



UNIVERSITAT  
POLITÈCNICA  
DE VALÈNCIA

– **TELECOM** ESCUELA  
TÉCNICA **VLC** SUPERIOR  
DE INGENIERÍA DE  
TELECOMUNICACIÓN

UNIVERSITAT POLITÈCNICA DE VALÈNCIA

Escuela Técnica Superior de Ingeniería de  
Telecomunicación

Prueba de concepto de una red de radares en bandas  
milimétricas para detección y seguimiento de drones  
pequeños

Trabajo Fin de Máster

Máster Universitario en Ingeniería de Telecomunicación

AUTOR/A: Gómez Fresneda, Juan Carlos

Tutor/a: Balbastre Tejedor, Juan Vicente

Cotutor/a: Vico Navarro, Joaquín

CURSO ACADÉMICO: 2022/2023

---

## Resumen

En los últimos años, hemos sido testigos del creciente uso de aeronaves no tripuladas (UAS) en numerosas aplicaciones a nivel global. Sin embargo, uno de los desafíos clave en el uso seguro de estos UAS es su baja capacidad de detección en sistemas de radar convencionales debido a su reducida Sección Recta Radar (RCS) y su construcción mayoritariamente de plástico. La solución propuesta en este proyecto implica la integración de dos radares operando a 24 GHz en una red de detección, además del desarrollo de algoritmos especializados en detección, asociación y seguimiento de estas aeronaves no tripuladas. El objetivo central de este proyecto es mejorar significativamente la capacidad de gestión del tráfico de UAS y fortalecer la seguridad del espacio aéreo. Para evaluar la efectividad de este enfoque, se implementará un filtro JPDA (Asociación Conjunta de Datos Probabilísticos) para lograr la asociación de las trayectorias generadas. Este proyecto representa un paso importante hacia la optimización de la detección y gestión de UAS en el espacio aéreo, contribuyendo a una mayor seguridad y eficiencia en sus operaciones.

## Resum

En els últims anys, hem sigut testimonis del creixent ús d'aeronaus no tripulades (UAS) en nombroses aplicacions a nivell global. No obstant això, un dels desafiaments clau en l'ús segur d'aquests UAS és la seua baixa capacitat de detecció en sistemes de radar convencionals a causa de la seua reduïda Secció Recta Radar (RCS) i la seua construcció majoritàriament de plàstic. La solució proposada en aquest projecte implica la integració de dos radars operant a 24 GHz en una xarxa de detecció, a més del desenvolupament d'algorismes especialitzats en detecció, associació i seguiment d'aquestes aeronaus no tripulades. L'objectiu central d'aquest projecte és millorar significativament la capacitat de gestió del trànsit de UAS i enfortir la seguretat de l'espai aeri. Per a avaluar l'efectivitat d'aquest enfocament, s'implementarà un filtre JPDA (Associació Conjunta de Dades Probabilístiques) per a aconseguir l'associació de les trajectòries generades. Aquest projecte representa un pas important cap a l'optimització de la detecció i gestió de UAS en l'espai aeri, contribuint a una major seguretat i eficiència en les seues operacions.

## Abstract

In recent years, we have witnessed the increasing use of unmanned aerial vehicles (UAS) in numerous applications globally. However, one of the key challenges in the safe use of these UAS is their low detection capability in conventional radar systems due to their reduced Radar Straight Section (RCS) and their mostly plastic construction. The solution proposed in this project involves the integration of two radars operating at 24 GHz in a detection network, in addition to the development of specialised algorithms for detection, association and tracking of these unmanned aircraft. The central objective of this project is to significantly improve UAS traffic management capabilities and strengthen airspace safety. To evaluate the effectiveness of this approach, a JPDA (Joint Probabilistic Data Association) filter will be implemented to achieve the association of the generated trajectories. This project represents an important step towards optimising the detection and management of UAS in airspace, contributing to increased safety and efficiency in their operations.

# Índice general

## I Memoria

<b>1. Introducción</b>	<b>7</b>
1.1. Objetivos	8
1.1.1. Objetivo principal	8
1.2. Plan de Trabajo	9
1.3. Organización de la Memoria	10
<b>2. Metodología</b>	<b>11</b>
2.1. Radar FM-CW DK-sR-1200e	11
2.2. Matlab	12
2.3. Filtro de Kalman	13
2.4. Filtro JPDA	14
<b>3. Contexto</b>	<b>17</b>
3.1. SESAR Joint Undertaking	17
3.2. U-space	18
3.2.1. Fases de desarrollo del U-space	19
3.2.2. Beneficios Económicos y Sociales del U-space	21
3.3. Proyecto BUBBLES	21
3.4. Vigilancia Aérea	23
3.4.1. Sensores Empleados	24
3.4.2. Tipos de Vigilancia Aérea	25
3.4.3. Mejoras de los Sistemas de Vigilancia	26
3.4.4. Riesgos y Desafíos en la Vigilancia Aérea	27
3.5. Sistemas de Detección con Redes de Sensores	28
3.6. Modelo de Error del Radar Empleado	32
<b>4. Desarrollo y Resultados</b>	<b>33</b>
4.1. Generador de Trayectorias	33
4.1.1. Ejemplo Trayectorias Generadas	36
4.2. Filtro de Kalman	38
4.2.1. Ejemplo Filtrado de Kalman	41
4.3. Filtro JPDA	43
4.3.1. Asociación de Trayectorias Mediante JPDA	46
<b>5. Conclusiones y Trabajo Futuro</b>	<b>49</b>
5.1. Conclusiones	49
5.2. Trabajo Futuro	49

---

<b>Bibliografía</b>	<b>51</b>
<b>II Anexos</b>	
<b>A. Relación del Trabajo con los Objetivos de Desarrollo Sostenible de la Agenda 2030</b>	<b>55</b>
A.1. Objetivo 9: Industria, Innovación e Infraestructura . . . . .	56
A.2. Objetivo 11: Ciudades y Comunidades Sostenibles . . . . .	56

# Índice de figuras

1.1. Representación del Plan de Trabajo. . . . .	9
2.1. Datasheet . . . . .	12
2.2. Proceso Iterativo de Kalman. . . . .	13
3.1. SESAR Joint Undertaking . . . . .	18
3.2. U-space. . . . .	19
3.3. Niveles del U-space . . . . .	20
3.4. Proceso de Provisión de Separación . . . . .	22
3.5. Ejemplo de Sistema de Vigilancia Aérea . . . . .	23
3.6. Radar ADS-B . . . . .	24
3.7. Latitud y Longitud . . . . .	25
3.8. Vigilancia Aérea . . . . .	27
3.9. Arquitectura de Sistemas de Seguimiento . . . . .	29
3.10. Configuración de Array de Sensores . . . . .	30
4.1. Diagrama de Flujo Generador de Trayectorias . . . . .	35
4.2. Variable Trajectories . . . . .	36
4.3. Trayectorias Ideales . . . . .	37
4.4. Trayectorias Simuladas con Ruido . . . . .	37
4.5. Diagrama de Flujo Filtrado de Kalman . . . . .	40
4.6. Trayectorias Ideales Generadas . . . . .	41
4.7. Trayectorias Obtenidas con Ruido . . . . .	41
4.8. Trayectorias Filtradas . . . . .	42
4.9. Zoom Trayectorias Filtradas . . . . .	42
4.10. Diagrama de Flujo JPDA . . . . .	44
4.11. Continuación Diagrama de Flujo JPDA . . . . .	45
4.12. Variable CellList . . . . .	46
4.13. Pimera Iteración JPDA . . . . .	46
4.14. Segunda Iteración JPDA . . . . .	47
4.15. Tercera Iteración JPDA . . . . .	47
4.16. última Iteración de Trayectorias Confirmadas . . . . .	48
4.17. Variable ainfo . . . . .	48
A.1. Relación de los Objetivos de Desarrollo Sostenible con el TFM . . . . .	55

**Parte I**

**Memoria**



# Capítulo 1

## Introducción

Durante los últimos años, hemos sido testigos del extraordinario progreso dentro del sector de los sistemas de aeronaves no tripuladas (UAS) a nivel global. El interés dentro de las aeronaves no tripuladas se centra en el uso de drones, ya que han demostrado ser extremadamente versátiles y ágiles. Estas características permiten su uso en una amplia gama de aplicaciones, como la fotografía y filmación aérea, la vigilancia y monitoreo o la entrega de paquetes. La creciente popularidad de los drones ha llevado a un aumento significativo en su presencia en los cielos, lo que plantea una serie de desafíos y preocupaciones en términos de seguridad y regulación [1][2].

Para habilitar el potencial que presenta la aviación no tripulada, la Unión Europea está trabajando de manera activa en el desarrollo de un marco legal y operativo para una integración segura y efectiva en el espacio aéreo. En este sentido, aparece el concepto U-space, que hace referencia a un ecosistema digital que permite una gestión segura y automática del tráfico de aeronaves no tripuladas junto con el de la aviación tripulada [3]. El U-space contempla tecnologías de comunicación V2V (vehículo a vehículo), V2I (vehículo a infraestructura) y V2C (vehículo a controlador). Además, las aeronaves no tripuladas incluyen sistemas para detectar y evitar colisiones (Detect and Avoid). Este tipo de tecnologías permitirán una supervisión y gestión del tráfico de manera eficiente, proporcionando información en tiempo real sobre la ruta, posición y estado de las aeronaves no tripuladas. Además, la Comisión Europea ha instaurado la estrategia 2.0 para la aviación no tripulada. Esta se basa en la creación de un mercado Europeo de aeronaves no tripuladas a gran escala. Llevando a cabo operaciones mediante el uso de estas, lo que implica nuevos retos y oportunidades para el sector. Otro factor determinante para el avance de estas tecnologías innovadoras es el apoyo de la sociedad a su uso. Las autoridades a nivel nacional serán las encargadas de responder a las preocupaciones sobre ruido, privacidad y seguridad por parte de los ciudadanos. Algunos servicios previstos durante los próximos años son: Servicios de emergencia, vigilancia mediante drones civiles, entrega urgente de pequeños envíos (como muestras biológicas y medicamentos) o servicios de movilidad aérea innovadores, como los taxis aéreos [4].

Para desarrollar el potencial del mercado y de los servicios proporcionados por las aeronaves no tripuladas la Unión Europea requiere la identificación de componentes tecnológicos esenciales para este propósito, como la inteligencia artificial, la robótica y los semiconductores, así como los servicios espaciales y las telecomunicaciones móviles. Esta estrategia también creará sinergias entre los drones de uso civil y los de defensa [4].



A pesar de las numerosas ventajas de los drones, estos presentan diversos problemas de detectabilidad en el espacio aéreo debido a su diseño y características técnicas. Algunos problemas a destacar son: La baja firma infrarroja (IR), la baja emisión de señales electromagnéticas y la pequeña sección radar (RCS) [5]. Si nos centramos en su RCS, el valor puede variar dependiendo del tamaño, material, frecuencia de operación y ángulo de visión del radar. Para el caso de los drones de uso civil, su reducido tamaño así como los materiales que se emplean (plásticos/fibras de carbono) proporcionan valores reducidos de RCS. Es decir, el área que reflejan las ondas del radar es pequeña. Esto se traduce en problemas en la detección por parte de sistemas de radar convencionales, especialmente si las distancias son largas. Estos problemas de detectabilidad tendrán un impacto directo en la gestión aérea, aumentando la separación entre aeronaves para mantener los niveles de seguridad requeridos. Como consecuencia, esto conducirá a un uso ineficiente del espacio aéreo [2][6].

## 1.1. Objetivos

Como se ha mencionado previamente, la reducida RCS de los radares presenta problemas en la detección por parte de sistemas de radar convencionales. Para solventar este problema, el uso de radares en la banda milimétrica (30-300 GHz) se presenta como una posible solución, ya que la RCS se incrementa. Sin embargo, para alcances de varios kilómetros, el coste de los transmisores así como la elevada atenuación a dichas frecuencias supone una solución inviable. La alternativa propuesta en este proyecto es el uso de radares de corto-medio alcance integrados en una red de manera que operen como un único sistema detector. Adicionalmente, dadas las particularidades del sector de las aeronaves no tripuladas, sería deseable conformar estas redes con dispositivos COTS (commercial-off-the-shelf).

### 1.1.1. Objetivo principal

El trabajo propuesto se centrará en la integración de dos radares FM-CW DK-sR-1200e operando a 24 GHz. Para ello, se realizará una maqueta integrando ambos radares (conexión en red). Además, se desarrollarán algoritmos de detección, asociación y seguimiento, así como una interfaz de usuario del sistema. Tanto los algoritmos a realizar como la interfaz gráfica se implementarán en Matlab. Para poder lograr el resultado esperado, se definen los siguientes objetivos específicos:

- Generación de trayectorias de vuelo similares a una situación real.
- Establecimiento de parámetros (ruido, ángulo de visión, etc) y posición de ambos radares para conseguir valores esperados.
- Filtrado óptimo de las trayectorias para obtener medidas coherentes.
- Asociación de trackings para ambos radares.
- Ajuste de parámetros y código para extrapolarlo a un entorno de vuelo real.

## 1.2. Plan de Trabajo

Para este apartado se ha realizado un diagrama temporal en el que podemos observar el orden seguido para las diferentes tareas y el tiempo aproximado que se ha dedicado a cada una de ellas.



Figura 1.1: Representación del Plan de Trabajo.

### 1.3. Organización de la Memoria

La estructura que se ha seguido para organizar la memoria es la siguiente:

- El segundo capítulo se centra en la metodología del proyecto, justificando las herramientas empleadas así como las decisiones que se han tomado para alcanzar los objetivos establecidos.
- En el tercer capítulo detallaremos el estudio teórico del trabajo. Este nos servirá para comprender el contexto y la motivación del proyecto. Nos centraremos en el U-space, el desarrollo del proyecto BUBBLES, los sistemas de vigilancia aérea y el uso de sistemas de detección mediante redes de sensores.
- El cuarto capítulo recogerá el desarrollo en Matlab de los diferentes algoritmos empleados. Además, expondremos los resultados que se han obtenido para diferentes escenarios de trabajo.
- Por último, en el quinto capítulo compartiremos las conclusiones que hemos extraído y las posibles líneas futuras del proyecto.

## Capítulo 2

# Metodología

Para poder alcanzar los objetivos propuestos en el proyecto, hemos seguido una metodología que nos ayude a la toma de decisiones y al cumplimiento de los requisitos establecidos. La metodología establecida se dividirá en la elección de los radares que van a conformar nuestro sistema, las herramientas numéricas utilizadas, el proceso de filtrado para comprobar la eficacia de las medidas generadas y el uso de algoritmos para lograr la asociación de las trayectorias de vuelo. Para poder llevar a cabo la toma de estas decisiones se ha realizado un proceso de documentación estudiando los diferentes conceptos de nuestro proyecto.

### 2.1. Radar FM-CW DK-sR-1200e

Como establecimos en los objetivos del proyecto, el trabajo propuesto se centra en la integración de dos radares FM-CW DK-sR-1200e operando a 24 GHz. Estos radares, emitirán señales sin modular o con una modulación en amplitud o frecuencia según los requisitos requeridos. Para nuestro proyecto nos vamos a centrar en radares FM-CW, lo que implica el uso de una modulación en frecuencia continua para generar una señal que varía su frecuencia de manera lineal a lo largo del tiempo. Otro factor destacable de estos radares es su reducido tamaño, lo que supone su uso cada vez más frecuente en vehículos aéreos no tripulados (UAVs). Como se ha comentado previamente, el objetivo principal es validar el uso de radares de corto-medio alcance integrados en una red de manera que operen como un único sistema detector. Trabajar con radares operando a 24 GHz supone ventajas como un módulo compacto, una mayor inmunidad al ruido o la mejora en la detección de objetos de tamaño reducido. Ya que trabajar con radares en banda milimétrica supondría un coste muy elevado así como una atenuación elevada.

En la Figura 2.1 podemos observar el Datasheet para el radar utilizado. Se muestran los parámetros principales y los valores que pueden presentar. Para estudiar de manera más detallada algún parámetro haremos uso del manual del radar [7].

