAV y fijarla en un círculo de 500 metros de radio, tal y como indica la legislación vigente.

Una vez realizados estos cambios resulta sencillo desplazar el área de trabajo a una zona de fácil acceso y en la que se pueda volar los UAV con seguridad y dentro de la legislación. Se ha optado por realizar estas pruebas sobre el campo de vuelo de aeromodelismo de Pedroso, en el municipio de Narón. Se han realizado pruebas para comprobar y afinar el algoritmo con los valores reajustados

para el área de trabajo, la chimenea y el número de UAV que, dada la reducción, del espacio de búsqueda se ha fijado en 4. En la Figura 6.16 se puede ver una secuencia de imágenes del proceso de simulación y ajuste del algoritmo.

6.6.4.1. Vuelos de realidad mixta en Pedroso

Una vez comprobado el correcto funcionamiento del algoritmo para las nuevas condiciones de la misión, se procede a realizar las pruebas del entorno de realidad mixta con un UAV real volando en el campo de aerodelismo de Pedroso y 3 UAV simulados. El UAV real vuela midiendo los datos suministrados por el sensor virtual, que registra los valores de la chimenea virtual.

La comunicación entre las redes del UAV real y los virtuales se produce a través de la estación en tierra, que se encuentra conectada por USB al ordenador donde se ejecuta la simulación. Una pasarela programada en Java se encarga de traducir los paquetes que llegan del USB a paquetes UDP para conectarlo con la plataforma de realidad mixta. Al mismo tiempo, el algoritmo envía sus consignas a través de UDP y la pasarela se encarga de que lleguen a la estación en tierra y se transmitan al UAV real de forma inalámbrica.



Figura 6.17: Secuencia de instantáneas con 1 UAV real y 3 simulados

De forma análoga a lo que sucedía en la sección 6.6.3 los UAV están representados por flechas. Las negras son para los UAV simulados y los rojos para

los UAV reales. La trayectoria seguida por cada uno también está representada usando líneas coloreadas. Durante su vuelo, ambos tipos de vehículos aéreos no tripulados están tomando datos del ambiente, esta información se utiliza para alimentar el CS-DE. El conjunto de puntos prometedores propuesto por el algoritmo evolutivo está representado por estrellas amarillas.

Como puede verse los UAV se dispersan y, tras un pequeño tiempo de exploración se centran en la zona de la pluma. Cabe destacar que aunque se trate de una visualización en dos dimensiones de las trayectorias el modelo de la pluma es tridimensional. Así, por ejemplo, en una de las imágenes parece que el UAV real atraviesa la zona de mayor concentración de contaminantes, pero lo hace a una altura diferente a la de la pluma, por lo que no obtiene una lectura demasiado alta. Hacia el final de la misión se puede apreciar como todas las estrellas que representan zonas prometedoras se centran en la zona de la pluma.

6.6.4.2. Incremento progresivo del número de UAV reales

Siguiendo la metodología propuesta a lo largo de todo este trabajo se continúa con el proceso de incrementar el número de UAV reales progresivamente, de forma que en esta segunda etapa se añade un nuevo UAV real. Así, se dispone de dos UAV reales y dos UAV simulados en el equipo.

Puede apreciarse que en cada uno de los experimentos la dirección de la pluma de la chimenea es diferente, ya que el modelo se adapta a la dirección e intensidad del viento presente en la zona en el momento de realizar los vuelos.

El proceso se muestra en la Figura 6.18 en la que se pueden ver los UAV reales como flechas rojas y los UAV simulados como flechas negras. Se aprecia como las trayectorias de los UAV reales presentan mayores curvas que los virtuales. Esto puede ser debido a las ráfagas de viento, las pequeñas discrepancias entre el modelo del simulador y el UAV real o incluso al ruido o vibraciones existentes en el UAV real y que el simulador no tiene en cuenta. Dichas diferencias quedan muy lejos, sin embargo, de afectar al funcionamiento del algoritmo, que localiza en poco tiempo la fuente de polución virtual.

