



# GRADO EN INGENIERÍA EN TECNOLOGÍAS DE TELECOMUNICACIÓN

TRABAJO FIN DE GRADO

## DESARROLLO DE UN SISTEMA DE DETECCIÓN Y CARACTERIZACIÓN DE SEÑALES RF DE DRONES MEDIANTE RADIO DEFINIDA POR SOFTWARE

Autor: Pedro Marcos Plaza

Director: Javier Matanza Domingo

Madrid



### Declaración de originalidad

Declaro bajo mi responsabilidad que el Proyecto presentado con el título **Desarrollo de un Sistema de Detección y Caracterización de Señales RF de Drones mediante Radio Definida por Software** en la ETS de Ingeniería – ICAI de la Universidad Pontificia Comillas en el curso académico 2025-2026 es de mi autoría y no ha sido presentado con anterioridad a otros efectos. El Proyecto no es plagio de otro, ni total ni parcialmente y la información que ha sido tomada de otros documentos está debidamente referenciada.

### Uso de Inteligencia Artificial<sup>1</sup>

Declaro bajo mi responsabilidad que (indicar la opción correcta):

☐ No he utilizado Inteligencia Artificial en la elaboración del presente documento.

☒ He utilizado Inteligencia Artificial en la elaboración del presente documento y/o del Anexo B siempre en las condiciones permitidas por la Universidad Pontificia Comillas, es decir, aplicando el Nivel 2 de la [Escala de Evaluación de Perkins et al. \(2024\)](#): *“La IA puede utilizarse para actividades previas a la tarea, como la lluvia de ideas, la descripción y la investigación inicial. Este nivel se centra en el uso de la IA para la planificación, las síntesis y la generación de ideas, pero las evaluaciones deben hacer hincapié en la capacidad de desarrollar y refinar estas ideas de forma independiente”*. En concreto, las Inteligencia Artificial ha sido empleada para:

La Inteligencia Artificial generativa (ChatGPT y Claude) se ha utilizado como herramienta de apoyo para la búsqueda de ideas, la estructuración de la memoria, la mejora de la redacción, la explicación de conceptos teóricos y la revisión de textos. La programación en MATLAB, el desarrollo experimental con Pluto SDR, el análisis de las señales RF, la obtención de resultados y las conclusiones del proyecto han sido realizados y validados por el autor.



Firmado: Pedro Marcos Plaza

Fecha: 23/06/2026

### Autorización para la entrega del Proyecto

| El Director del Proyecto                                                                                         | El co-Director del Proyecto (si aplica) |
|------------------------------------------------------------------------------------------------------------------|-----------------------------------------|
| Firmado por MATANZA DOMINGO JAVIER - ***4397** el día 24/06/2026 con un certificado emitido por AC FNMT Usuarios |                                         |
| Fdo: Javier Matanza Domingo                                                                                      | Fdo:                                    |
| Fecha: 23/06/2026                                                                                                | Fecha:                                  |

<sup>1</sup> Esta declaración se refiere al uso de la Inteligencia Artificial generativa para realizar los documentos del Proyecto (Anexo B y Memoria). No aplica a Proyectos donde, por su naturaleza, deban emplear inteligencia artificial como parte de los mismos (aplicación de técnicas de aprendizaje automático, redes neuronales, análisis de datos...)





# GRADO EN INGENIERÍA EN TECNOLOGÍAS DE TELECOMUNICACIÓN

TRABAJO FIN DE GRADO

## DESARROLLO DE UN SISTEMA DE DETECCIÓN Y CARACTERIZACIÓN DE SEÑALES RF DE DRONES MEDIANTE RADIO DEFINIDA POR SOFTWARE

Autor: Pedro Marcos Plaza

Director: Javier Matanza Domingo

Madrid



# Agradecimientos

Quiero expresar mi más sincero agradecimiento a mi tutor, Javier Matanza Domingo, por su gran ayuda, dedicación y apoyo durante el desarrollo de este Trabajo Fin de Grado, tanto a nivel académico como personal. Su orientación y confianza han sido fundamentales para la realización de este proyecto.

Asimismo, quiero agradecer a mi familia, por su apoyo incondicional, su paciencia y su constante ánimo a lo largo de esta etapa.