Parameter	Specification
<b>General</b>	
Modulation	CW/FMCW
Operating Frequency	24.00 GHz – 24.25 GHz (@ ISM band)
Number of channels	1 Transmit-Channel (Tx), 2 Recieve-Channels (Rx) with I/Q demodulator for each channel
Data Interface	SPI, CAN, Ethernet (with PoE)
Certification	CE Approval (Ethernet versión only)
<b>Antenna</b>	
Antenna Type	Integrated Patch Antenna
Anntena Characteristics	65º Azimuth and 24º Elevation
Anntena Polarization	linear
<b>Measurement</b>	
Measurement Range	0.6m – 307m (@ ISM band)
Range Resolution	Max. 0.6m (@ ISM band)
Measurement Speed	±3200 m/s (theoretical value @24GHz)
Speed Resolution (1-100 ms)	6.25 m/s – 6.25 cm/s (@24GHz)
Angle Measurement Accuracy	2° – 3°
<b>Operation Parameters</b>	
Frecuency Ramp Duration	1 ms – 100 ms
Update Rate	typ 10 Hz – 200 Hz
EIRP Output Power	typ 10 Hz – 200 Hz
Operating Temperature	min -40°C, max 60°C
<b>Power Supply</b>	
Operation Voltage	10.5 V – 13 V, 44 V – 54 V PoE
Standby Power	3.0 W
Operating Power	4.5 W
<b>Housing</b>	
Dimension (L x W x H)	98.0 mm x 87.0 mm x 42.5 mm (Housting) 114.0 mm x 87.0 mm x 42.2 mm (with Bushing)
Weight	280 g
Mounting	4 Mounting Holes (∅ 5 mm)
Cable and Connector	Shielded 9 Wire Cable (SPI/CAN) CAT-6 with RJ45 Female Plug (Ethernet)
Protection Code	IP 65

Figura 2.1: Datasheet

## 2.2. Matlab

Para la recogida de datos se han desarrollado algoritmos simulando un entorno real de vuelo, siendo capaz de emplearse en posteriores pruebas de campo. Tanto para el análisis de las diferentes medidas realizadas como para el desarrollo de los algoritmos de vuelo se ha utilizado la herramienta de cómputo numérico Matlab. Por tanto, con Matlab procesaremos todos los datos recogidos y analizaremos las trayectorias generadas. La elección de esta herramienta de cómputo numérico se debe a las numerosas ventajas que presenta. Algunos ejemplos son: La interfaz de usuario propor-

cionada (GUI) lo que facilita la interacción así como la ejecución del software, la amplia variedad de operaciones disponibles para trabajar con vectores y matrices o las numerosas bibliotecas de funciones y herramientas que permiten aprovechar funcionalidades avanzadas. Además, desarrollaremos también un filtro del Kalman para conseguir unos valores de trayectoria sin ruido próximos a la estimación previamente realizada. Por último, mediante un filtro JPDA conformaremos los trackings medidos por los radares.

## 2.3. Filtro de Kalman

El filtro de Kalman se trata de un conjunto de ecuaciones matemáticas que proporciona una estimación del estado de un proceso minimizando el promedio del error cuadrado. Este filtro tiene un alto poder en varios aspectos: es capaz de realizar estimaciones de estados pasados, presentes e incluso futuros, aún cuando la naturaleza del sistema que está siendo modelado es desconocida. Esta técnica se emplea en sistema de vigilancia, militares y sistemas robóticos automatizados. Si nos centramos en aviación, este proceso de filtrado se emplea para mejorar la precisión y confiabilidad de la estimación del estado de una aeronave, como su posición, velocidad u orientación [8].

Como el filtro de Kalman es un estimador iterativo, es capaz de evaluar los estados en tiempo real mediante el uso de las medidas de entrada actuales, el estado previo y la matriz de incertidumbre. El filtro consta de dos variables, que son el tiempo del estado y la covarianza del error [8][9].

- $X_{K/K}$ : Estimación del estado para el instante  $K$ .
- $P_{K/K}$ : Matriz de covarianza.

Como hemos descrito, el filtro de Kalman es un algoritmo iterativo que permite estimar de manera eficiente el estado de un sistema. En la Figura 2.2 observamos las iteraciones que realiza el proceso de estimación.

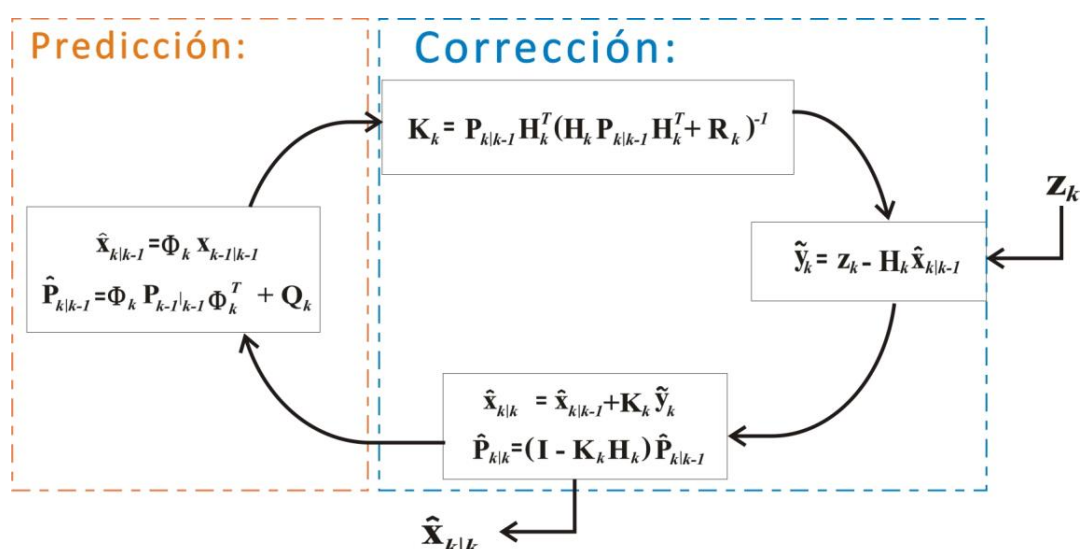


Figura 2.2: Proceso Iterativo de Kalman.

Para llevar a cabo este proceso de estimación, se utilizan dos métodos principales. En primer lugar, encontramos la actualización temporal, donde se realiza una predicción del estado del sistema. Posteriormente, viene la fase de medición, donde se corrige esta predicción empleando la información de la medida actual. Por tanto, el proceso se compone de dos tipos de ecuaciones. Tenemos las ecuaciones de predicción, que se encargan de estimar la relación entre las diferentes partes del sistema, y la ecuación de corrección, que ajustan esta estimación mediante la información de la medición[10].

En resumen, el filtro de Kalman busca optimizar el proceso de estimación combinando predicciones en el tiempo con correcciones basadas en medidas reales. Este proceso es crucial para obtener una estimación precisa y útil del estado del sistema, incluso cuando las mediciones contienen ruido [8].

## 2.4. Filtro JPDA

El filtro JPDA (Asociación Conjunta Probabilística de Datos) es una técnica de alto nivel empleada en la monitorización y rastreo de objetos, como drones, en entornos complejos y con múltiples fuentes de información. En el ámbito de los drones, el JPDA es una herramienta valiosa para mejorar la precisión y confiabilidad del seguimiento, especialmente cuando tenemos múltiples objetos en movimiento y varios sensores proporcionando datos. Este filtro, se basa en el filtro de Kalman clásico y el algoritmo de asociación de datos probabilísticos (PDA). En el caso de múltiples trayectorias, se construyen todos los posibles casos de asignación y se excluyen las asignaciones que se superponen. Esto permite una gestión más eficiente de los datos de detección y una mejor comprensión sobre los múltiples objetos en movimiento, lo que resulta en un seguimiento más preciso en entornos complejos [11][12].

El JPDA opera bajo las siguientes suposiciones [11]:

- El número de objetivos dentro del clutter del radar es conocido. Es decir, se asume que dentro de una nube de detecciones se conocen el número de objetivos reales.
- Las mediciones de un determinado objeto pueden caer en la región de validación de un objetivo vecino. Esta situación puede ocurrir durante varios instantes de muestreo y actúa como una interferencia persistente.
- El pasado del sistema se resume mediante una estadística aproximada que consiste en estimaciones de estado.
- Se asume que los estados están distribuidos de manera gaussiana con medias y covarianzas de acuerdo con la estadística aproximada que se ha establecido.
- Cada objetivo tiene un modelo de estado (transición) y un modelo de medición. Los modelos para los diferentes objetivos no tienen que ser idénticos.

El ruido de fondo se modela como una interferencia aleatoria con una distribución espacial uniforme. Este parámetro también se conoce como densidad de ruido de fondo y está relacionado con la probabilidad de detección falsa. Para cada instante de tiempo, el algoritmo JPDA sigue tres fases:

- Las probabilidades de asociación de medición se calculan de manera conjunta para todos los objetivos.
- Las probabilidades de asociación se calculan solo para el conjunto más reciente de mediciones. Ya que se asume que tenemos una estadística disponible.
- La estimación del estado se realiza por separado para cada objetivo.

En cuanto a las características de los JPDA destacan las siguientes:

- Gestión de incertidumbre: Los modos de operación de los drones están sujetos a condiciones cambiantes, como problemas meteorológicos y otros factores que pueden introducir cierto nivel de incertidumbre en las mediciones. El filtro JPDA es capaz de manejar esta incertidumbre al combinar información proveniente de múltiples fuentes. Esto resulta en una mejora en la precisión del seguimiento.
- Asociación de datos: En entornos de vuelo complejos, múltiples mediciones pueden coincidir con varios objetos. Mediante la consideración de todas las posibles combinaciones de mediciones y su asociación con los objetos que se consideran más probables se permite abordar esta situación de manera exitosa.
- Adaptabilidad: El JPDA es adaptable a cambios en la dinámica de los drones y las condiciones del entorno. Esto resulta vital, ya que los drones pueden alterar de manera instantánea su velocidad, dirección y altitud de manera impredecible. Esta capacidad de adaptación supone un gran refuerzo en la efectividad del seguimiento de drones.





## Capítulo 3

# Contexto

En este capítulo vamos a desarrollar la teoría necesaria para comprender el contexto y motivación de nuestro proyecto. Hablaremos sobre la institución SESAR-JU, el concepto de U-space a un nivel más detallado, el desarrollo del proyecto BUBBLES, el concepto de vigilancia aérea tanto a nivel general como enfocándonos en aeronaves no tripuladas y por último los sistemas de detección que emplean redes de sensores. Todos estos puntos nos ayudaran a entender mejor el desarrollo del proyecto así como las decisiones tomadas durante su realización.

### 3.1. SESAR Joint Undertaking

Como punto previo al U-space debemos hablar de SESAR-JU. Esta institución de carácter público-privado, se centra en acelerar el proceso de despliegue y uso de UAS en todo el continente Europeo. Promoviendo el uso de tecnologías innovadoras, procedimientos de mejora para la gestión del tráfico aéreo y una mejora en la eficiencia y sostenibilidad del tráfico Europeo. Esta iniciativa involucra una amplia gama de socios, incluyendo fabricantes de aviones, proveedores de servicios de navegación aérea, aerolíneas o instituciones de investigación [13].

Por tanto, mediante la investigación y la ejecución de distintos proyectos, se buscan soluciones para abordar varios desafíos, como el incremento del tráfico aéreo, la reducción de emisiones y las mejoras en seguridad. De este modo, a raíz de la solicitud hecha por la Comisión Europea, se comenzó a trabajar en la concepción de un proyecto relacionado con la integración de drones en el espacio aéreo de baja altitud. El objetivo se centra en garantizar que el crecimiento de esta nueva área se lleve a cabo de manera segura y eficiente, surgiendo así el concepto de U-space.



**Figura 3.1: SESAR Joint Undertaking**

### 3.2. U-space

Anteriormente definimos el U-space como un ecosistema digital que permite una gestión segura y automática del tráfico de aeronaves no tripuladas junto con el de la aviación tripulada. Por tanto, el propósito principal que se persigue con este ecosistema es lograr una gestión automatizada de UAS, especialmente en el espacio aéreo de baja altitud y en entornos urbanos, e integrarlo con seguridad con el sistema de gestión de tráfico aéreo ya existente para la aviación tripulada.[3]. El despliegue en España se realizará de manera progresiva, siguiendo el PANDU (Plan de Acción Nacional para el Despliegue del U-space).

En cuanto a los principios claves que se definen para la creación del U-space, destacan los siguientes [14]:

- Velar por la integridad y bienestar de los usuarios del espacio aéreo involucrados en las operaciones dentro del marco U-space, al igual que salvaguardar la seguridad de las personas que se encuentran en tierra.
- Ofrecer un sistema versátil, de gran capacidad y adaptable, capaz de ajustarse a las variaciones en la demanda, el volumen de tráfico y las tecnologías emergentes, sin dejar de convivir de manera armoniosa con la aviación tripulada.
- Habilitar la realización de múltiples operaciones con drones automatizados de manera eficiente y segura, todo ello bajo la supervisión responsable de los operadores.
- Garantizar sin ningún tipo de distinción un acceso equitativo e imparcial al espacio aéreo para todos los usuarios involucrados.

- Provisionar en todo momento un servicio competitivo, apoyando los modelos ya existentes en la industria de los operadores de drones.
- Minimizar los costes de despliegue y operación aprovechando en la medida de lo posible los servicios e infraestructura del sector aeronáutico, así como los servicios de comunicaciones móviles.
- Acelerar todo el proceso de despliegue adoptando diferentes tecnologías y estándares de otros sectores que satisfagan las necesidades del U-space.
- Seguir un enfoque impulsado por el rendimiento mientras se establecen requisitos apropiados para la seguridad ciudadana así como para el impacto ambiental.



**Figura 3.2: U-space.**

### 3.2.1. Fases de desarrollo del U-space

Por otra parte, hemos comentado que el establecimiento de mecanismos y reglas de despliegue se realizará de manera progresiva. Es por ello, que se han establecido las siguientes fases de desarrollo.

- U1: El enfoque se dirige hacia los servicios fundamentales del espacio aéreo no tripulado, los cuales comprenden el registro electrónico, la identificación electrónica y la delimitación geográfica establecida. El registro electrónico será obligatorio para todos los operadores de drones exceptuando los drones con un peso inferior a 250 gramos. Por otra parte, la identificación electrónica permitirá a las autoridades identificar un dron así como consultar la información establecida en el registro.
- U2: Se refiere a los servicios iniciales para la gestión de drones, que abarcan la planificación de vuelo, la aprobación de vuelos, el seguimiento de los vuelos y la interconexión con el tráfico aéreo convencional.

- U3: Los servicios avanzados se enfocan en operaciones más complejas en áreas con alta densidad de drones, y abarcan la gestión y asistencia para la detección de conflictos. Estas funcionalidades se denominan DAA (Detect and Avoid). Además, los servicios avanzados también incluyen capacidades para el seguimiento y monitoreo en tiempo real de los drones, la gestión de rutas más complejas y la integración con sistemas de navegación avanzados. Estas mejoras contribuirán a una operación más eficiente y segura de los drones en diversos escenarios, permitiendo un mayor aprovechamiento del espacio aéreo de manera coordinada y confiable.
- U4: La última fase se concentra en los servicios completos del U-space, con el objetivo de proporcionar niveles extremadamente elevados de automatización, conectividad y digitalización tanto para los drones como para el entorno U-space en su conjunto. Esto permitirá alcanzar un nivel avanzado de operaciones autónomas, asegurando una gestión eficiente y segura del espacio aéreo no tripulado, y facilitando una integración óptima con el tráfico aéreo tradicional.