6.7. Conclusiones

Este capítulo presenta un estudio inicial sobre la viabilidad de llevar a la realidad las estrategias de coordinación para que múltiples UAV colaboren para la consecución de una meta común. La meta seleccionada en este caso es la localización y caracterización de una fuente de polución. En una etapa inicial se

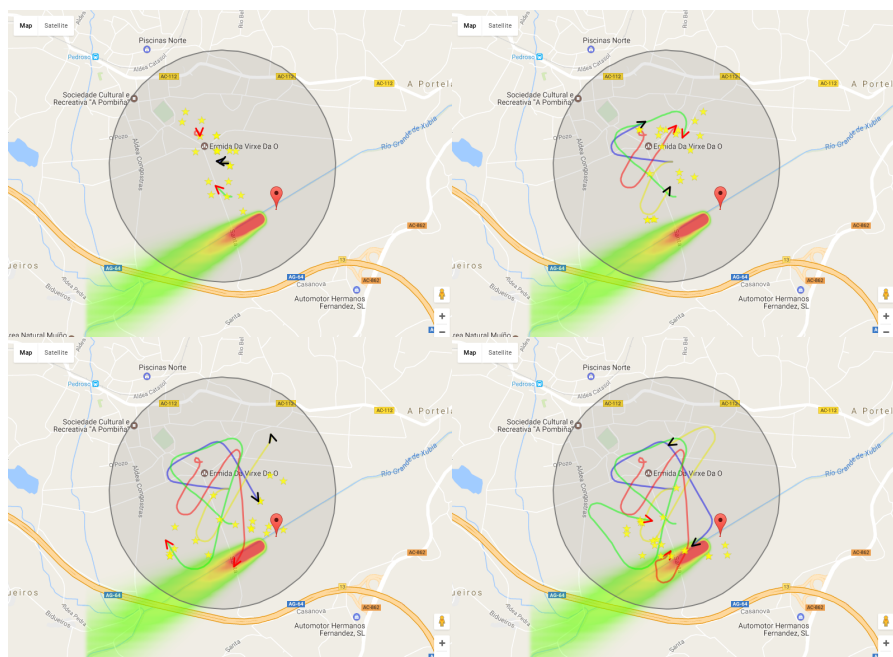


Figura 6.18: Secuencia de instantáneas con 2 UAV reales y 2 simulados

ha tomado como fuente de polución la central térmica de carbón de As Pontes.

Para alcanzar este objetivo, en primer lugar se ha realizado un modelo paramétrico de la chimenea con los datos que se dispone del proceso y se han aproximado los valores desconocidos de forma razonada. A continuación se han realizado una serie de vuelos para el ajuste de los controladores encargados de permitir el vuelo autónomo del UAV, que se habían ensayado previamente sobre el simulador. Estos vuelos también se han empleado para la comprobación de funcionamiento del sensor de CO_2 y para la calibración del modelo de la chimenea virtual.

Una vez se dispone de un modelo calibrado con los datos reales, se ha empleado el simulador para probar una estrategia de coordinación que encuentre la fuente de polución en un espacio de búsqueda amplio y sin información sobre el entorno salvo la proporcionada por el sensor de CO_2 . Una vez obtenidos resultados satisfactorios con los UAV simulados, se ha empleado la plataforma de realidad mixta para comenzar a trasladar la estrategia de coordinación desarrollada sobre el simulador a la realidad. La plataforma facilita este proceso al emplear el mismo control para ambos UAV. El resultado fue un vuelo satisfactorio con un UAV real y cinco UAV simulados, en los que se puede apreciar el correcto funcionamiento de la plataforma.

Sin embargo, antes de poder incorporar más UAV reales al equipo este camino se ve truncado por un cambio de legislación, que no permite realizar los ensayos previstos y obliga a replantearse la aplicación para poder finalizar la demostración del potencial de la plataforma y de la incorporación progresiva de la complejidad real.

Para hacer frente al contratiempo, se reduce el área de búsqueda, el número de UAV y se modifica la chimenea virtual, ajustada con datos reales, a las limitaciones existentes para el vuelo de UAV reales. Se comienza de nuevo el proceso para esta aplicación reajustada en la que el objetivo se mantiene inalterado: localizar y caracterizar la fuente de polución. Se reajusta la estrategia de coordinación para que funcione en este nuevo escenario y, una vez conseguido se realiza de nuevo el proceso con un UAV real y el resto simulados.

Las pruebas con un UAV real y tres simulados son exitosas, por lo que se continúa y se incorporan dos UAV reales, que ejecutan la misión junto con dos UAV simulados. De esta forma se demuestra que se puede emplear la plataforma para cruzar el reality gap de forma progresiva y primando la seguridad. Así, durante el proceso se parte de modelos sencillos que se mejoran con la información real de forma iterativa hasta obtener comportamientos razonablemente similares a los reales. Sobre estos modelos se desarrollan los controladores que también son probados en la realidad de forma progresiva.