Por último, gracias a mis amigos, por acompañarme durante estos años, por su apoyo y por todos los momentos compartidos, que han hecho este camino mucho más enriquecedor.



# **DESARROLLO DE UN SISTEMA DE DETECCIÓN Y CARACTERIZACIÓN DE SEÑALES RF DE DRONES MEDIANTE RADIO DEFINIDA POR SOFTWARE**

**Autor: Marcos Plaza, Pedro.**

Director: Matanza Domingo, Javier.

Entidad Colaboradora: ICAI – Universidad Pontificia Comillas

## **RESUMEN DEL PROYECTO**

Este proyecto desarrolla un sistema completo de detección y caracterización espectral de señales de dron mediante radio definida por software (SDR). Se han caracterizado cinco señales de referencia, generado señales simuladas equivalentes, implementado una plataforma de transmisión y recepción con el ADALM-PLUTO, y diseñado dos algoritmos de detección de ancho de banda. Los resultados, obtenidos sobre 170.000 estimaciones, muestran tasas de acierto del 100% hasta -10 dB de SNR y un comportamiento robusto incluso en condiciones de ruido severas.

**Palabras clave:** SDR, detección de drones, ADALM-PLUTO

### **1. Introducción**

El uso creciente de drones, tanto en aplicaciones civiles como en contextos potencialmente hostiles, ha impulsado el desarrollo de sistemas capaces de detectar su presencia de forma fiable. Entre las distintas estrategias disponibles (acústica, óptica, radar, RF), la detección basada en la señal de radiofrecuencia que el propio dron emite para comunicarse con su mando resulta especialmente atractiva, ya que no depende de condiciones de visibilidad ni de la cooperación del objetivo. Dentro de este enfoque, la radio definida por software (SDR) se ha consolidado como una herramienta flexible y de bajo coste para implementar receptores reconfigurables capaces de analizar el espectro radioeléctrico [1], [2].

Este TFG aborda el problema desde una perspectiva práctica: caracterizar señales reales de varios modelos de dron, generar señales sintéticas equivalentes, y desarrollar algoritmos capaces de estimar los límites de la banda ocupada por estas señales a partir de su espectro, incluso en presencia de ruido. Todo ello se valida de forma experimental mediante un dispositivo SDR real, el ADALM-PLUTO, en lugar de quedarse en un entorno puramente simulado.

### **2. Definición del proyecto**

El proyecto se plantea cinco objetivos: analizar señales reales de distintos drones; generar y estudiar señales OFDM simuladas equivalentes; desarrollar algoritmos de detección espectral y estimación de ancho de banda; comparar el comportamiento de señales simuladas y reales frente a dichos algoritmos; y validar experimentalmente los resultados mediante transmisiones y recepciones reales con el ADALM-PLUTO.

El trabajo parte de cinco señales de referencia, correspondientes a distintos modelos y protocolos de dron (M250, V2, OcuSync 2, Mini 2 y DJI Neo), de las que se dispone de tres capturas reales por señal, facilitadas por el director del proyecto. Sobre esta base, el proyecto se estructura en tres bloques: generación y caracterización de señales, implementación de la plataforma SDR, y desarrollo y evaluación de los algoritmos de detección.

### 3. Descripción del modelo/sistema/herramienta

El sistema desarrollado sigue el flujo representado en la Ilustración 1. En primer lugar, se caracterizan espectralmente las cinco señales reales, obteniendo sus parámetros de banda de referencia ( $f_l$ ,  $f_h$ , ancho de banda), y se generan señales simuladas equivalentes mediante ruido gaussiano acotado en frecuencia, una aproximación cuya validez se respalda en el comportamiento estadístico de las señales OFDM con un elevado número de subportadoras [5]. Las señales reales se someten además a un proceso de acotación espectral, que elimina componentes ajenas a la banda de interés y atenúa picos anómalos, preservando la fase para obtener una versión limpia en el dominio temporal.

Las tres categorías de señal resultantes (simulada, real y real acotada) se remuestrean a 61,44 MHz, la frecuencia máxima soportada por el ADALM-PLUTO [6], y se transmiten y reciben mediante este dispositivo, tras un proceso de prueba y error que determinó que la configuración física más típica (antenas conectadas directamente sobre un único Pluto) era la única viable frente a alternativas con mayor separación física.