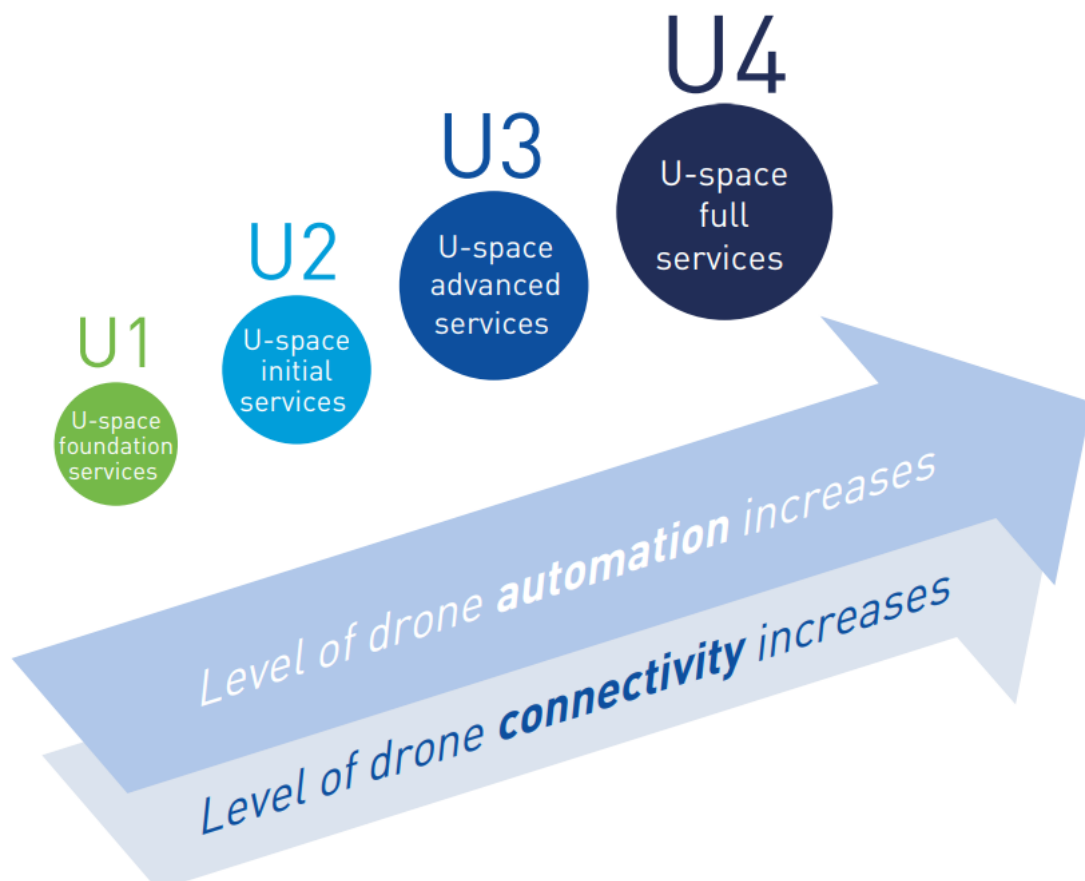


Figura 3.3: Niveles del U-space

### 3.2.2. Beneficios Económicos y Sociales del U-space

En relación a los beneficios proporcionados por el U-space a nivel Europeo, nos vamos a centrar en tres grupos: En primer lugar, tenemos a los operadores de drones. Estos experimentarán un acceso al espacio aéreo libre, justo y equitativo, así como una ampliación en la cobertura y alcance de las operaciones realizadas. En segundo lugar, tenemos a los ciudadanos, surgirán servicios novedosos e innovadores gracias al uso de las aeronaves no tripuladas, que impulsarán un avance tecnológico positivo para la calidad de vida de las personas, contribuyendo a la reducción de ruido y contaminación en entornos urbanos. Por último, en el ámbito de los negocios, se fomentará el desarrollo de nuevos modelos de negocio, estimulando el crecimiento de empleo y cuota de mercado, también se promoverá la digitalización y automatización.

## 3.3. Proyecto BUBBLES

El proyecto BUBBLES (BUilding Basic BLocks for a U-space SEparation) es una iniciativa respaldada por SESAR-JU para desarrollar el U-space de manera eficiente. El objetivo principal del proyecto es facilitar la gestión efectiva de la separación de tráfico aéreo, garantizando la prevención de posibles colisiones. Además, una parte fundamental de BUBBLES se enfoca en identificar los requisitos de seguridad y el rendimiento de los servicios empleados en el contexto U-space [15].

Desde los comienzos de la aviación, la seguridad ha sido un factor clave. Para mantener unos niveles aceptables de riesgo de colisión se estableció la gestión de conflictos. Esta está distribuida en tres capas:

- Gestión estratégica del conflicto
- Provisión de separación
- Evitación de colisiones

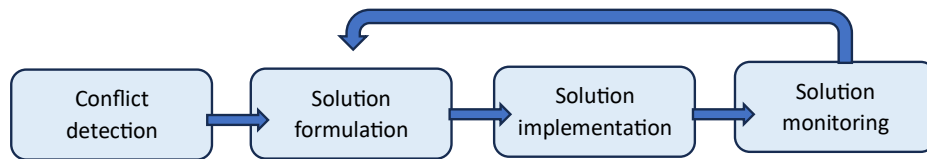
La gestión de conflictos en la aviación tradicional se caracteriza por la intervención activa de los actores implicados, la baja densidad de operaciones y la separación centralizada que establece ATC (Control del Tráfico Aéreo). Esto es posible ya que la aviación tradicional suele realizar trayectorias repetitivas con actuaciones similares en todas las ocasiones. Sin embargo, el U-space presenta una cantidad de escenarios muy diferentes, unos altos niveles de automatización y una mayor densidad de operaciones. Debido a esto, el empleo de los procesos tradicionales para la gestión de conflictos no es la opción más óptima [15][16].

Teniendo en cuenta las diferencias entre estos dos escenarios, BUBBLES propone un concepto de operaciones (ConOps) para la gestión de conflictos. Mediante el ConOps propuesto en el proyecto se proporcionará a los operadores de drones la información requerida para garantizar la separación entre vehículos, siguiendo unos criterios específicos dentro del U-space. El escenario donde se propone el uso de este ConOps es el nivel U3 de la figura 3.5, puesto que tenemos situaciones con una densidad relativamente alta de operaciones no tripuladas, siendo necesario mitigar el riesgo de colisión en el aire mediante un conjunto de medidas de seguridad enfocadas principalmente en la provisión de separación. Aunque las propuestas de el ConOps establecido se podrían aplicar para cualquier elemento que pueda causar una colisión, como objetos fijos, elementos del terreno

o aeronaves, su implementación se ha enfocado específicamente en considerar que el riesgo de colisión se origina por otra aeronave, ya sea tripulada o no tripulada [16].

Como mencionamos anteriormente, la gestión del tráfico aéreo se enfoca en establecer una provisión de separación dentro de un U-space proporcionado. Consiguiendo así mitigar todos los conflictos siguiendo las mismas reglas y procesos. De esta manera, se garantiza una gestión coherente y eficiente para evitar cualquier tipo de colisión.

Para ello se define la provisión de separación como un proceso iterativo que consta de cuatro pasos [16].



**Figura 3.4: Proceso de Provisión de Separación**

Como primer paso de este proceso tenemos la **detección de conflictos**. Se basa en la última posición reportada por las aeronaves implicadas y su trayectoria prevista. En este proceso, se predice la distancia esperada en el punto de mayor aproximación (CPA) y el tiempo restante hasta el CPA dentro de un horizonte de conflicto específico. Si la distancia CPA resulta inferior a las distancias mínimas de separación y el tiempo hasta el CPA es inferior a un valor previamente establecido, se activará un conflicto táctico.

En segundo lugar tenemos la **formulación de la solución**. Involucra la designación de un agente de separación, el establecimiento de los modos de separación, la elección de las distancias mínimas y los métodos a emplear para evitar conflictos. La elaboración de la solución depende en gran medida de la capacidad de intervención de humanos y sistemas para detectar y resolver un conflicto.

- Elección del agente de separación: El agente puede ser tanto un humano (opcionalmente apoyada por un sistema electrónico) o un sistema totalmente autónomo. Será el responsable de gestionar la separación entre las aeronaves cuando se detecte un conflicto.
- Elección de modo de separación: Consiste en un conjunto de condiciones, normas y procedimientos relacionados con las distancias mínimas y los métodos de separación que se deben aplicar.
  - Separación mínima: Son las distancias a las que las aeronaves deben permanecer para mantenerse alejada de un posible conflicto, de manera que se alcance un nivel objetivo de seguridad (TLS).
  - Métodos de separación: Son un conjunto de maniobras específicas que deben ejecutarse para poder mantener las distancias mínimas de separación.
  - Reglas y condiciones: Especifican la forma en la que el agente de separación debe aplicar las distancias mínimas de separación y los distintos métodos.

En tercer lugar tenemos la **implementación de la solución**. Consiste en un proceso compuesto por tres pasos:

- Se notifica al agente que está controlando la aeronave la solución que se ha definido en el paso previo.
- El agente ejecuta las maniobras requeridas para evitar el obstaculo que provocaría la colisión.
- El Sistema de Aeronave No Tripulada (UAS) realiza las maniobras ejecutadas por el agente a cargo del control. La implementación de la solución depende en gran medida del nivel de automatización del sistema.

Por último, tenemos el **seguimiento de la solución ejecutada**. BUBBLES contempla diferentes agentes encargados de supervisar la implementación de la solución que se ha establecido y advertir al agente de separación por si el resultado no fuese eficaz para evitar la colisión.

### 3.4. Vigilancia Aérea

El concepto de vigilancia aérea hace referencia al proceso de supervisar y rastrear el movimiento y posición de aeronaves en el espacio aéreo. Para ello, se cuenta con un sensor (o red de sensores) que detecta la aeronave, calcula su posición y obtiene datos como el identificador, nivel, etc. Posteriormente, estos datos se presentarán al controlador mediante un display. Las principales aplicaciones de la vigilancia son el control de área y el control de aproximación, aunque también es posible utilizarla en la torre (por ejemplo, el servicio A-SMGCS) o en el espacio aéreo no controlado donde se proporcionan servicios de Asesoramiento, de Información de Vuelo o de Alerta [17].

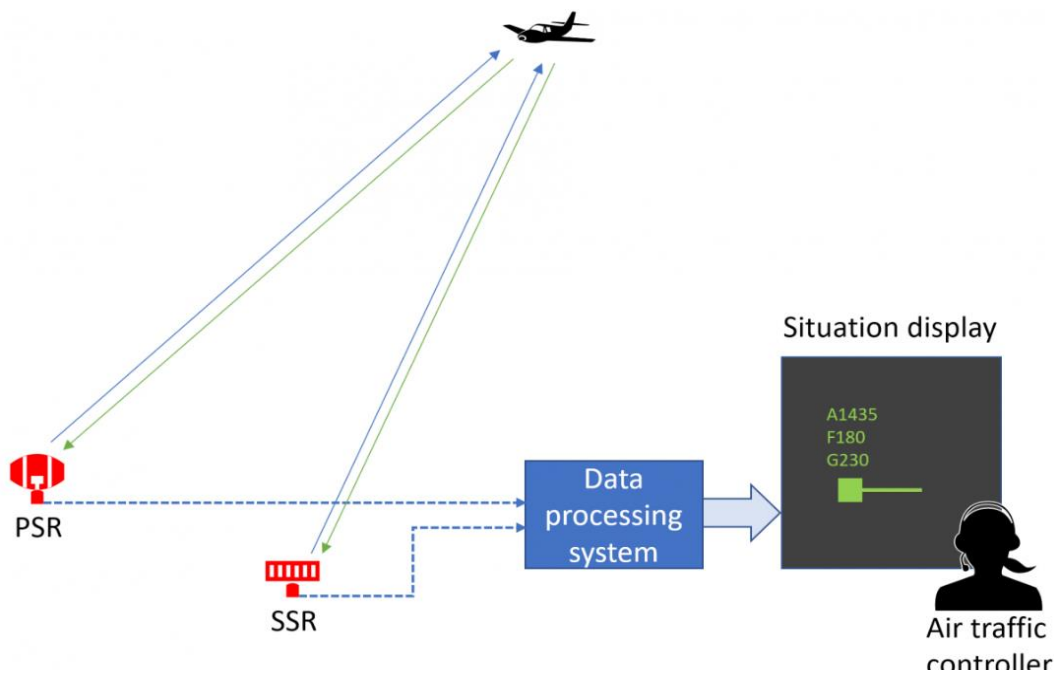


Figura 3.5: Ejemplo de Sistema de Vigilancia Aérea



### 3.4.1. Sensores Empleados

Hoy en día los sensores más empleados para el proceso de vigilancia aérea son los siguientes:

- Radar Primario de Vigilancia (PSR): En primera instancia, este equipo emite una onda electromagnética. Mediante el procesado de la onda que se recibe reflejada se determina la posición del objetivo (el retardo se utiliza para el cálculo de la distancia y la posición de la antena proporciona el azimut).
- Radar Secundario de Vigilancia (SSR): Este sensor emite señales especiales que son recibidas por el equipo a bordo (transpondedor). La respuesta proporcionada por el transpondedor es recibida por la antena del radar. Además de la información adicional que se incluye en la respuesta, esta se emplea para determinar la posición de la aeronave.
- Multilateración (MLAT): Se trata de un conjunto de sensores que utiliza las respuestas del transpondedor para calcular la posición de la aeronave y extraer datos adicionales. El transpondedor se puede activar mediante una consulta del SSR o por una consulta MLAT designada de forma previa. El cálculo de la posición de la aeronave se realiza mediante la diferencia de tiempo de llegada de la señal en los diferentes sensores.
- Transmisión Automática Dependiente de Vigilancia (ADS-B): La aeronave realiza el cálculo de su posición empleando sus sistemas de navegación (GNSS,INS,etc.). A diferencia del SSR y el MLAT no es necesaria ninguna consulta. La señal se transmite en intervalos cortos y regulares, cuando es recibida, se decodifica por estaciones terrestres y se extrae así la posición de la aeronave.

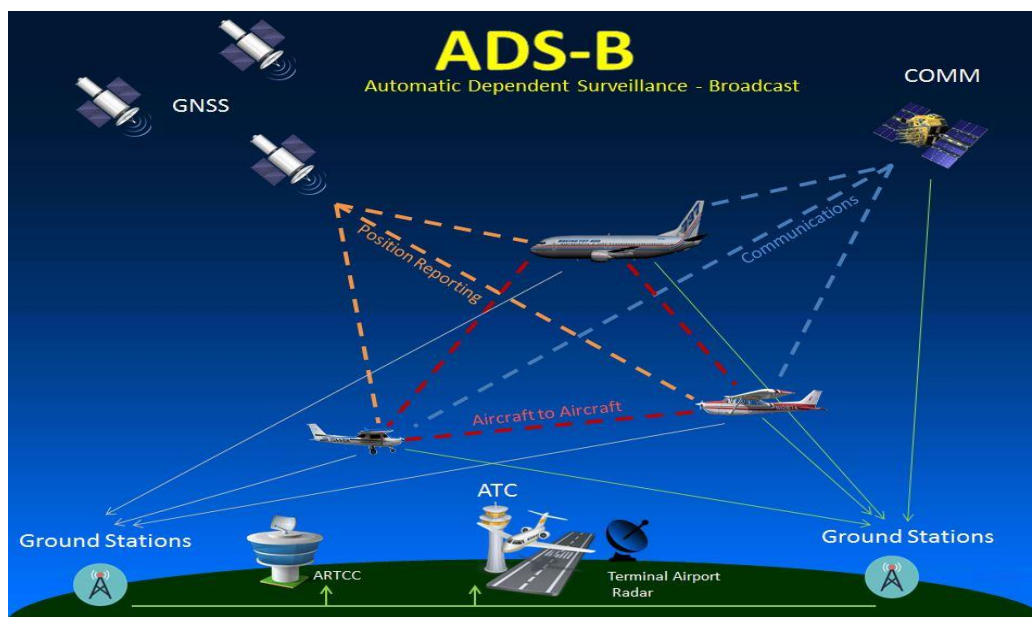


Figura 3.6: Radar ADS-B

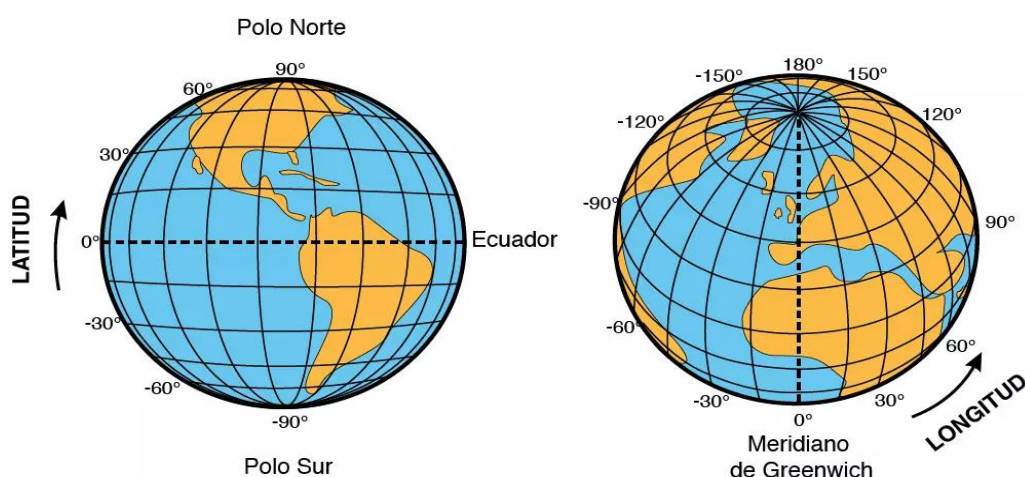
### 3.4.2. Tipos de Vigilancia Aérea

La vigilancia aérea puede ser dependiente o independiente según el lugar donde se lleve a cabo el cálculo de la posición de la aeronave. Si se emplea la vigilancia independiente, la posición se determina mediante la estación terrestre (utilizando la respuesta recibida o la señal reflejada). Ejemplos de estos sensores son los radares primarios, los secundarios y los MLAT. Con el uso de la vigilancia dependiente, la posición de la aeronave se calcula mediante el equipo a bordo (receptor GNSS, INS, etc) y luego se transmite a la estación terrestre. Un ejemplo de este principio es el ADB-S [18].