Capítulo 7

Conclusiones y trabajo futuro

La meta principal de esta tesis doctoral es proponer una aproximación que facilite el desarrollo e implementación de las estrategias de coordinación de múltiples UAV para la realización de tareas colaborativas. Esta aproximación consiste en una metodología que, unido a una plataforma de realidad mixta, permite agilizar, facilitar y hacer más seguro el proceso de desarrollo e implementación de nuevas estrategias para lograr que varios UAV colaboren para una tarea común. En este capítulo se presenta de manera resumida las principales conclusiones y aportaciones del trabajo realizado.

Las aportaciones en lo que se refiere a la metodología y los principios de diseño han sido:

- La generación de una estructura normalizada y modular para el software de los UAV, definida desde un punto de vista funcional e independiente del tipo y tamaño del UAV.
- La definición de un middleware entre el firmware y el software de alto nivel de los UAV para conseguir la independencia del software del hardware concreto que ofrece la funcionalidad.
- La conservación del código de la estrategia de coordinación del simulador a la realidad mediante la implementación de la estrategia de coordinación en un lenguaje de programación compatible y la inclusión del hardware necesario a bordo del UAV.

En lo que se refiere al aumento de la seguridad durante el desarrollo e implementación de las estrategias de coordinación se ha trabajado para integrarlo tanto en el software como en el hardware, especialmente en lo que se refiere a la

superación del reality gap. Las principales aportaciones de este trabajo en este ámbito han sido:

- Tener en cuenta la seguridad en todo el proceso, tanto en el software como en el hardware real, anteponiendo estas consideraciones a otras como la mejora de la logística para los vuelos.
- Desarrollar herramientas de visualización de datos en tiempo real, como la representación de las trayectorias de todos los UAV del equipo, que facilitan la lectura e interpretación por parte del operador de la información recibida de los UAV.
- Diseñar y generar un entorno de simulación de realidad mixta que permita el uso indistinto de UAV reales y simulados: mediante el uso de la plataforma propuesta se puede superar el reality gap para la estrategia de coordinación de forma incremental, aumentando progresivamente el número de UAV reales en vez de realizar el cambio de la simulación a la realidad en una única etapa. El trato homogéneo a ambos tipos de UAV facilita el empleo de esta forma, manteniendo el mismo código para la estrategia de coordinación sean los UAV simulados o reales.

Finalmente, para demostrar la validez de la aproximación propuesta se ha diseñado una aplicación consistente en la localización y caracterización de una fuente de polución atmosférica por parte de un equipo de UAV coordinados mediante un algoritmo evolutivo. Se han realizado simulaciones y se ha empleado la plataforma de realidad mixta para incrementar progresivamente el número de UAV reales en el equipo, implementando así la estrategia de coordinación en la realidad. Las principales aportaciones en este caso son:

- Un modelo de dispersión de contaminantes gaussiano para la pluma de la chimenea parametrizable en tiempo real que se ha ajustado con valores reales.
- Una plataforma de UAV reales con capacidad de vuelo autónomo: se ha realizado el diseño y desarrollo de los mismos, instrumentándolos con una combinación de hardware y software específico para permitir la realización de vuelos autónomos, la comunicación inalámbrica y se ha incluido la sensorización adecuada para la realización de pruebas reales.
- Un modelo de UAV de comportamiento similar al real, que permite realizar pruebas sobre el simulador de las estrategias de coordinación previamente a su implementación en la realidad.
- Una adaptación para emplear algoritmos evolutivos en la coordinación de UAV. Para ello se ha ajustado el mismo manteniendo el funcionamiento

básico y teniendo en cuenta las limitaciones para la adquisición de datos de los UAV. Este algoritmo, denominado CS-DE, se ha empleado para la localización de la fuente de polución.

La plataforma de la realidad mixta ha proporcionado además de la posibilidad de incorporar la estrategia de coordinación de forma iterativa otras ventajas y funcionalidades no contempladas inicialmente en los objetivos de este trabajo:

- La posibilidad de incorporar sensores virtuales a los UAV reales, aumentando así su percepción. Esto permite la realización de pruebas de estrategias de coordinación con UAV reales y sensores basados en modelos realizados a partir de datos anteriores, así como la simulación de fallos en los sensores para estudiar la robustez y la respuesta frente a estos eventos de la estrategia de coordinación.
- El potencial de mejora en la resolución de las misiones por la integración de información con los UAV simulados. Mediante los sensores virtuales, basados en modelos que se ajustan a la realidad, los UAV simulados proporcionan estimaciones que pueden ayudar al equipo de UAV reales a resolver la misión.