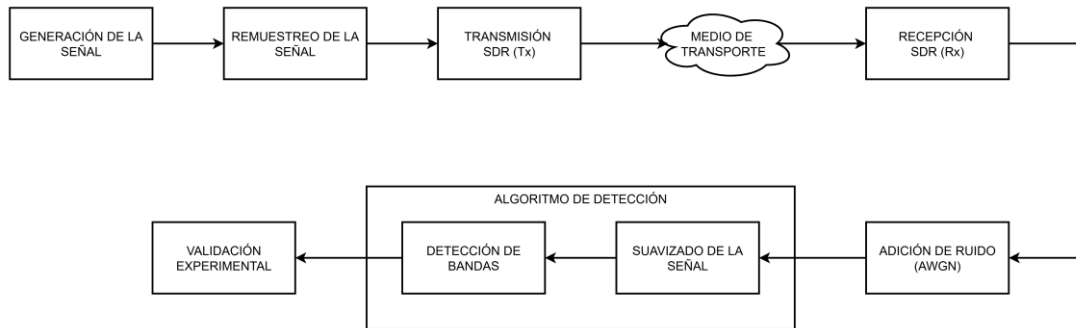


*Ilustración 1: Arquitectura general del sistema desarrollado*

Sobre la señal recibida, y tras añadir ruido AWGN controlado, se aplican dos algoritmos de detección desarrollados de forma propia. El primero suaviza el espectro mediante media móvil y localiza los bordes de la banda a partir de su derivada y un umbral configurable. El segundo suaviza el espectro mediante promediado por bloques y localiza los bordes a partir de las diferencias entre bloques. Ambos algoritmos evalúan varios valores de step y combinan los resultados mediante la mediana, como mecanismo de robustez frente a estimaciones individuales anómalas.

#### 4. Resultados

La validación experimental se realizó mediante una campaña de 170.000 estimaciones, combinando cinco drones, tres tipos de señal, dos algoritmos, dos modos de detección y diez niveles de SNR (de 10 a -35 dB), con 100 repeticiones independientes por punto. Para cada combinación se calculó el porcentaje de estimaciones válidas, considerando válida una estimación cuando se encuentra dentro de un margen del 1% del ancho de banda total disponible respecto al valor de referencia.

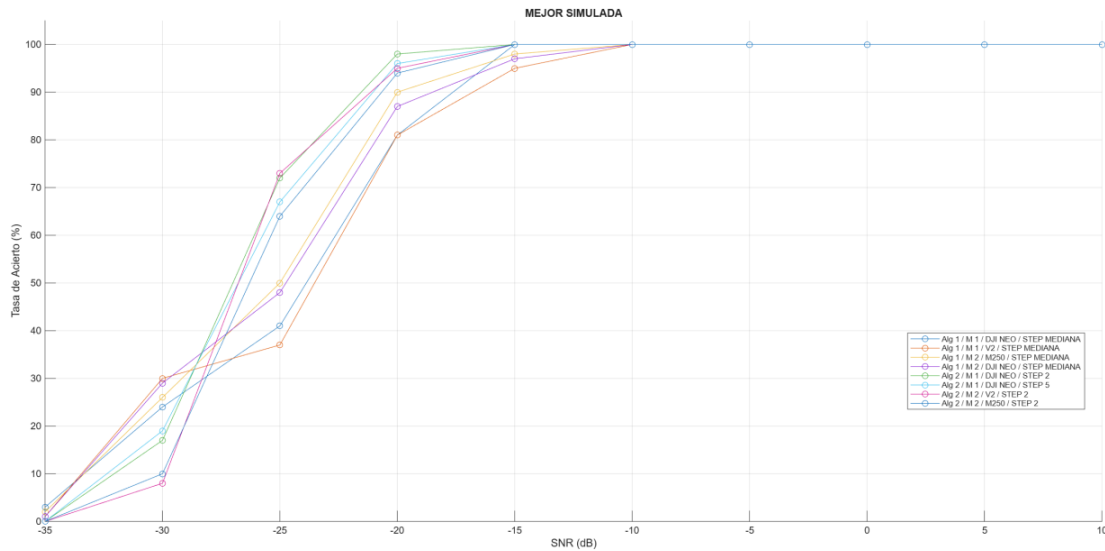


*Ilustración 2: Tasa de detecciones válidas frente a SNR para las mejores configuraciones obtenidas sobre señales simuladas.*