En paralelo, la vigilancia también se puede categorizar como cooperativa o no cooperativa, en función de si es necesario el involucramiento de los sistemas internos de la aeronave. Un sensor de vigilancia no cooperativa puede detectar aeronaves independientemente de su equipamiento. Un caso ilustrativo es el radar primario. Sin embargo, los otros sensores mencionados (SSR, MLAT y ADS-B) requieren cierta información de la aeronave (aunque MLAT y SSR realizan su propio cálculo de posición) [17].

Cada sensor (o conjunto de sensores en el caso de MLAT) pueden proporcionar información de posición de dos posibles formas:

- Rango y azimut: Se proveen las coordenadas de la detección con la distancia y rumbo relativo desde el sensor. Conociendo la posición del sensor se pueden proyectar estas coordenadas a otros sistemas.
- Coordenadas Geodésicas (Latitud y Longitud): El sistema específico que se usa generalmente es WGS84. Este sistema se trata de un patrón matemático conformado por tres dimensiones, la representación de la tierra se realiza por medio de un elipsoide, un cuerpo geométrico más regular que la Tierra. Se emplea en MLAT o ADS-B



**Figura 3.7: Latitud y Longitud**

Dos parámetros a destacar en los sensores son la precisión de la posición y la frecuencia de refresco. La posición de la precisión vendrá determinada por las siguientes características:

- Tipo de sensor: La precisión de un rango está determinada por la longitud del pulso y la velocidad de la luz. Por tanto, un pulso más corto proporcionará una mayor precisión en la determinación de la distancia.
- Tecnología del sensor empleado: La propia tecnología del sensor empleado afectará en la precisión de la posición. Por ejemplo, un SSR monopolso (MSSR) es más preciso en la determinación de la posición angular que el SSR clásico.
- Posición relativa de la aeronave y del sensor: La determinación del ángulo por los radares es proporcional a la distancia, es decir, los azimuts de los objetivos que están más cerca del radar son más precisos. En el entorno MLAT la posición relativa resulta muy importante. La precisión es mayor cuando la aeronave está en el centro del conjunto y disminuye si todos los sensores están en un lado de la aeronave. Una excepción que encontramos es el ADS-B, que es un ejemplo de vigilancia dependiente.

En cuanto a la frecuencia de refresco, esta determinará con que velocidad se actualiza la imagen en la pantalla de situación. Con los radares, esto está determinado por la velocidad de rotación de la antena, lo que significa que un radar en ruta típico actualizará cada objetivo unas 6 veces por minuto de manera aproximada (una rotación completa dura sobre 10 segundos). Un radar de aeródromo suele ser más rápido, con unas 10-12 actualizaciones por minuto.

Por la parte de ADS-B y MLAT las actualizaciones suelen ser mucho más frecuentes (actualizaciones cada segundo), pero esto también depende del entorno particular. Por ejemplo, la frecuencia de actualización de MLAT depende de la frecuencia de las consultas en el área específica (es decir, cuántos SSR y equipos de petición de consultas están involucrados). En el caso de ADS-B, la frecuencia de actualización depende únicamente del equipo a bordo.

### 3.4.3. Mejoras de los Sistemas de Vigilancia

La implementación de sistemas de vigilancia es un requisito indispensable y obligatorio para el control del tráfico aéreo, algunas de las mejoras más destacables de estos sistemas son:

- Mejora de la conciencia situacional para el controlador:
  - Las aeronaves se pueden identificar en la pantalla de situación, es decir, la posición de un símbolo puede ser correlacionada con un informe del piloto. Esto proporcionará un nivel adicional de seguridad contra errores humanos.
  - Actualización automatizada de la pantalla sin que el controlador o las tripulaciones de vuelo intervengan.
  - El uso de transpondedores proporciona información adicional (como identificación, nivel, rumbo, etc). Además, los pilotos pueden usar códigos designados para indicar una emergencia o situación anormal (por ejemplo, interferencia ilícita o pérdida de comunicación).

- El controlador puede informar si se están cumpliendo las instrucciones requeridas.
- Las aeronaves que presenten un fallo en la comunicación por radio pueden ser identificadas y su trayectoria de vuelo monitoreada.
- Reducción de las separaciones mínimas.
- Más opciones para la resolución de conflictos. Los sistemas de vigilancia permiten a los controladores resolver conflictos mediante vectorización o autorizando rutas directas.
- Menos congestión de frecuencias. En un entorno de vigilancia, la necesidad de informes de la tripulación se reduce drásticamente en comparación con el escenario de control procedimental.
- Disponibilidad de herramientas de apoyo así como sistemas de seguridad.
- Reducción de la carga de trabajo del controlador.



**Figura 3.8: Vigilancia Aérea**

#### 3.4.4. Riesgos y Desafíos en la Vigilancia Aérea

Como hemos visto previamente, el uso de sistemas de vigilancia aérea supone grandes beneficios para la industria de la aviación. Sin embargo, también presenta algunos riesgos y desafíos. Estos se deben abordar de manera adecuada para sacar el máximo rendimiento de los sistemas empleados. Algunos fallos a destacar son:

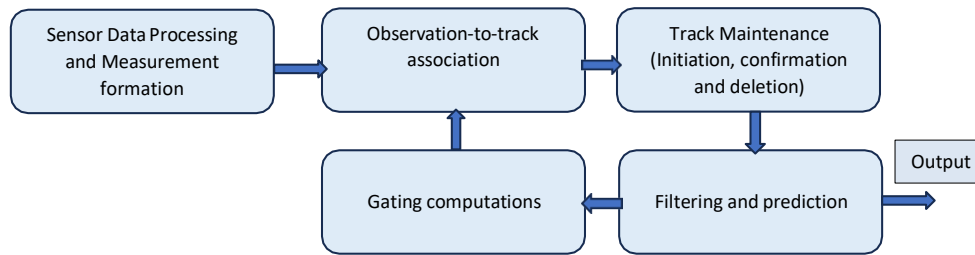
- Fallo del sensor: Si un sensor (por ejemplo, un radar) falla, es posible que algunas aeronaves ya no sean visibles en la pantalla. Para lograr mitigar este fallo se utilizan múltiples sensores con coberturas superpuestas.

- Fallo del transpondedor: La mayoría de la vigilancia se basa en el uso de transpondedores. En algunos países, no se utilizan radares primarios, lo que significa que en caso de fallo del transpondedor (o cambio no intencional al modo STANDBY), faltará información para la detección de la aeronave, ya que no se recibirán respuestas por parte de los transpondedores. Esto se puede mitigar utilizando radares primarios en adición a los sensores dependientes de la aeronave.
- Fallo del sistema ATS: Al igual que con cualquier sistema, es posible experimentar un fallo de hardware o software. Este riesgo generalmente se mitiga al contar con un sistema de respaldo separado (y a veces más de uno) que se puede activar en cuestión de segundos. Los sistemas son completamente independientes, con ordenadores y líneas de energía separadas. A veces se utilizan diferentes versiones de software o incluso sistemas de proveedores diferentes. Esto tiene la ventaja de ser más resistente a fallos de software (es menos probable que el mismo error que inutiliza un sistema haga lo mismo con un software diferente). Por otro lado, cambiar a una interfaz humano-máquina diferente (aunque sea similar) reduce el rendimiento del controlador (la ejecución de tareas rutinarias lleva más tiempo) y, por lo tanto, es posible que sea necesario ajustar las capacidades de los sectores.
- Cálculo incorrecto de la posición (por el sistema a bordo) en un entorno solo de ADS-B: En este caso, el controlador sabría que los datos de posición son incorrectos, pero sería difícil determinar la posición correcta y, en consecuencia, aplicar la separación adecuada sería problemático.
- El uso exclusivo de ADS-B como fuente de vigilancia tiene un posible problema de seguridad, ya que es técnicamente posible crear una aeronave falsa.
- Retrasos en el procesamiento de datos: El cálculo de la posición, la extracción de trazos y la determinación de trayectorias a veces requieren cálculos complejos (especialmente el cálculo de posición MLAT). Si bien existe un inevitable retraso en todos los sistemas de vigilancia, los ordenadores modernos lo han reducido a niveles insignificantes.

### 3.5. Sistemas de Detección con Redes de Sensores

Como se ha descrito en apartados anteriores, la vigilancia aérea se puede categorizar como cooperativa o no cooperativa, en función de si es necesario el involucramiento de los sistemas internos de la aeronave. Una vez un objeto es detectado, se necesita una función para determinar si representa una posible amenaza o no. Esta se divide en dos subfunciones. El Seguimiento y la Detección de Conflictos.

La arquitectura general de seguimiento mostrada en la Figura 3.9 se puede aplicar a diferentes sistemas SAA y escenarios operativos.

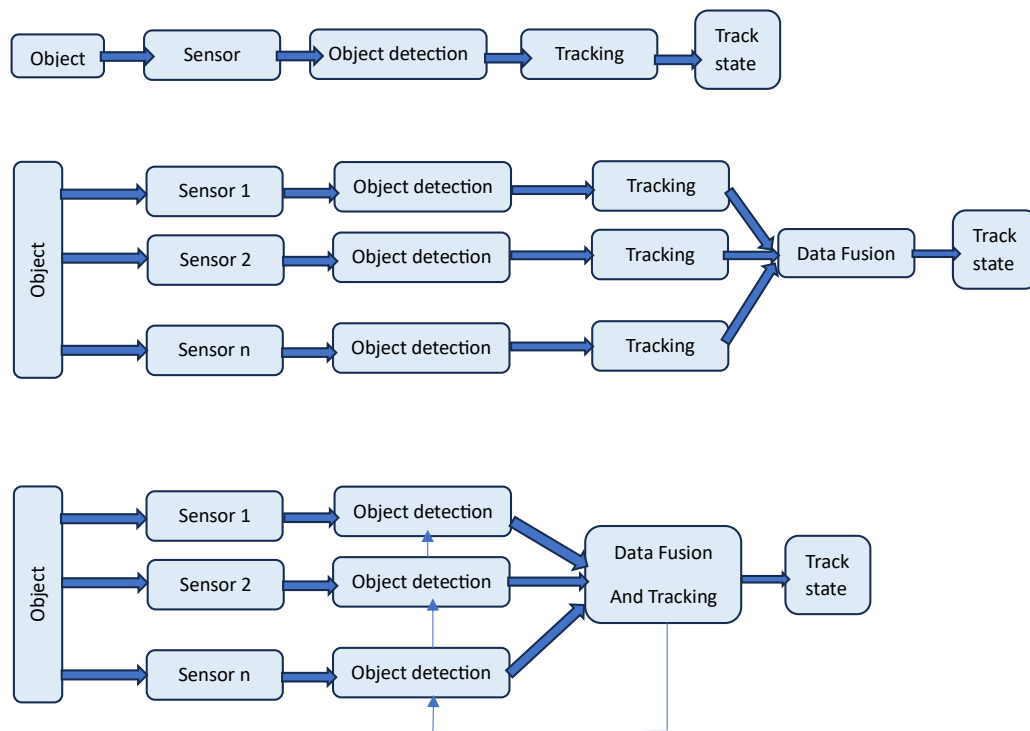


**Figura 3.9: Arquitectura de Sistemas de Seguimiento**

Si desglosamos el diagrama presentado en la Figura 3.9 tenemos los siguientes pasos:

- **Procesamiento de Datos del Sensor y Formación de Mediciones:** En esta primera etapa, los datos recibidos del sensor se procesan y transforman en mediciones significativas para el seguimiento de las aeronaves.
- **Asociación y Observación del Track:** El segundo paso que se realiza consiste en la asociación de las observaciones individuales con las trayectorias de seguimiento. Este proceso determinará que observaciones se corresponden con las aeronaves en seguimiento.
- **Mantenimiento de Trayectorias (Iniciación, Confirmación y Eliminación):** Esta tercera etapa se centra en la gestión de las trayectorias existentes. Nuevas trayectorias serán iniciadas, las existentes se confirmarán o refutarán, y las obsoletas se eliminan del proceso.
- **Filtrado y Predicción:** La 4 fase del proceso emplea técnicas de filtrado para mejorar y actualizar las estimaciones que se han realizado sobre las aeronaves en función de las observaciones y predicciones anteriores.
- **Por último, tenemos dos posibles opciones.** Por una parte, podemos pasar al Cálculo de Gating que nos ayudará a evaluar si las observaciones recientes se encuentran en un rango aceptable para cada trayectoria (nos permitirá refinar la asociación entre observaciones y trayectorias). Por otra parte, tenemos la Salida. Cuando se logre ajustar con la mayor precisión posible las observaciones tomadas, daremos por concluido el proceso realizado por el sistema de seguimiento.

En términos generales, los sistemas de detección que utilizan una red con varios sensores aumentan el rendimiento de la detección. Esta mejora se alcanza mediante la fusión de datos, que aprovecha la combinación del conjunto de sensores. Esta fusión puede ejecutarse en diferentes etapas del proceso. Se puede realizar después de haber llevado a cabo el seguimiento (Figura 3.10 b)) o en los datos brutos del sensor (Figura 3.10 c) . Para el escenario que se propone en el proyecto, haremos uso de la solución b) [19].



**Figura 3.10: Configuración de Array de Sensores**

Como hemos visto, la fusión de datos es especialmente importante en contextos donde es necesario combinar, fusionar y condensar grandes volúmenes de datos para obtener información de calidad y coherencia. Además, esta importancia se extiende a escenarios donde los procesos automatizados de integración de datos permiten la unión de mediciones y datos esenciales, generando el conocimiento necesario para formular y ejecutar decisiones de manera autónoma. La fusión de datos resulta útil en diversas áreas, como los sistemas militares, el control de procesos o la vigilancia y seguimiento civil. Los métodos de fusión de datos son particularmente importantes en la dirección hacia sistemas autónomos en todas estas aplicaciones. En el caso de la detección de UAVs, la fusión de datos podría utilizarse para mejorar el rendimiento del sistema de detección de UAVs, mejorando o aliviando los problemas y desventajas de los sensores individuales [19][20].

En cuanto a la fusión de datos, pueden surgir los siguientes problemas:

- El sensor puede no proporcionar observaciones en un intervalo de tiempo específico.
- Las observaciones pueden contener ruido.
- No se conoce de forma previa la observación generada por un objetivo específico en cada intervalo de tiempo.