Como un pensamiento final, hay dos aspectos generales que siempre han estado presentes en este trabajo. El primero es que para la creación o construcción de cualquiera de las distintas piezas que componen el sistema (estrategias de coordinación, UAV reales, modelos de dispersión de contaminantes, etc.) se ha utilizado siempre el elemento más sencillo que pudiera hacer frente razonablemente a las especificaciones. La plataforma propuesta y su metodología incremental ayudan a probar cómo funciona y si su rendimiento real resulta ser suficiente para cumplir con los requisitos. El segundo aspecto general es que las preocupaciones de seguridad deben tenerse siempre en cuenta. Este doble aspecto ha estado presente a lo largo del desarrollo de todo este trabajo.

7.1. Trabajo futuro

El trabajo realizado en esta tesis doctoral ha permitido abrir una línea de investigación que continuará su desarrollo en los próximos años.

A partir del trabajo realizado para integrar los UAV reales en la plataforma de realidad mixta se tratará de añadir una mayor variedad en lo que se refiere a los UAV empleados. Así, toda la metodología y la plataforma están pensadas para permitir trabajar con UAV heterogéneos, tanto en lo que se refiere a los propios UAV como a la instrumentación equipada en los mismos. Sin embargo,

debido a la extensión del presente trabajo, el mismo se ha realizado con un equipo de UAV homogéneos de ala fija. Para demostrar la capacidad de funcionamiento con hardware completamente heterogéneo convendría integrar en la plataforma una mayor variedad tanto de controladores como de hardware. Una opción para ello es la de integrar un UAV de ala rotatoria o multirrotor en la plataforma, lo que plantea una serie de retos a superar:

En primer lugar el uso de simuladores diferentes, ya que el comportamiento dinámico de ambos UAV es diferente. Mientras que los UAV de ala fija generan la sustentación a partir de la depresión generada por el perfil aerodinámico del ala, los multirrotores lo consiguen a partir del empuje de los rotores. Tomando partido de la capacidad de la plataforma de emplear diferentes simuladores se podría, ejecutar los UAV de ala fija en un simulador y los de ala rotatoria en otro, de forma simultánea.

En segundo lugar, las diferencias en el control: en los UAV de ala fija se modifican diversas superficies de control para conseguir rotar la aeronave y hacer que esta apunte a la trayectoria deseada. Por el contrario, en los multirrotores, se varía la potencia entregada a los rotores, de forma que por la diferencia de empuje uno de los lados tiende a caer y el otro a levantarse. Una vez inclinado el propio empuje de los rotores genera un desplazamiento lateral. Esto provoca que los controladores de ambas aeronaves tienden a ser diferentes, por lo que habría que desarrollar en el middleware una traducción de las consignas provenientes del bloque de control de vuelo básico adaptadas a una nueva plataforma física completamente diferente.

A cambio de la superación de estos retos, se podrían generar opciones de colaboración interesantes aprovechando las ventajas de mayor autonomía y alcance en vuelo de los UAV de ala fija junto con la capacidad de mantener una posición fija en el espacio de los UAV de ala rotatoria. El desarrollo de estrategias de coordinación inteligentes capaces de aprovechar las sinergias obtenidas al reunir un equipo de UAV con características diferentes entre si supone un campo de trabajo prometedor.

En lo referente a las estrategias de coordinación, convendría probar también una estrategia completamente distribuida empleando varios UAV reales al igual que se han realizado las pruebas con el algoritmo evolutivo. Este tipo de pruebas que se han realizado, aunque únicamente en el entorno simulado, con el algoritmo *swarm* [Varela et al., 2011]. De esta forma se podrían establecer comparaciones en el rendimiento de varias estrategias. Además, permitiría poner en vuelo un UAV con una inteligencia independiente que, a diferencia de lo que sucede con las estrategias centralizadas, no requiriese del canal de comunicación con el nodo central. Así, se podrían estudiar las ventajas e inconvenientes de una y otra opción a la hora de desarrollar aplicaciones y misiones reales.

Con respecto a las posibilidades de la realidad mixta, se continuará explorando el potencial de la misma. El aumento de la percepción de los UAV reales mediante los sensores virtuales aporta la posibilidad de simular fallos en los mismos y comprobar así la robustez de las estrategias propuestas para la coordinación. De la misma forma, se pueden simular fenómenos y desarrollar estrategias de coordinación para el estudio de los mismos sin incurrir en los costes de instrumentar al equipo de UAV reales con sensores reales ni verse limitado por el espacio o el tiempo de ocurrencia de los mismos. Por su parte, la mejora de los resultados de los equipos de UAV por la aportación de información proveniente de los UAV simulados se podría verificar con pruebas reales. De esta forma se continuaría trabajando para obtener el máximo provecho de la plataforma de realidad mixta desarrollada para esta trabajo.