Los resultados son notablemente favorables. Las mejores configuraciones, tanto sobre señales simuladas como sobre señales reales acotadas, mantienen el 100% de detecciones válidas hasta -10 dB, y conservan tasas de acierto superiores al 90% incluso a -20 dB, un nivel de SNR en el que la banda ocupada resulta completamente invisible en el espectro de magnitud sin procesar. Incluso en los casos más desfavorables dentro de cada categoría, el sistema se mantiene funcional en un margen amplio de SNR, con la excepción de una captura real de calidad especialmente baja (M250), cuyo resultado deficiente se atribuye a las condiciones de la propia captura y no a una limitación de los algoritmos.

## 5. Conclusiones

El sistema desarrollado cumple los cinco objetivos planteados al inicio del proyecto. La generación de señales simuladas mediante ruido acotado en frecuencia resulta una aproximación válida, confirmada por su comportamiento similar al de las señales reales acotadas ante los algoritmos de detección. La plataforma SDR, pese a las dificultades iniciales con configuraciones físicas más ambiciosas, resultó robusta en su configuración final. Los dos algoritmos de detección muestran un comportamiento complementario: el primero más preciso con ruido moderado, el segundo más robusto con ruido severo y con señales reales sin procesar.

Como líneas de trabajo futuro se proponen, entre otras, la resolución del problema de sincronización entre unidades SDR independientes, la mejora de los algoritmos frente a armónicos espurios propios de señales reales, y la incorporación de técnicas de aprendizaje automático para construir un clasificador capaz de identificar el modelo concreto de dron a partir de la señal detectada.

## 6. Referencias

- [1] Analog Devices. "ADALM-PLUTO Active Learning Module – Product Overview". Analog Devices Inc., Norwood, MA, USA, 2017.
- [2] Lichtman, M. PySDR: A Guide to SDR and DSP using Python. 2022. Disponible en: <https://pysdr.org/content/pluto.html>.
- [3] Aouladhadj, D.; Kpre, E.; Deniau, V.; Kharchouf, A.; Gransart, C.; Gaquière, C. "Drone Detection and Tracking Using RF Identification Signals". *Sensors*, vol. 23, no. 17, p. 7650, Septiembre de 2023.
- [4] Taha, B.; Shoufan, A. "Machine Learning-Based Drone Detection and Classification: State-of-the-Art in Research". *IEEE Access*, vol. 7, pp. 138669-138682, 2019.
- [5] Ochiai, H.; Imai, H. "On the distribution of the peak-to-average power ratio in OFDM signals". *IEEE Transactions on Communications*, vol. 49, no. 2, pp. 282-289, Feb. 2001.
- [6] Analog Devices, "PLUTO HDL Project — HDL Reference Design Documentation". Disponible en: <https://analogdevicesinc.github.io/hdl/projects/pluto/index.html>.

# **DEVELOPMENT OF A DETECTION AND CHARACTERIZATION SYSTEM FOR DRONE RF SIGNALS USING SOFTWARE DEFINED RADIO**

**Author: Marcos Plaza, Pedro.**

Supervisor: Matanza Domingo, Javier.

Collaborating Entity: ICAI – Universidad Pontificia Comillas

## **ABSTRACT**

This project develops a complete system for the detection and spectral characterization of drone signals using Software Defined Radio (SDR). Five reference drone signals were characterized, equivalent simulated signals were generated, an ADALM-PLUTO-based transmission and reception platform was implemented, and two bandwidth detection algorithms were designed. The results, obtained from 170,000 individual estimations, show 100% detection rates down to an SNR of  $-10$  dB and robust performance even under severe noise conditions.