Para resolver estos problemas, las técnicas de asociación de datos tienen como objetivo determinar, a partir de un conjunto de mediciones, cuáles corresponden a cada objetivo. La asociación de datos se lleva a cabo a menudo antes de la estimación del estado de los objetivos detectados y es un paso clave, ya que la estimación o clasificación se comportará de manera incorrecta si la

fase de asociación de datos no funciona de manera consistente. Sin embargo, este proceso podría aparecer en todos los niveles de fusión, pero para un sistema SAA, cobra especial importancia en el procesamiento del sensor, ya que, en términos generales, determinará el número de obstáculos que un sensor está detectando. Los siguientes algoritmos ofrecen un buen rendimiento para la tarea de asociación de datos en algunos entornos [19][21]:

- **K-means:** Este algoritmo divide los valores de un conjunto de datos en  $K$  grupos diferentes, teniendo en cuenta la similitud entre los valores. Se trata de un algoritmo iterativo que emplea el conjunto de datos y el número deseado de grupos  $K$  como entrada. El K-means es relativamente fácil de implementar, además, es capaz de encontrar una solución factible en un intervalo de tiempo reducido (no implica que sea la solución óptima).
- **Asociación Probabilística de Datos (PDA):** Este algoritmo fue propuesto para lidiar con un entorno de interferencias. Se asigna una probabilidad de asociación a cada hipótesis a partir de una medición válida. Para el cálculo de la estimación del estado del objetivo se realiza una suma ponderada del estado estimado bajo todas las hipótesis. El PDA presenta un rendimiento deficiente cuando los objetivos se cruzan o están muy próximos entre sí, también es incapaz de manejar de forma adecuada múltiples objetivos.
- **Asociación Conjunta de Datos Probabilísticos (JPDA):** El algoritmo JPDA es una versión optimizada del PDA. En este caso somos capaces de manejar múltiples objetivos de forma adecuada. La diferencia radica en que las probabilidades de asociación se calculan empleando todas las observaciones y objetivos, es decir, considera diversas hipótesis de manera conjunta y las combina.
- **Prueba de Múltiples Hipótesis (MHT):** La idea detrás del algoritmo MHT se basa en el uso de más de dos observaciones consecutivas para reducir la probabilidad de errores. Para lograrlo, el MHT estima todas las hipótesis posibles y las mantiene en cada iteración. Aunque el MHT obtiene densidades menores de falsos positivos si lo comparamos con otros algoritmos (como el JPDA) el costo computacional aumenta de manera exponencial al incrementar el número de trayectorias o mediciones. Por esta razón, la implementación práctica de este algoritmo está limitada debido a su alto costo tanto en tiempo como en memoria.
- **Algoritmos de Reconocimiento de Imagen:** Se han desarrollado modelos capaces de detectar, rastrear y reconocer de manera automática aeronaves no tripuladas mediante una única cámara. El proceso que se realiza consiste en detectar objetos en vuelo y obtener sus trayectorias mediante el uso de la cámara. Para clasificar las imágenes que se han obtenido se empleará un clasificador de redes neuronales convolucionales, entrenado previamente con un conjunto de imágenes.

Para el escenario de operación que hemos realizado, se ha desarrollado mediante la aplicación de cómputo numérico Matlab un algoritmo JPDA para el seguimiento y asociación de los datos.



### 3.6. Modelo de Error del Radar Empleado

Para poder establecer un funcionamiento óptimo del radar descrito en la metodología del proyecto debemos tener en cuenta los siguientes errores [22][23]:

- **Error de Rango:** El error de rango está directamente relacionado con la precisión del radar en la medida de distancia. Por tanto, representará el grado de conformidad entre la posición estimada de una aeronave en un momento dado y su verdadera posición. Se trata de un ruido gaussiano con media 0.
- **Error de Ganancia:** El error de ganancia nos indica la discrepancia existente entre la ganancia real de la antena del radar y la ganancia esperada o nominal.
- **Error de Azimut:** El error de azimut consiste en una combinación de dos errores. Tenemos el error de medición de azimut y el sesgo de azimut del radar. El error de medición de azimut hace referencia al error en la medida de la posición azimutal de la aeronave en relación con el haz del radar. Por otra parte, el sesgo de azimut del radar es un error causado por el desalineamiento del radar con el norte.
- **Error de Altura:** El error de altura de un radar se refiere a la inexactitud en la medida de altitud de un objetivo por parte del radar.

## Capítulo 4

# Desarrollo y Resultados

En este capítulo, nos vamos a centrar en el código de Matlab desarrollado para cumplir los requerimientos establecidos. Nos enfocaremos en las tres principales partes: La generación de los datos de vuelo, el filtrado de Kalman para las medidas generadas y el uso del filtro JPDA para la asociación de las diferentes trayectorias de vuelo. En todos los casos expondremos tanto los inputs y outputs que presentan estas funciones como el modelo de programación que se ha seguido. Por último, presentaremos diversos resultados para comprobar la eficacia del código generado así como su posible implementación en un entorno de vuelo real.

### 4.1. Generador de Trayectorias

Esta función se encarga de generar  $N$  trayectorias en línea recta con movimiento lineal uniforme en un plano cartesiano. Somos capaces de aproximar la esfera terrestre como un plano cartesiano ya que las distancias son reducidas en relación a la curvatura de la tierra, por tanto, los errores de aproximación son mínimos. Como variables de entrada a la función tenemos las siguientes:

- **samples**: Número de muestras por trayectoria.
- **N**: Número de trayectorias que queremos generar.
- **conf**: Matriz que contiene información de cada trayectoria. Cada columna representa una trayectoria y contiene los siguientes datos:
  - origen
  - altitud
  - rumbo (curso)
  - velocidad en tierra (GS)
  - tiempo inicial ( $t_0$ )
  - transponder para ADS-B (transponderUL)
- **nRadar**: Número de sensores de radar.
- **tRadar**: Periodos para los radares.

- **rLoc**: Matriz de posiciones de los radares en el plano cartesiano.
- **rBias**: Matriz de sesgos para los radares.
- **rNoise**: Matriz de ruido para los radares.
- **nADSB**: Número de sensores ADS-B.
- **tADSB**: Periodos para los sensores ADS-B.
- **aBias**: Matriz de sesgos para los ADS-B.
- **aNoise**: Matriz de ruido para los ADS-B.

Para comprender los pasos que realiza la función, podemos establecer las siguientes etapas:

- 1. Parámetros de Entrada: Definimos los valores para todas las variables de entrada de la función.
- 2. Generación de Trayectorias: El código recorrerá las **N** trayectorias que se deben generar. Para cada trayectoria se extraerán los datos de configuración de la matriz **conf**. Posteriormente, mediante el uso de la función externa **trajgenerator** se generará la trayectoria de vuelo ideal en función de los parámetros proporcionados. Los valores de estas trayectorias se almacenarán en la variable **trajectories**.
- 3. Generación de Detecciones de Radar: Para cada uno de los **nRadar**, en nuestro caso particular 2, el código generará detecciones simuladas mediante el uso de la función externa **addRADARerror**. Esta función toma la trayectoria ideal que se ha generado en el paso previo junto con otros parámetros específicos del radar, mediante estos valores genera datos simulados de rango, azimut y altitud. Estos datos se convierten en coordenadas cartesianas y se almacenan en la variable **trajectories** en dos posiciones diferentes: una para las detecciones de rango, azimut y altitud, y otra para las detecciones en coordenadas cartesianas.
- 4. Generación de detecciones ADS-B: Para cada sensor ADS-B establecido (**nADSB**), el código usará la función externa **addADSBError** para generar detecciones simuladas de posición en función de la trayectoria ideal y los parámetros de los sensores ADS-B. Las detecciones generadas se almacenarán en la variable **trajectories**.
- 5. Salida: Tras concluir todos los pasos citados, la función devolverá una celda **trajectories** con dimensiones  $(1+2*\mathbf{nRadar}+\mathbf{nADSB}) \times N$ , donde cada columna representa una trayectoria. Los datos que se almacenan en cada columna incluyen la trayectoria ideal, las detecciones de radar y las detecciones ADS-B.

Para tener una visión más clara y esquemática de la función encargada de generar las trayectorias, entendiendo las etapas descritas así como el proceso que realiza el código, se ha diseñado un diagrama de flujo que se muestra en la Figura 4.1.

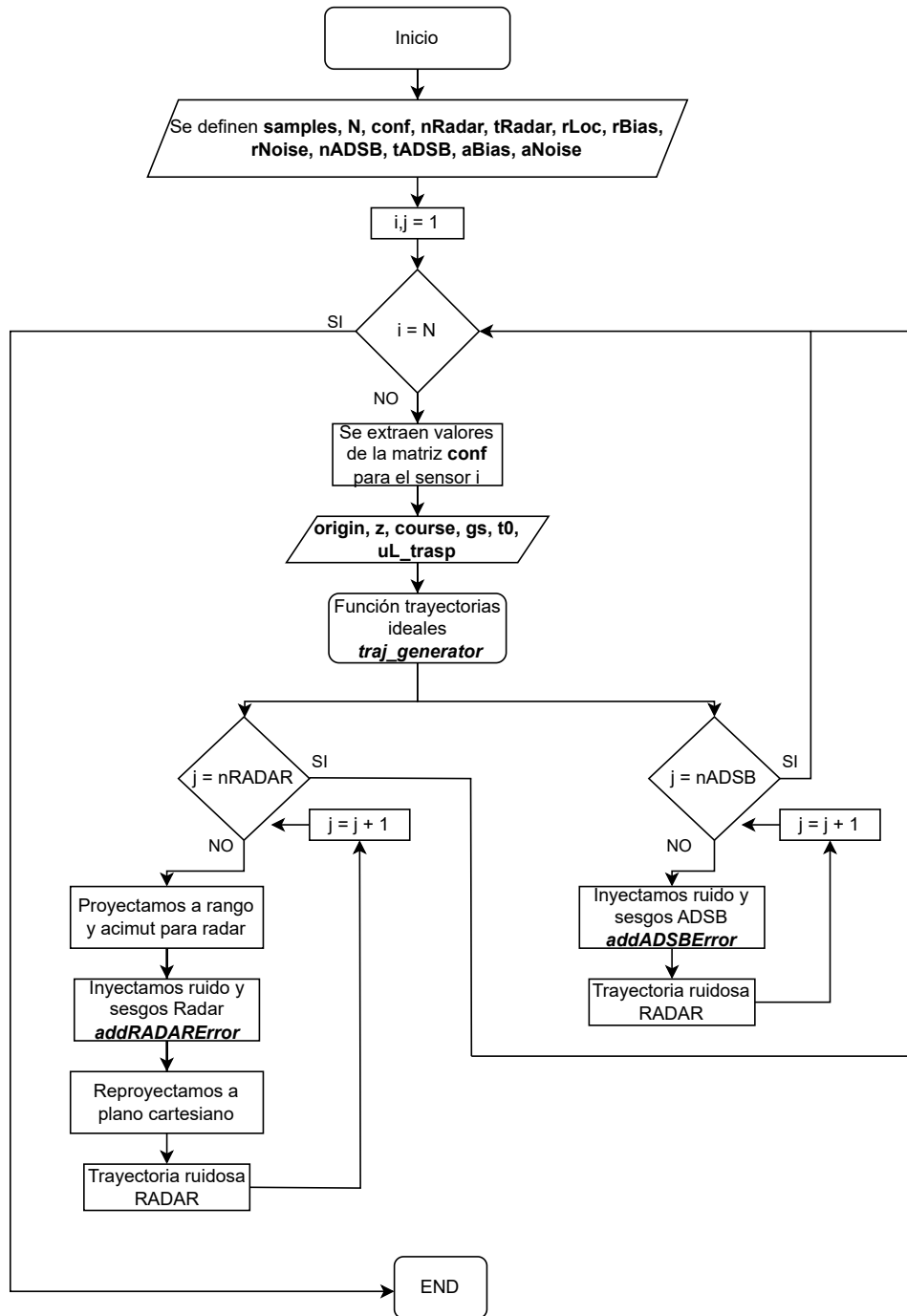


Figura 4.1: Diagrama de Flujo Generador de Trayectorias

### 4.1.1. Ejemplo Trayectorias Generadas

Tras haber visto el funcionamiento del código encargado de generar las trayectorias, vamos a mostrar un ejemplo práctico en el que se representan las trayectorias ideales y las trayectorias tras la inyección de sesgos y ruido. Teniendo así unas detecciones realistas basadas en los ruidos y sesgos definidos de forma previa, representado las medidas que tomarían los radares.

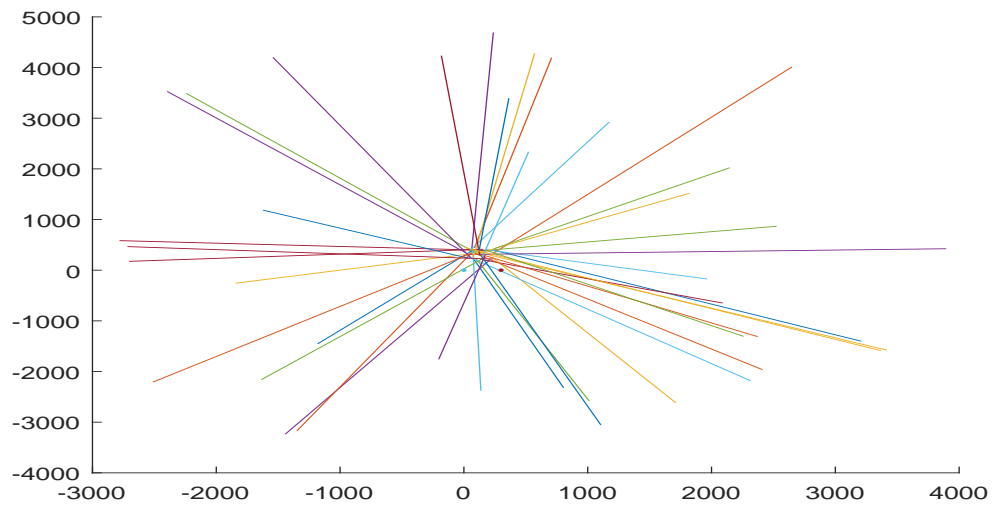
En la Figura 4.2 podemos observar la variable cell que contiene las trayectorias (**trajectories**). Para este primer ejemplo, hemos definido 50 trayectorias (**N=50**) con 70 muestras por trayectoria (**samples=70**). También se ha establecido la posición de los radares (en el plano x separados 300 metros entre sí) mediante el parámetro **RADAR.rLoc(2,2)=300** y el ángulo de visión de los radares, siendo este de 60°. Al definir estos valores, obtenemos una variable cell 6x50. Donde cada fila representa los siguientes valores:

- 1: Trayectorias ideales.
- 2: Detecciones del primer Radar con valores de Rango, Azimut, Altura y Tiempo de Detección.
- 3: Detecciones del primer Radar con valores de posiciones x e y, Altura y Tiempo de Detección.
- 4: Detecciones del segundo Radar con valores de Rango, Azimut, Altura y Tiempo de Detección.
- 5: Detecciones del segundo Radar con valores de posiciones x e y, Altura y Tiempo de Detección.
- 6: Detecciones para el sensor ADS-B.