Bibliografía

- [Ardupilot, 2016] Ardupilot (2016). <http://ardupilot.org/>. 4.3
- [Barry, 2016] Barry, A. J. (2016). High-Speed Autonomous Obstacle Avoidance with Pushbroom Stereo by. pages 1–127. 1
- [Berkeley, 2016] Berkeley, U. (2016). <http://unmanned.berkeley.edu/>. 3.4.1
- [Bertuccelli et al., 2009] Bertuccelli, L., Choi, H., Cho, P., and How, J. (2009). Real-time multi-UAV task assignment in dynamic and uncertain environments. *American Institute of Aeronautics and Astronautics*, pages 1–16. 1, 3.3.2
- [Blender, 2016] Blender (2016). <https://www.blender.org/>. 6.8, 6.4.2, 7.1
- [Brezoescu et al., 2012] Brezoescu, a., Espinoza, T., Castillo, P., and Lozano, R. (2012). Adaptive Trajectory Following for a Fixed-Wing UAV in Presence of Crosswind. *Journal of Intelligent & Robotic Systems*, 69(1-4):257–271. 1
- [Cekmez et al., 2014] Cekmez, U., Ozsiginan, M., and Sahingoz, O. K. (2014). A UAV path planning with parallel ACO algorithm on CUDA platform. In *2014 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 347–354. 3.3.2
- [Choi et al., 2011] Choi, H., Kim, Y., and Kim, H. (2011). Genetic algorithm based decentralized task assignment for multiple unmanned aerial vehicles in dynamic environments. *International Journal of Aeronautical and Space Sciences*, 12(2):163–174. 3.3.2
- [Clough, 2002] Clough, B. (2002). Metrics, schmetrics! How the heck do you determine a UAV’s autonomy anyway? *Security*, (990):313–319. 3.1.2
- [Coiro et al., 2007] Coiro, D. P., Marco, A. D., and Nicolosi, F. (2007). A 6DOF Flight Simulation Environment for General Aviation Aircraft with Control Loading Reproduction. *Proceedings of the AIAA Modeling and Simulation Technologies Conference and Exhibit*, pages 1–23. 5.2.4.2

- [Cole, 2009] Cole, D. T. (2009). A Cooperative UAS Architecture for Information-Theoretic Search and Track. *Philosophy*, (March). 3.4.3
- [Cole et al., 2006] Cole, D. T., Sukkarieh, S., Göktoğuşan, A. H., Stone, H., and Hardwick-Jones, R. (2006). The development of a real-time modular architecture for the control of UAV teams. *Springer Tracts in Advanced Robotics*, 25:465–476. 3.4.3
- [Cole et al., 2010] Cole, D. T., Thompson, P., Goktogan, a. H., and Sukkarieh, S. (2010). System Development and Demonstration of a Cooperative UAV Team for Mapping and Tracking. *The International Journal of Robotics Research*, 29(11):1371–1399. 3.4.3
- [Dunbabin and Marques, 2012] Dunbabin, M. and Marques, L. (2012). Robots for Environmental Monitoring: Significant Advancements and Applications. *IEEE Robotics & Automation Magazine*, (March):24–39. 1
- [Flightgear, 2016] Flightgear (2016). <http://www.flightgear.org/>. 5.2.4.2, 6.4.1
- [Freitas et al., 2010] Freitas, E. P. D., Allgayer, R. S., Heimfarth, T., Larsson, T., Pereira, C. E., and Ferreira, A. M. (2010). Coordination Mechanism and Customizable Hardware Platform to Provide Heterogeneous Wireless Sensor Networks Support. *Electrical Engineering*, (i):77–88. 3.3.2
- [Frew et al., 2012] Frew, B. E. W., Elston, J., Argrow, B., Houston, A., and Rasmussen, E. (2012). Sampling Severe Local Storms and Related Phenomena: Using Unmanned Aircraft Systems. *IEEE Robotics & Automation Magazine*, (February):85–95. 3.2.2
- [Garvey et al., 2011] Garvey, J., Kehoe, B., Basso, B., Godwin, M., Wood, J., Love, J., Liu, S.-Y., Kim, Z., Jackson, S., Fallah, Y., Fu, T., Sengupta, R., and Hedrick, K. (2011). An Autonomous Unmanned Aerial Vehicle System for Sensing and Tracking. In *Infotech@Aerospace 2011*, Infotech@Aerospace Conferences. American Institute of Aeronautics and Astronautics. 3.4.1
- [Giard et al., 2007] Giard, A. R., Dharba, S., Pachter, M., and Chandler, P. R. (2007). Stochastic Dynamic rogramming for Uncertainty Handling in UAV Operations. In *2007 American Control Conference*, pages 1079–1084. 3.3.2
- [Glover, 1986] Glover, F. (1986). Future paths for integer programming and links to artificial intelligence. *Computers & Operations Research*, 13(5):533–549. 3.3.1
- [Göktogan and Sukkarieh, 2009] Göktogan, A. H. and Sukkarieh, S. (2009). Distributed simulation and middleware for networked UAS. *Journal of Intelligent and Robotic Systems: Theory and Applications*, 54:331–357. 1