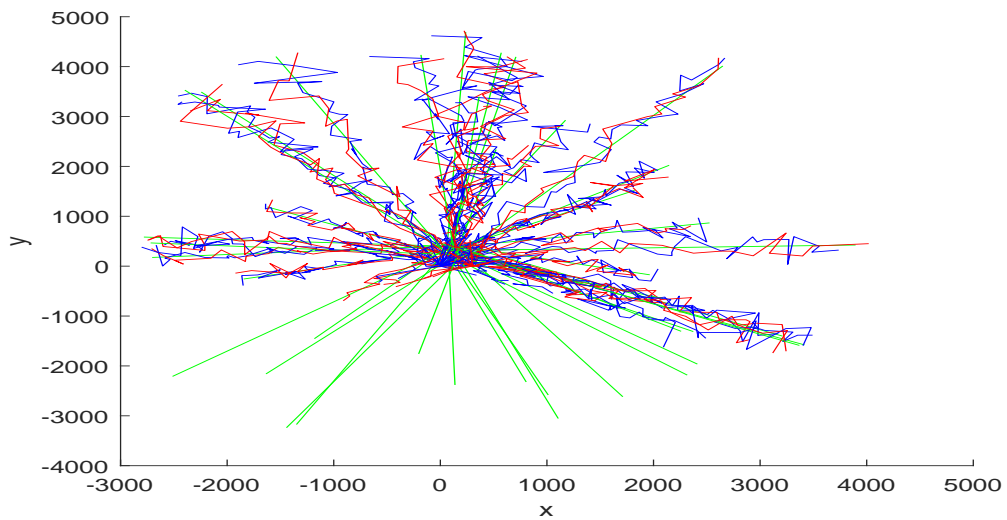
	1	2	3	4	5	6	7	8	9	10	11
1	70x5 double	70x5 double	70x5 double	70x5 double	70x5 double	70x5 double	70x5 double	70x5 double	70x5 double	70x5 double	70x5 double
2	10x4 double	10x4 double	10x4 double	[148.7908,3...	9x4 double	10x4 double	[]	[]	[]	[]	[]
3	10x4 double	10x4 double	10x4 double	[62.6351,97...	9x4 double	10x4 double	[]	[]	[]	[]	[]
4	10x4 double	[]	9x4 double	[]	5x4 double	7x4 double	[]	[]	[]	[]	[]
5	10x4 double	[]	9x4 double	[]	5x4 double	7x4 double	[]	[]	[]	[]	[]
6	69x3 double	69x3 double	69x3 double	69x3 double	69x3 double	69x3 double	69x3 double	69x3 double	69x3 double	69x3 double	69x3 double

**Figura 4.2: Variable Trajectories**

En las Figuras 4.3 y 4.4 se muestran las trayectorias ideales que se han generado y los valores resultantes tras inyectar el ruido. Además, en las trayectorias ideales podemos observar la ubicación de ambos radares en la posición 0 y 300 del plano x, como se ha establecido previamente. Por otra parte, para generar las trayectorias ruidosas se han definido los siguientes errores: 1. Error de Azimut de 3° (indicado en el datasheet del fabricante), 2. Error de Altura de 150 metros, 3. El Error de Rango indicado en el datasheet por el fabricante va de 0.6-306 metros, para nuestro entorno asumimos un error de rango inferior al 10 % de la distancia máxima de medida (30 metros).



**Figura 4.3: Trayectorias Ideales**



**Figura 4.4: Trayectorias Simuladas con Ruido**

## 4.2. Filtro de Kalman

Una vez hemos generado las trayectorias ideales y les hemos aplicado los segos y el ruido establecido, vamos a proceder a filtrar estas trayectorias. Para ello, se ha desarrollado un filtro de Kalman de cuatro estados, con el que intentaremos lograr valores próximos a las medidas ideales. En primer lugar, vamos a definir las variables de entrada que se van a utilizar.

- **X**: Matriz con las posiciones  $x$  e  $y$  de todas las trayectorias.
- **t**: Escala de tiempos de las medidas.
- **R**: Matriz de covarianza de  $X$ .
- **sa**: Desviación estándar.
- **H**: Matriz de observación.
- **Pp**: Matriz de covarianza de la predicción.
- **k**: Ganancia de Kalman.
- **Q**: Matriz de covarianza del ruido.
- **r**: Matriz de residuos.
- **Xa**: Matriz de salida con todas las estimaciones generadas.
- **MSEfil**: Error cuadrático medio para las trayectorias filtradas.
- **MSEruid**: Error cuadrático medio para las trayectorias ruidosas.

Para comprender los pasos que se realizan hasta conseguir los valores filtrados, se definen las siguientes etapas:

- 1. Inicialización: Cargamos los datos de las trayectorias y establecemos los parámetros iniciales del filtro, como la desviación estándar **sa** y el número de mediciones **H**. El siguiente paso será inicializar el bucle que recorrerá todas las trayectorias que vamos a filtrar.
- 2. Preparación de los Datos: Definimos la variable **radar1data**, que hace referencia a las trayectorias almacenadas en la tercera fila de **trajectories**. Seleccionamos esta fila ya que es donde tenemos las medidas que vamos a filtrar, pertenecientes al primer radar. El siguiente paso será comprobar si para la medida que estamos realizando tenemos valores de trayectorias (puede ocurrir que para alguna medida no se generen trayectorias). Si tenemos medidas, vamos a recopilar los valores de posición y de tiempo. Estos se almacenarán en **X** y **t** respectivamente. Para las siguientes iteraciones concatenaremos los valores de **X** y los de **t**, teniendo así todas las posiciones e instantes de tiempo. Si por el contrario no tenemos medidas de trayectorias, pasaremos a la siguiente iteración.
- 3. Cálculo de la Matriz de Covarianza: A partir de las posiciones almacenadas en **X** calculamos la matriz de covarianza. Esta matriz representa el error de medición.

- 4. Interpolación: Interpolamos las trayectoria ideal para que las medidas tengan el mismo intervalo de tiempo que las medidas reales. Este paso nos va a permitir comparar las medidas filtradas con las ideales.
- 5. Filtrado de Kalman: Realizamos el proceso de filtrado para cada medición de la trayectoria. En cada iteración se realizarán los siguiente pasos:
  - Predicción: Calculamos la predicción del siguiente estado usando el modelo dinámico del sistema. A continuación, obtenemos la matriz de covarianza de la predicción  $P_p$ .
  - Observación: Calculamos la matriz de innovación  $s$ , que representa la suma de incertidumbres del estado estimado y de las mediciones. Posteriormente, realizaremos el cálculo de la ganancia de Kalman  $k$ .
  - Actualización del Estado y Covarianza: Se calcula el residuo entre la medición real y la predicción. Mediante la ganancia de Kalman coregimos el estado estimado y la matriz de covarianza.
- 6. Interpolación tras el Filtrado: Una vez se ha realizado el filtrado, los datos de salida se almacenan en  $X_a$ , esta variable es una matriz con todas las estimaciones, tenemos la posición en  $y$ , la velocidad en  $y$ , la posición en  $x$  y la velocidad en  $x$ . Posteriormente, realizamos la interpolación de la trayectoria filtrada para que coincida con las mediciones reales.
- 7. Cálculo de Errores: Una vez tenemos las tres medidas de interés (trayectorias ideales, trayectorias con ruido y trayectorias filtradas) vamos a calcular el error cuadrático medio (RMSE) para cada trayectoria. El primer paso para realizar el cálculo de los errores, será reordenar todas las posiciones (ideales, ruidosas y filtradas) eliminando las medidas que no tengan valores. El siguiente paso será calcular la diferencia de valores entre las trayectorias ideales y las ruidosas y entre las trayectorias ideales y las filtradas. Para ello haremos uso de la función **Cellfun** de Matlab. Por último, obtendremos los valores de RMSE para las medidas ruidosas y para las filtradas. A partir de estos valores, seremos capaces de ver la desviación que sufren los valores de las trayectorias ideales tras aplicarles ruido y tras realizar su filtrado.
- 8. En el último paso representaremos las tres trayectorias (Ideales, Ruidosas y Filtradas) en una misma gráfica, así tendremos una imagen de las medidas en su conjunto.

Para tener una visión más clara y esquemática del proceso de filtrado de las trayectorias ruidosas, entendiendo las etapas descritas así como el proceso que realiza el código, se ha diseñado un diagrama de flujo que se muestra en la Figura 4.5.



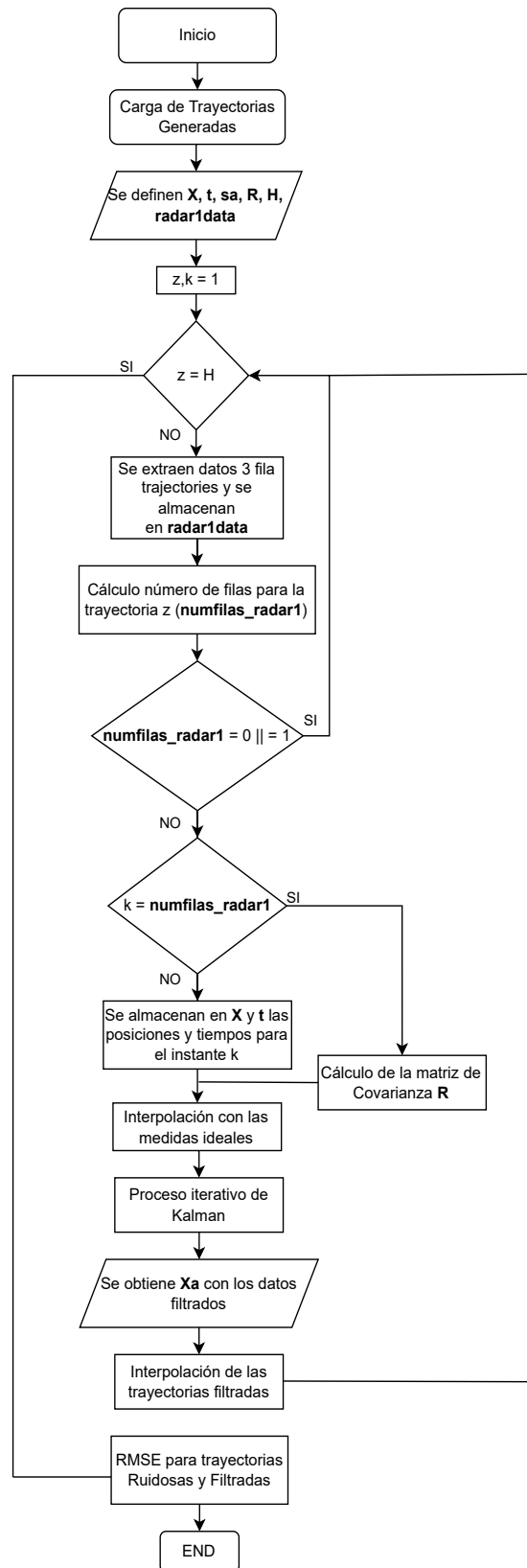
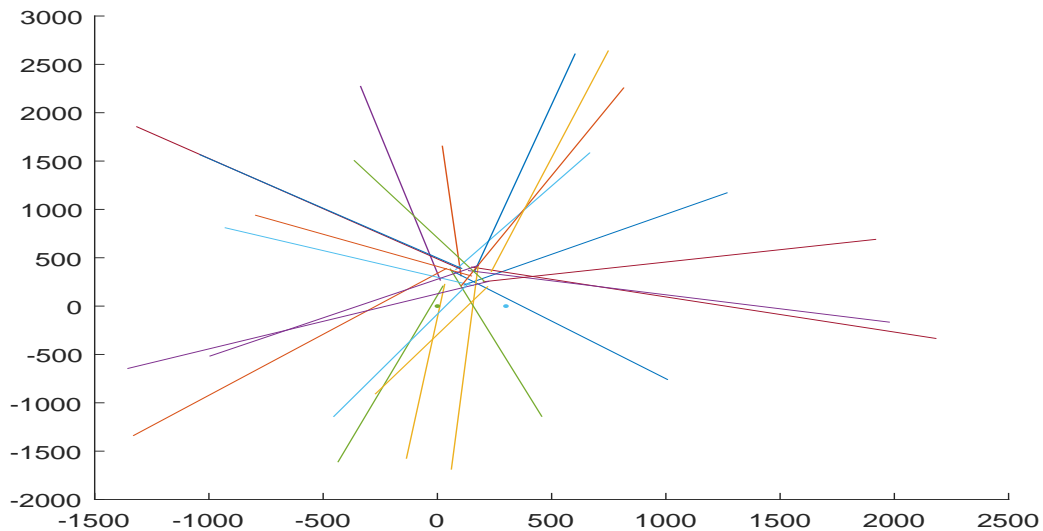


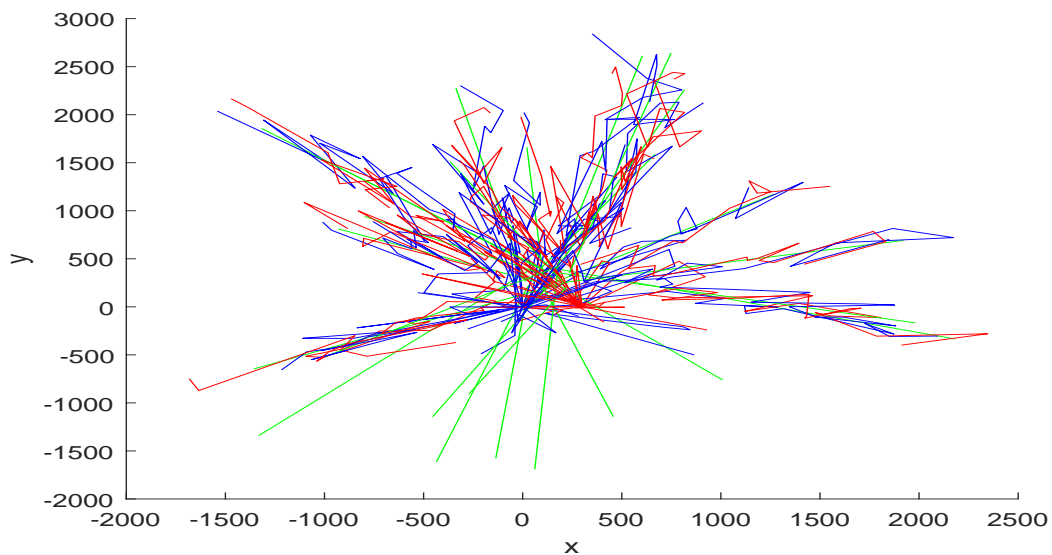
Figura 4.5: Diagrama de Flujo Filtrado de Kalman

### 4.2.1. Ejemplo Filtrado de Kalman

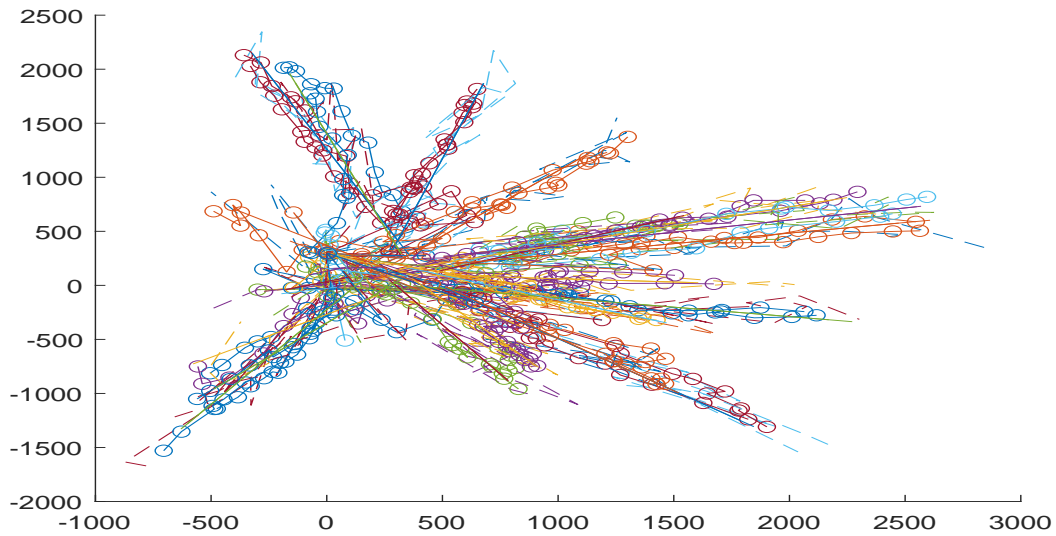
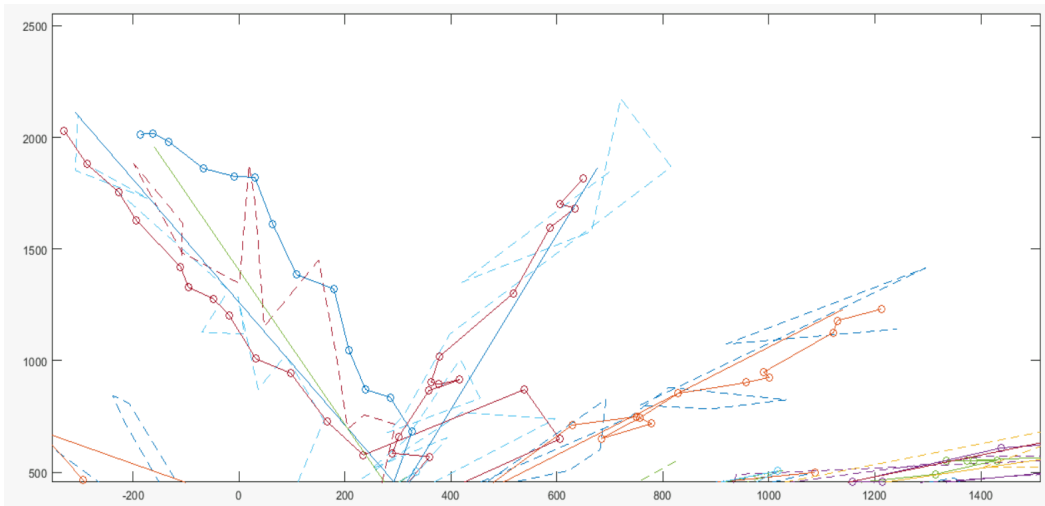
Después de observar el funcionamiento del código empleado para realizar el filtrado, se va a presentar un caso concreto del proceso realizado. Para el ejemplo realizado, se establecen 25 trayectorias ( $N=25$ ) con 100 muestras por trayectoria (**samples=100**). En primer lugar, las gráficas 4.6 y 4.7 representan las trayectorias ideales generadas y los valores de trayectoria después de inyectar ruido.



**Figura 4.6: Trayectorias Ideales Generadas**



**Figura 4.7: Trayectorias Obtenidas con Ruido**

**Figura 4.8: Trayectorias Filtradas****Figura 4.9: Zoom Trayectorias Filtradas**

En la gráfica 4.8 vemos el conjunto de trayectorias ideales, ruidosas y filtradas superpuestas entre sí, en 4.9 hemos hecho zoom sobre un conjunto de trayectorias para ver de forma más nítida los resultados. Se observa que cada trayectoria está formada por 3 líneas diferentes. En primer lugar, la línea continua representa la medida ideal que se ha generado. En segundo lugar, tenemos la línea discontinua que representa la trayectoria ruidosa. Por último, la línea compuesta por círculos representa el valor de la trayectoria filtrada. Teniendo en cuenta esta superposición, podemos concluir que con el paso de iteraciones el proceso de filtrado se asemeja en mayor medida a los valores ideales que se han generado de forma previa.

Por otra parte, para el ejemplo mostrado hemos obtenido un valor de error cuadrático medio de 225 metros para las trayectorias con ruido y de 168 metros para las trayectorias filtradas. Obteniendo una mejora del 25 % con el proceso de filtrado.

### 4.3. Filtro JPDA

Esta última parte del capítulo se centra en el desarrollo de un filtro JPDA para la asociación de las diferentes trayectorias de vuelo. El código propuesto cuenta con las siguientes variables de entrada:

- **xJPDA**: Matriz con las coordenadas x e y de las trayectorias.
- **tJPDA**: Matriz con los instantes de tiempo de las detecciones.
- **H**: Número de trayectorias que se van a generar.
- **CellList**: Variable cell para almacenar las detecciones.

Tras haber recopilado las variables de entrada que vamos a utilizar, se establecen los siguientes pasos para explicar el funcionamiento del código:

- 1. Definición de los Parámetros de Entrada: Cargamos los datos de las trayectorias e inicializamos las variables de entrada. El siguiente paso será inicializar el bucle que recorrerá todas las trayectorias generadas.
- 2. Preparación de los Datos: El segundo paso a relizar es idéntico al realizado en el código del filtrado de Kalman. Para ello definimos la variable **radar1data**, que hace referencia a las trayectorias almacenadas en la tercera fila de **trajectories**. Como establecimos previamente, seleccionamos esta fila ya que es donde tenemos las medidas que vamos a filtrar, pertenecientes al primer radar. El siguiente paso será comprobar si para la medida que estamos realizando tenemos valores de trayectorias (puede ocurrir que para alguna medida no se generen trayectorias). Si tenemos medidas, vamos a recopilar los valores de posición y de tiempo. A diferencia del filtrado de Kalman, estos se almacenarán en **xJPDA** y **tJPDA** respectivamente. Para las siguientes iteraciones concatenaremos los valores de X y los de t, teniendo así todas las posiciones e instantes de tiempo. Si por el contrario no tenemos medidas de trayectorias, pasaremos a la siguiente iteración.
- 3. Reorganización de los datos: Al finalizar el recorrido de todas las trayectorias generadas, vamos a proceder a reordenar las variables **xJPDA** y **tJPDA** según el instante de tiempo de la medida.
- 4. Generación de las Detecciones: Después del paso 3, contamos con todas las posiciones ordenadas por tiempo. Para cada posición crearemos la detección en x e y para un instante t mediante **objectDetection**. Los valores de detección generados se irán almacenando en la variable **CellList**.
- 5. Configuración de los Parámetros de Seguimiento: Se crea un objeto tracker de tipo **trackerJPDA**. En el objeto se inicializa un filtro de Kalman de velocidad constante (**initcvkf**). Además, se define el objeto **trackP** para trazar las trayectorias y **detectionP** para trazar las detecciones.
- 6. Seguimiento de las Trayectorias: Por último, hacemos uso del objeto **tracker** para confirmar las trayectorias según el instante de tiempo actual y las detecciones medidas en ese instante. Si se confirma una trayectoria, actualizaremos la posición, velocidad y covarianza.

Para recopilar el preproceso del filtro JPDA para lograr la asociación de las trayectorias, se ha diseñado el siguiente diagrama de flujo:

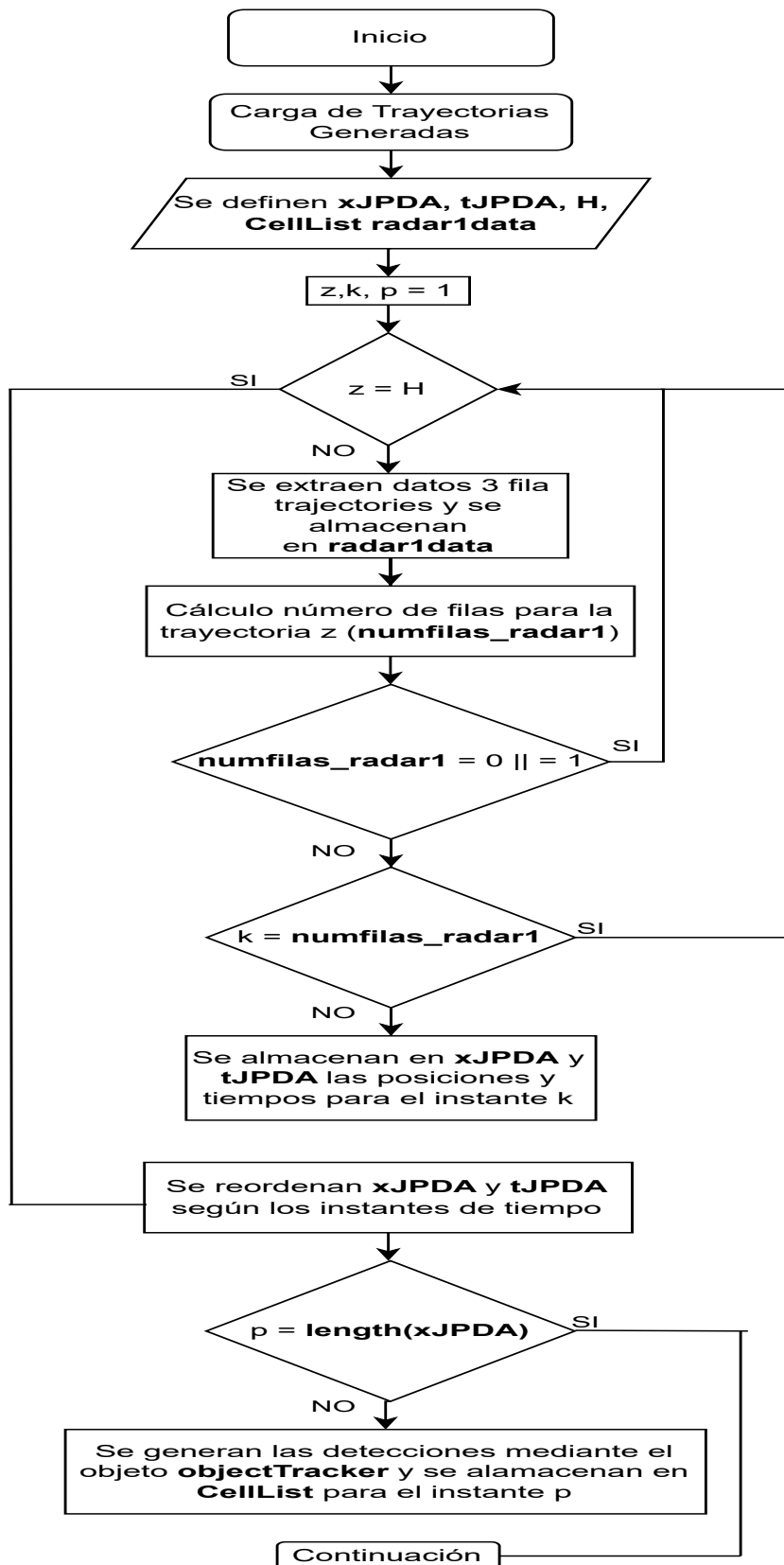


Figura 4.10: Diagrama de Flujo JPDA

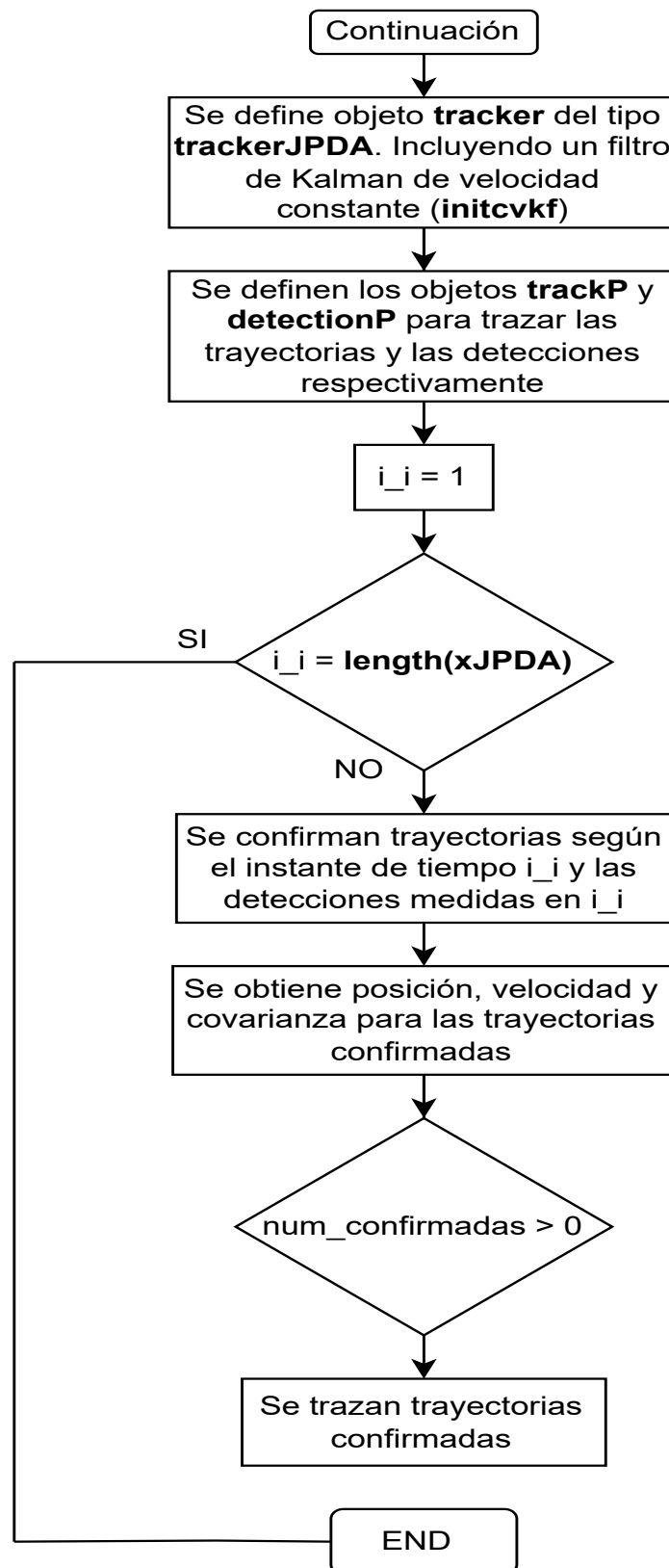
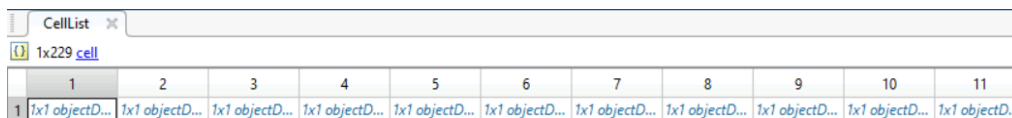


Figura 4.11: Continuación Diagrama de Flujo JPDA

### 4.3.1. Asociación de Trayectorias Mediante JPDA

Tras haber observado el funcionamiento del código empleado para realizar el proceso de asociación de trayectorias, se va a presentar un caso concreto. En el ejemplo realizado se establecen 25 trayectorias ( $N=25$ ) con 100 muestras por trayectoria (**samples**=100). En primer lugar, las gráficas 4.6 y 4.7 representan las trayectorias ideales generadas y los valores de trayectoria después de inyectar ruido.

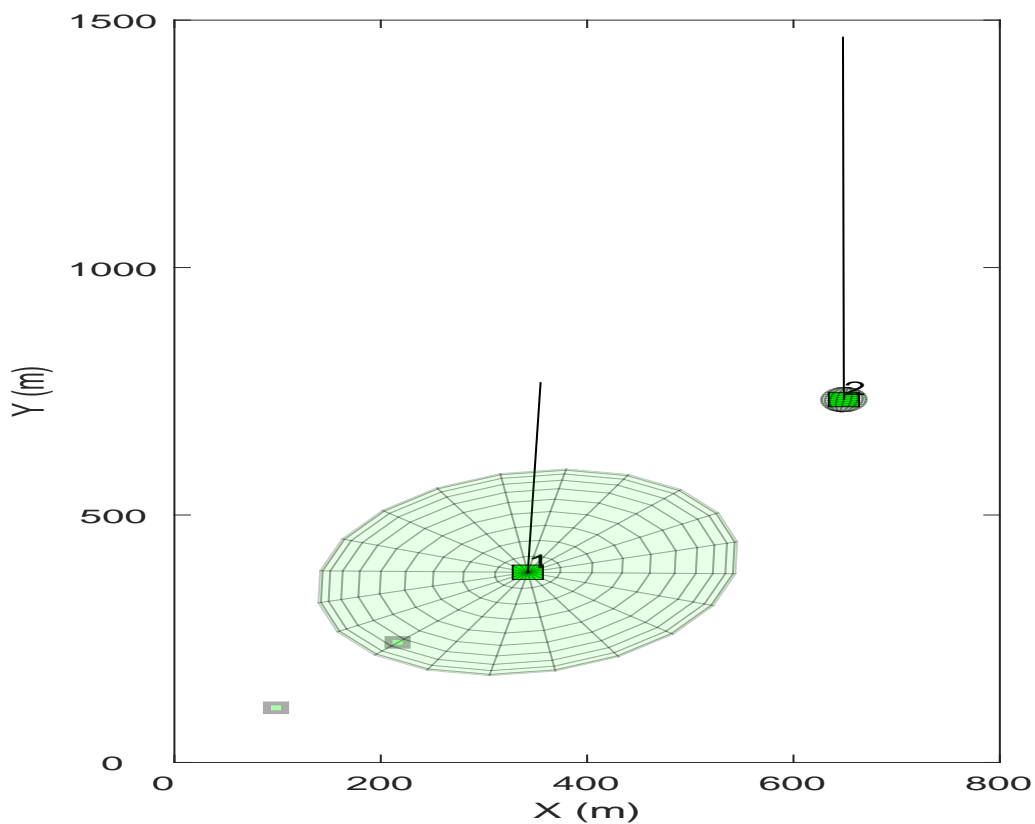
En la Figura 4.12 vemos la estructura de la variable **CellList** que se ha generado. Se trata de una variable 1x229, lo que significa que tenemos 229 posiciones en x e y (ordenadas por tiempo).