- [Han, 2009] Han, Y. (2009). An autonomous unmanned aerial vehicle-based imagery system development and remote sensing images classification for agricultural applications. *Graduate Theses and Dissertations*. 1
- [Ho et al., 2013] Ho, D. T., Grotli, E. I., Sujit, P. B., Johansen, T. A., and Sousa, J. B. D. (2013). Performance evaluation of cooperative relay and Particle Swarm Optimization path planning for UAV and wireless sensor network. In *2013 IEEE Globecom Workshops (GC Wkshps)*, pages 1403–1408. 3.3.2
- [Honkavaara et al., 2013] Honkavaara, E., Saari, H., Kaivosoja, J., Pölonen, I., Hakala, T., Litkey, P., Mäkynen, J., and Pesonen, L. (2013). Processing and assessment of spectrometric, stereoscopic imagery collected using a lightweight UAV spectral camera for precision agriculture. *Remote Sensing*, 5(10):5006–5039. 1, 3.2.2
- [Jackson et al., 2008] Jackson, S., Tisdale, J., Kamgarpour, M., Basso, B., and Hedrick, J. K. (2008). Tracking controllers for small UAVs with wind disturbances: Theory and flight results. *Proceedings of the IEEE Conference on Decision and Control*, pages 564–569. 3.4.1
- [Jakob et al., 2012] Jakob, M., Pechoucek, M., Cap, M., Novak, P., and Vanek, O. (2012). Mixed-Reality Testbeds for Incremental Development of HART Applications. *IEEE Intelligent Systems*, 27(2):19–25. 3.4.2
- [JavaProp, 2016] JavaProp (2016). <http://www.mh-aerotoools.de/airfoils/javaprop.htm>. 6.4.2
- [JSBSim, 2016] JSBSim (2016). <http://jsbsim.sourceforge.net/>. 5.2.4.2, 6.4.1
- [Karaman et al., 2015] Karaman, S., Koyuncu, E., and Inalhan, G. (2015). Innovative Collaborative Task Allocation for UAVs. In Valavanis, K. P. and Vachtsevanos, G. J., editors, *Handbook of Unmanned Aerial Vehicles*, pages 1601–1617. Springer Netherlands, Dordrecht. 1
- [Kim, 2004] Kim, J. (2004). *Autonomous Navigation for Airborne Applications*. PhD thesis. 3.4.3
- [Kooistra et al., 2013] Kooistra, L., Suomalainen, J., and Iqbal, S. (2013). Crop monitoring using a light-weight hyperspectral mapping system for unmanned aerial vehicles: first results for the 2013 season. (September):1–7. 1, 3.2.2
- [Kruijff et al., 2012] Kruijff, G., Colas, F., Svoboda, T., van Diggelen, J., Balmer, P., Pirri, F., and Worst, R. (2012). Designing Intelligent Robots for Human-Robot Teaming in Urban Search & Rescue. *AAIA Technical Report*. 3.2.2