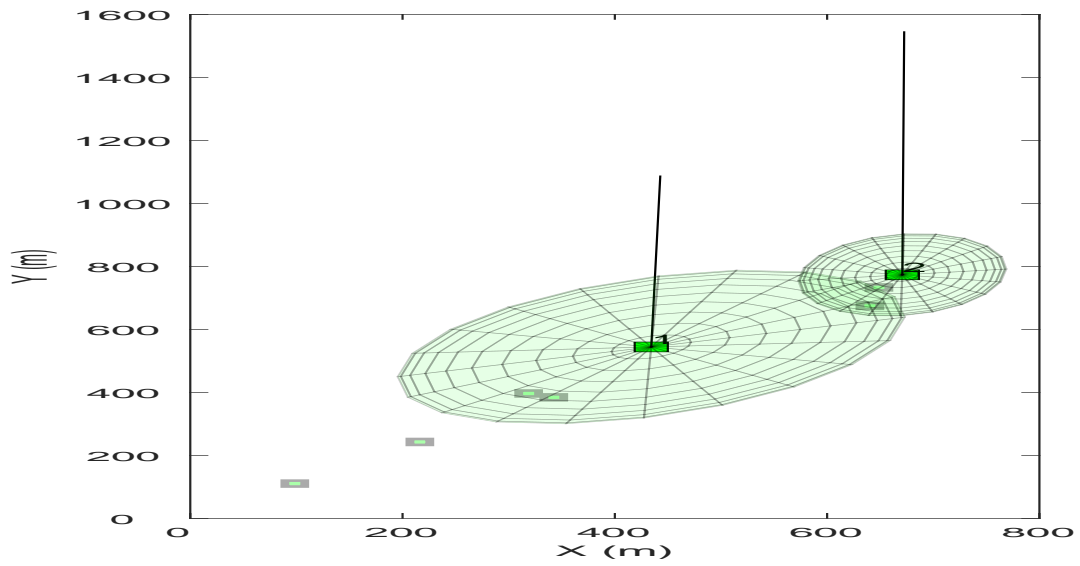
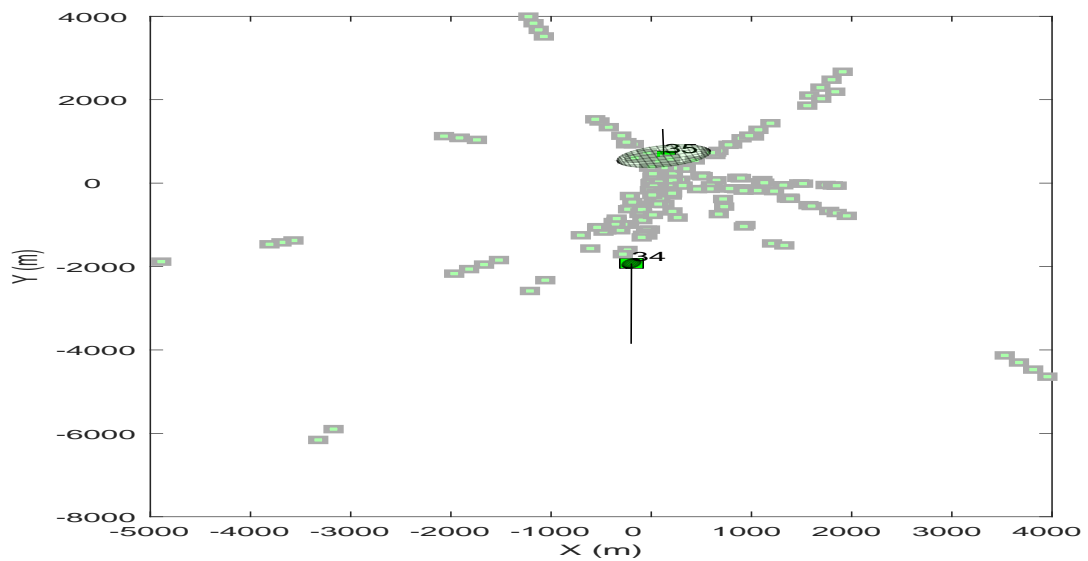
	1	2	3	4	5	6	7	8	9	10	11
1	1x1 objectD...	1x1 objectD...	1x1 objectD...	1x1 objectD...	1x1 objectD...	1x1 objectD...	1x1 objectD...	1x1 objectD...	1x1 objectD...	1x1 objectD...	1x1 objectD...

**Figura 4.12: Variable CellList**

En cuanto al proceso de asociación que realiza el JPDA, las Figuras 4.13, 4.14 y 4.15 representan distintas iteraciones del proceso.

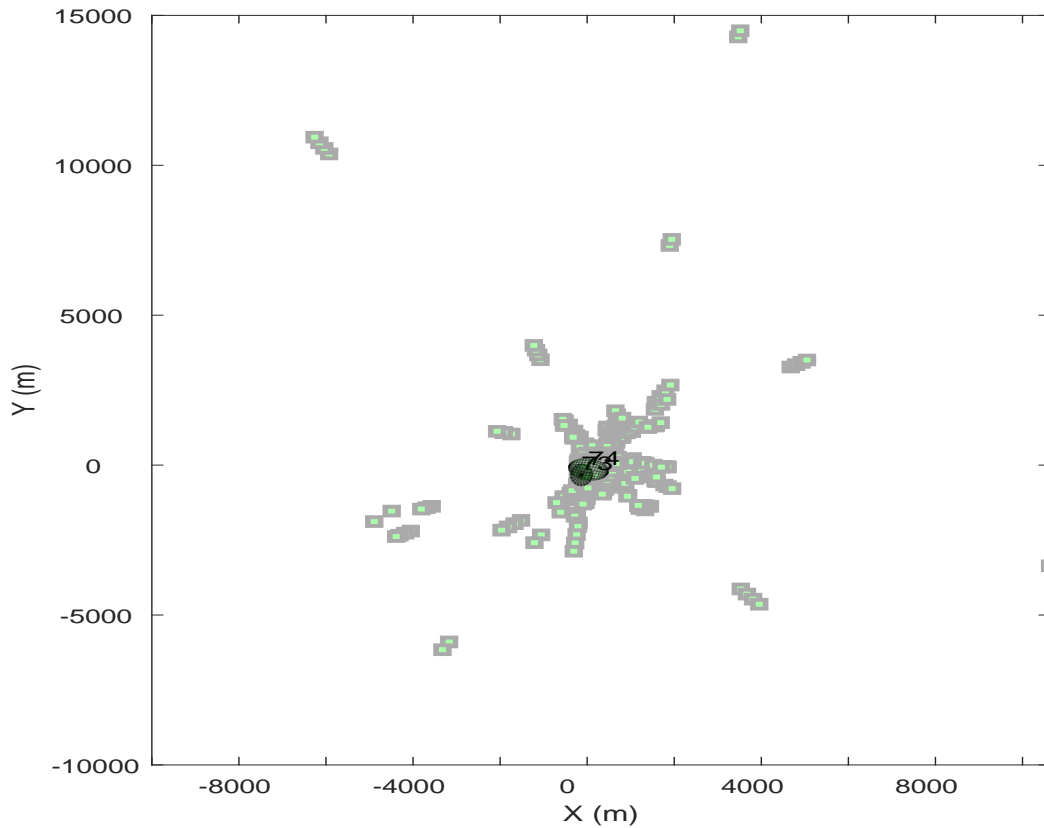


**Figura 4.13: Primera Iteración JPDA**

**Figura 4.14: Segunda Iteración JPDA****Figura 4.15: Tercera Iteración JPDA**

En las iteraciones del proceso de asociación podemos observar como se van representando las distintas detecciones generadas. Para cada detección se evalúa si esta pertenece a una trayectoria establecida anteriormente, en caso contrario se establece una nueva. Por último, la Figura 4.16 muestra el resultado final del proceso de asociación realizado.





**Figura 4.16: última Iteración de Trayectorias Confirmadas**

Para poder obtener el número de trayectorias confirmadas en cada iteración empleamos la variable **ainfo**. Se trata de una variable struct compuesta por 9 campos con información relativa al estado del sistema. En el ejemplo mostrado en la Figura 4.17 vemos el campo **UnassignedTracks** formado por dos valores. Esto indica las trayectorias que el filtro considera que han finalizado en el último paso (2 en este caso).

Field	Value
OOSMDetectionIndices	[]
TrackIDsAtStepBeginning	[124,126]
UnassignedTracks	[124,126]
UnassignedDetections	1
CostMatrix	[48,8430;93,1460]
Clusters	1x0 cell
InitializedTrackIDs	127
DeletedTrackIDs	[]
TrackIDsAtStepEnd	[124,126,127]

**Figura 4.17: Variable ainfo**

## Capítulo 5

# Conclusiones y Trabajo Futuro

### 5.1. Conclusiones

Después de finalizar el proyecto, contamos con una visión general del trabajo que se ha realizado. Recopilando la información expuesta y los resultados obtenidos, podemos establecer las conclusiones del trabajo realizado.

Cabe a destacar el cumplimiento del objetivo principal del proyecto (integración de dos radares FM-CW-sR-1200e operando a 24 GHz).

Para poder lograr este objetivo, se han ido alcanzando los objetivos específicos propuestos (generación de trayectorias de vuelo similares a una situación real, configuración de los parámetros del radar, asociación de los trackings para ambos radares, etc).

La solución propuesta cuenta con un gran potencial para brindar mejoras significativas en la capacidad de detección de aplicaciones relacionadas con aeronaves no tripuladas, como la seguridad y la vigilancia.

### 5.2. Trabajo Futuro

Tras haber recopilado las conclusiones extraídas del proyecto, el siguiente paso se centra en la presentación de las futuras líneas de trabajo. Vamos a destacar los siguientes puntos:

- 1. En primer lugar, la acción inmediata más importante consiste en el uso de un entorno real de vuelo. Como se ha visto durante todo el proyecto, las trayectorias de partida han sido generadas mediante funciones desarrolladas en Matlab. Mediante el uso de estas trayectorias hemos sido capaces de establecer un punto de partida y obtener una estimación del modelo propuesto. Sin embargo, para poder cerciorarnos del correcto funcionamiento del sistema, es necesario el desarrollo de prueba en un entorno de vuelo controlado. Para ello, haremos uso de una aeronave no tripulada y estableceremos la configuración propuesta de los radares.

- 2. En segundo lugar, hemos comprobado que el proceso de filtrado de Kalman consigue valores bastante próximos a las trayectorias ideales. No obstante, resulta interesante establecer un proceso de mejora mediante el uso de las variables utilizadas. Consiguiendo optimizar los valores resultantes.
- 3. Por último, enlazando con el primer punto, la asociación de las detecciones de vuelo mediante el uso de medidas reales nos permitirá conocer en más detalle el funcionamiento del entorno planteado. Lo que nos ayudará a establecer posibles mejoras tanto en el sistema como en proceso de asociación del JPDA.

## Bibliografía

- [1] Paula Höhrová, Jakub Soviar y Włodzimierz Sroka. “Market Analysis of Drones for Civil Use”. En: *LOGI - Scientific Journal on Transport and Logistics* 14 (mar. de 2023), págs. 55-65. DOI: 10.2478/logi-2023-0006.
- [2] Álvaro Duque de Quevedo et al. “Drone Detection With X-Band Ubiquitous Radar”. En: *2018 19th International Radar Symposium (IRS)*. 2018, págs. 1-10. DOI: 10.23919/IRS.2018.8447942.
- [3] U-space. URL: <https://www.mitma.gob.es/aviacion-civil/politica-espacio-aereo/portal-para-la-coordinacion-del-u-space-en-espana/concepto-uspace>.
- [4] Estrategia 2.0. URL: [https://ec.europa.eu/commission/presscorner/detail/es/ip\\_22\\_7076](https://ec.europa.eu/commission/presscorner/detail/es/ip_22_7076).
- [5] Michael Jian, Zhenzhong Lu y Victor C. Chen. “Drone detection and tracking based on phase-interferometric Doppler radar”. En: *2018 IEEE Radar Conference (RadarConf18)*. 2018, págs. 1146-1149. DOI: 10.1109/RADAR.2018.8378723.
- [6] Jarez S. Patel, Francesco Fioranelli y David Anderson. “Review of radar classification and RCS characterisation techniques for small UAVs or drones”. En: *IET Radar, Sonar & Navigation* 12.9 (2018), págs. 911-919. DOI: <https://doi.org/10.1049/iet-rsn.2018.0020>. eprint: <https://ietresearch.onlinelibrary.wiley.com/doi/pdf/10.1049/iet-rsn.2018.0020>. URL: <https://ietresearch.onlinelibrary.wiley.com/doi/abs/10.1049/iet-rsn.2018.0020>.
- [7] RADAR MODULE. URL: <https://radar-sensor.com/products/radar-modules/sr-1200e.html>.
- [8] Pramod R. Gunjal et al. “Moving Object Tracking Using Kalman Filter”. En: *2018 International Conference On Advances in Communication and Computing Technology (ICACCT)*. 2018, págs. 544-547. DOI: 10.1109/ICACCT.2018.8529402.
- [9] Proyecto BUBBLES. URL: <https://www.kalmanfilter.net/kalmanmulti.html>.
- [10] Qingfei Gao, Xiang Wang y Yang Liu. “A Kalman Filter-Based Method for Diagnosing the Structural Condition of Medium- and Small-Span Beam Bridges during Brief Traffic Interruptions”. En: *Sensors* 20.15 (2020). ISSN: 1424-8220. DOI: 10.3390/s20154130. URL: <https://www.mdpi.com/1424-8220/20/15/4130>.
- [11] Peter Carniglia, Bhashyam Balaji y Anthony Damini. “Investigation of Sensor Bias and Signal Quality on Target Tracking with Multiple Radars”. En: *2022 IEEE International Instrumentation and Measurement Technology Conference (I2MTC)*. 2022, págs. 1-6. DOI: 10.1109/I2MTC48687.2022.9806656.

- [12] Shuoyuan Xu et al. “Real-time Implementation of YOLO+JPDA for Small Scale UAV Multiple Object Tracking”. En: *2018 International Conference on Unmanned Aircraft Systems (ICUAS)*. 2018, págs. 1336-1341. DOI: 10.1109/ICUAS.2018.8453398.
- [13] SESAR Join Undertaking. URL: <https://www.sesarju.eu/>.
- [14] U-space Blueprint. URL: <https://www.sesarju.eu/sites/default/files/documents/reports/U-space%20Blueprint%20brochure%20final.PDF>.
- [15] Proyecto BUBBLES. URL: <https://www.sesarju.eu/projects/bubbles>.
- [16] *U-space separation management service: concept definition and validation*. Zenodo, dic. de 2022. DOI: 10.5281/zenodo.7438005. URL: <https://doi.org/10.5281/zenodo.7438005>.
- [17] SURVEILANCE. URL: <https://skybrary.aero/articles/surveillance#:~:text=The%20concept%20of%20surveillance%20is,controller%20on%20a%20situation%20display..>
- [18] SURVEILANCE<sub>ATM</sub>. URL: [https://www.atmmasterplan.eu/data/sesar\\_solutions/20431962](https://www.atmmasterplan.eu/data/sesar_solutions/20431962).
- [19] P. Izquierdo. “Evaluation of Non-Cooperative Sensors for Sense and Avoid in UAV Systems”. En: 2019. URL: <https://api.semanticscholar.org/CorpusID:212903379>.
- [20] Juan A. Besada et al. “Review and Simulation of Counter-UAS Sensors for Unmanned Traffic Management”. En: *Sensors* 22.1 (2022). ISSN: 1424-8220. DOI: 10.3390/s22010189. URL: <https://www.mdpi.com/1424-8220/22/1/189>.
- [21] Yicheng Liu et al. “Trajectory and image-based detection and identification of UAV”. En: *The Visual Computer* 37 (jul. de 2021). DOI: 10.1007/s00371-020-01937-y.
- [22] RADAR MODULE. URL: <https://www.radartutorial.eu/01.basics/Radars%20Accuracy.en.html>.
- [23] RADAR MODULE. URL: <https://apps.dtic.mil/sti/tr/pdf/ADA569702.pdf>.
- [24] Objetivos de Desarrollo Sostenible. URL: <https://www.un.org/sustainabledevelopment/es/objetivos-de-desarrollo-sostenible/>.
- [25] Industria, Innovación e Infraestructura. URL: <https://www.un.org/sustainabledevelopment/es/infrastructure/>.
- [26] Ciudades y Comunidades Sostenibles. URL: <https://www.un.org/sustainabledevelopment/es/cities/>.