- [Liu and Sycara, 1996] Liu, J. and Sycara, K. (1996). Multiagent coordination in tightly coupled task scheduling. *Proceedings of the Second International Conference on Multi-Agent Systems*, (3):181–188. 3.3.2
- [Lopez Peña et al., 2016] Lopez Peña, F., Caamaño, P., Varela, G., Orjales, F., and Deibe, A. (2016). Setting up a mixed reality simulator for using teams of autonomous uavs in air pollution monitoring. *International Journal of Sustainable Development and Planning*, 11(4):616–626. 5.2.4.2
- [Maza et al., 2011] Maza, I., Caballero, F., Capitán, J., Martínez-de Dios, J. R., and Ollero, A. (2011). Experimental Results in Multi-UAV Coordination for Disaster Management and Civil Security Applications. *Journal of Intelligent {&} Robotic Systems*, 61(1):563–585. 3.2.2
- [Maza et al., 2010] Maza, I., Kondak, K., Bernard, M., and Ollero, A. (2010). Multi-UAV cooperation and control for load transportation and deployment. *Journal of Intelligent and Robotic Systems: Theory and Applications*, 57(1-4):417–449. 1
- [Merino et al., 2010] Merino, L., Caballero, F., Martínez-de dios, J. R., and Maza, I. (2010). Automatic Forest Fire Monitoring and Measurement using Unmanned Aerial Vehicles. *VI International Conference on Forest Fire Research*, page 15. 3.2.2
- [Miller, 2011] Miller, D. (2011). Autonomous Vehicle Laboratory for Sense and Avoid Research and Hardware-in-the-Loop Simulations. *Report PPT*, pages 1–2. 5.2.4.2
- [MIT, 2016] MIT (2016). <http://web.mit.edu/drela/Public/web/avl/>No Title. 6.4.2
- [Oh et al., 2015] Oh, H., Shin, H.-S., Kim, S., Tsourdos, A., and White, B. A. (2015). Cooperative Mission and Path Planning for a Team of UAVs. In Valavanis, K. P. and Vachtsevanos, G. J., editors, *Handbook of Unmanned Aerial Vehicles*, pages 1509–1545. Springer Netherlands, Dordrecht. 1
- [Px4, 2016] Px4 (2016). <http://px4.io/>. 4.3
- [Rathinam et al., 2005] Rathinam, S., Kim, Z., Soghikian, A., and Sengupta, R. (2005). Vision Based Following of Locally Linear Structures using an Unmanned Aerial Vehicle. *Proceedings of the 44th IEEE Conference on Decision and Control*, pages 6085–6090. 3.4.1
- [Ryan et al., 2007] Ryan, A., Tisdale, J., Godwin, M., Coatta, D., Nguyen, D., Spry, S., Sengupta, R., and Hedrick, J. K. (2007). Decentralized Control of

- Unmanned Aerial Vehicle Collaborative Sensing Missions. In *2007 American Control Conference*, pages 4672–4677. 3.4.1
- [Saber and Murray, 2003] Saber, R. and Murray, R. (2003). Consensus protocols for networks of dynamic agents. *Proceedings of the 2003 American Control Conference, 2003.*, 2:951–956. 3.3.2
- [Scerri et al., 2008] Scerri, P., Gonten, T. V., Fudge, G., and Owens, S. (2008). Transitioning multiagent technology to UAV applications. *Proceedings of the 7th Int. Conf. on Autonomous Agents and Multiagent Systems, (AAMAS):89–96.* 3.3
- [Selecký et al., 2015] Selecký, M., Rollo, M., Engineering, E., and Technical, C. (2015). Framework for incremental development of complex unmanned aircraft systems. pages 1–9. 1, 3.4.2
- [Sengupta, 2008] Sengupta, R. (2008). Target detection and position likelihood using an aerial image sensor. *2008 IEEE International Conference on Robotics and Automation*, pages 59–64. 3.4.1
- [Silver et al., 1980] Silver, E. A., Victor, R., Vidal, V., and de Werra, D. (1980). A tutorial on heuristic methods. *European Journal of Operational Research*, 5(3):153–162. 3.3
- [Southampton, 2016] Southampton (2016). http://www.southampton.ac.uk/engineering/news/2011/07/28_southampton_engineers_fly_the_worlds_first_printed_aircraft.page. 4.3
- [Suomalainen et al., 2014] Suomalainen, J., Anders, N., Iqbal, S., Roerink, G., Franke, J., Wenting, P., Hünninger, D., Bartholomeus, H., Becker, R., and Kooistra, L. (2014). A Lightweight Hyperspectral Mapping System and Photogrammetric Processing Chain for Unmanned Aerial Vehicles. *Remote Sensing*, 6(11):11013–11030. 1
- [Tahir and Brooker, 2009] Tahir, N. and Brooker, G. (2009). Feasibility of UAV Based Optical Tracker for Tracking Australian Plague Locust. *Australasian Conference on Robotics and Automation (ACRA 2009). Sydney, Australia.* 3.2.2
- [Valavanis and Vachtsevanos, 2015] Valavanis, K. and Vachtsevanos, G. (2015). *Handbook of Unmanned Aerial Vehicles.* 1, 3.1
- [Varela et al., 2011] Varela, G., Caamamo, P., and Orjales, F. (2011). Swarm intelligence based approach for real time UAV team coordination in search operations. *Third World Congress on Nature and Biologically Inspired Computing*, pages 365–370. 5.2.4.1, 5.4.4, 7.1

- [Varela et al., 2014a] Varela, G., Caamaño, P., Orjales, F., Deibe, Á., López-Peña, F., and Duro, R. J. (2014a). Autonomous UAV based search operations using constrained sampling evolutionary algorithms. *Neurocomputing*, 132:54–67. 5.2.4.2, 6.5.1, 6.6.3
- [Varela et al., 2014b] Varela, G., Caamaño, P., Orjales, F., Deibe, A., Lopez-Pena, F., and Duro, R. J. (2014b). Differential Evolution in Constrained Sampling Problems. In *2014 IEEE Congress on Evolutionary Computation (CEC)*, pages 2375–2382. 3.3.2
- [Wilson et al., 2015] Wilson, D. B., Ali, H. G., and Sukkarieh, S. (2015). Guidance and Navigation for UAV Airborne Docking. *Robotics: Science and Systems*. 3.4.3
- [Zanakis and Evans, 1981] Zanakis, S. H. and Evans, J. R. (1981). Heuristic “optimization”: Why, When, and How to Use It. *Interfaces*, 11(5):84–91. 3.3
- [Zhang and Duan, 2015] Zhang, X. and Duan, H. (2015). An improved constrained differential evolution algorithm for unmanned aerial vehicle global route planning. *Applied Soft Computing*, 26:270–284. 3.3.2

Índice de figuras

3.1. MQM-57 Falconer, uno de los primeros UAV para toma de datos	15
3.2. Clasificación de las metaheurísticas	27
3.3. Esquema de la estructura software de la plataforma de la Universidad Técnica Checa	33
3.4. Niveles de software definidos en AFDAS	36
4.1. Categorías de la estructura software propuesta	43
4.2. Bloques funcionales propuestos para los UAV	45
4.3. Bloques funcionales propuestos para la estación en tierra	50
4.4. Niveles de software propuestos	54
4.5. Esquema del proceso de adquisición datos	56
4.6. Esquema del proceso de actuación	57
4.7. Esquema del proceso de desarrollo del modelo y el control de vuelo básico	64
5.1. Paradigma de uso de la plataforma	68
5.2. Estructura de la plataforma	69
5.3. Bloques funcionales para el software de los UAV	71
5.4. Proceso de selección del modo de control	72
5.5. Estructura del control de vuelo básico	74
5.6. Estructura del bloque de evasión de colisiones	78
5.7. Bloques funcionales para el software de la estación de tierra	82
5.8. Simulación de varios UAV con jME3	86
5.9. Simulación de varios UAV con FlightGear	87

5.10. Esquema general de la estructura del middleware	92
5.11. Lógica de control inicial del piloto automático con Eagle Tree . .	97
5.12. Lógica de control final del piloto automático	98
5.13. Estructura de la comunicación entre UAV reales y simulados . .	101
5.14. Patrón de radiación de las antenas de las radiobalizas	107
5.15. Montaje realizado para homogeneizar la recepción	108
5.16. Comparación entre la intensidad del modelo y la radiobaliza real	109
6.1. Vista de la central térmica de As Pontes	113
6.2. Disposición de la pluma	115
6.3. Visualización del modelo de la pluma sobre Google Maps	121
6.4. Parte del equipo de UAV reales	123
6.5. Esquema del conexionado de la electrónica del UAV	125
6.6. Esquema de la lógica de control del UAV	127
6.7. Explicación de los parámetros del OSD	128
6.8. Captura del modelo gráfico del UAV en Blender [Blender, 2016] .	132
6.9. Diagrama de flujo del algoritmo evolutivo para el UAV	135
6.10. Diagrama de flujo del controlador central del algoritmo evolutivo	137
6.11. Trayectoria seguida por el UAV durante uno de los vuelos de ajuste del controlador	142
6.12. Estimación de la velocidad del viento	143
6.13. Concentraciones de CO2 medidas durante un vuelo representado con KML de Google Earth	145
6.14. Secuencia de instantáneas del API de Google Maps obtenidas utilizando un algoritmo CS-DE para el controlador del equipo. .	147
6.15. Zona de vuelo permitida actualmente para los UAV	148
6.16. Secuencia de instantáneas con 4 UAV simulados tras el reajuste de la misión	149
6.17. Secuencia de instantáneas con 1 UAV real y 3 simulados	150
6.18. Secuencia de instantáneas con 2 UAV reales y 2 simulados	152

Índice de tablas

3.1. Clasificación de UAV según la OTAN	14
6.1. Clases de estabilidad atmosférica según Pasquill	117
6.2. Perfil de viento en función de la clase de estabilidad atmosférica.	118
6.3. Sigma para un área rural	119