**Keywords:** SDR, drone detection, ADALM-PLUTO

## **1. Introduction**

The increasing use of drones in both civilian applications and potentially hostile scenarios has driven the development of systems capable of reliably detecting their presence. Among the different available approaches (acoustic, optical, radar, and RF-based), radio-frequency detection is particularly attractive because it exploits the signals that drones naturally transmit to communicate with their controllers, without requiring visibility conditions or cooperation from the target. Within this context, Software Defined Radio (SDR) has become a flexible and low-cost solution for implementing reconfigurable receivers capable of analysing the radio spectrum [1], [2].

This Final Degree Project addresses the problem from a practical perspective: characterising real signals from several drone models, generating equivalent synthetic signals, and developing algorithms capable of estimating the occupied bandwidth of these signals from their spectra, even in the presence of noise. All developments are experimentally validated using a real SDR platform, the ADALM-PLUTO, rather than remaining in a purely simulated environment.

## **2. Project definition**

The project pursues five main objectives: to analyse real signals from different drones; to generate and study equivalent simulated OFDM signals; to develop spectral detection and bandwidth estimation algorithms; to compare the behaviour of simulated and real signals when processed by these algorithms; and to experimentally validate the results through real transmissions and receptions using the ADALM-PLUTO platform.

The work is based on five reference signals corresponding to different drone models and communication protocols (M250, V2, OcuSync 2, Mini 2, and DJI Neo). For each signal, three real captures provided by the project supervisor were available. Based on this dataset, the project is structured into three main blocks: signal generation and characterisation, SDR platform implementation, and development and evaluation of detection algorithms.

### 3. System description

The developed system follows the architecture shown in Illustration 1. First, the five real signals are spectrally characterised to obtain their reference parameters (lower frequency limit, upper frequency limit, and bandwidth). Equivalent simulated signals are then generated by means of frequency-limited Gaussian noise, an approach supported by the statistical behaviour of OFDM signals with a large number of subcarriers [5]. The real signals also undergo a spectral bounding process that removes components outside the band of interest and attenuates anomalous peaks while preserving phase information to obtain a cleaner time-domain representation.

The three resulting signal categories (simulated, real, and bounded real signals) are resampled to 61.44 MHz, the maximum sampling frequency supported by the ADALM-PLUTO [6], and transmitted and received using this device. After an extensive trial-and-error process, it was concluded that the most conventional physical configuration, with both antennas connected directly to a single Pluto device, was the only viable solution compared with alternatives involving greater physical separation.

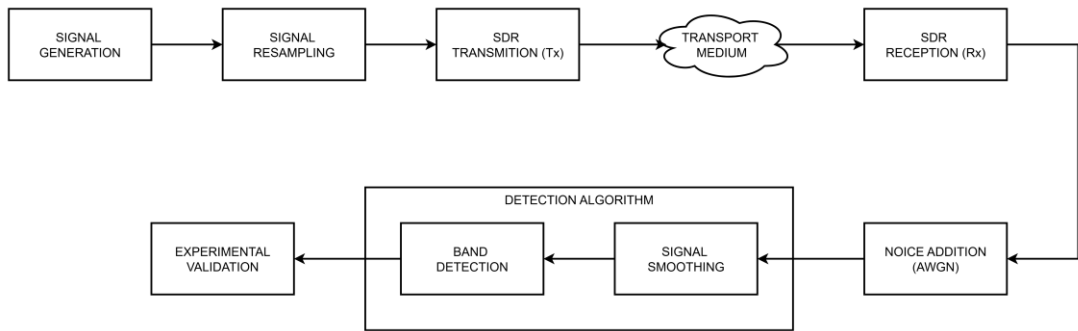


*Illustration 1: General architecture of the developed system*

Controlled AWGN is then added to the received signals, and two original detection algorithms are applied. The first algorithm smooths the spectrum using a moving average and estimates the bandwidth edges from its derivative and a configurable threshold. The second algorithm smooths the spectrum through block averaging and estimates the edges from differences between adjacent blocks. Both algorithms evaluate several step values and combine the results using the median as a robustness mechanism against anomalous individual estimations.

## 4. Results

The experimental validation consisted of a campaign of 170,000 estimations, combining five drones, three signal types, two algorithms, two detection modes, and ten SNR levels (from 10 dB to  $-35$  dB), with 100 independent repetitions per operating point. For each combination, the percentage of valid estimations was calculated, considering an estimation to be valid when it lies within a margin of 1% of the total available bandwidth with respect to the reference value.

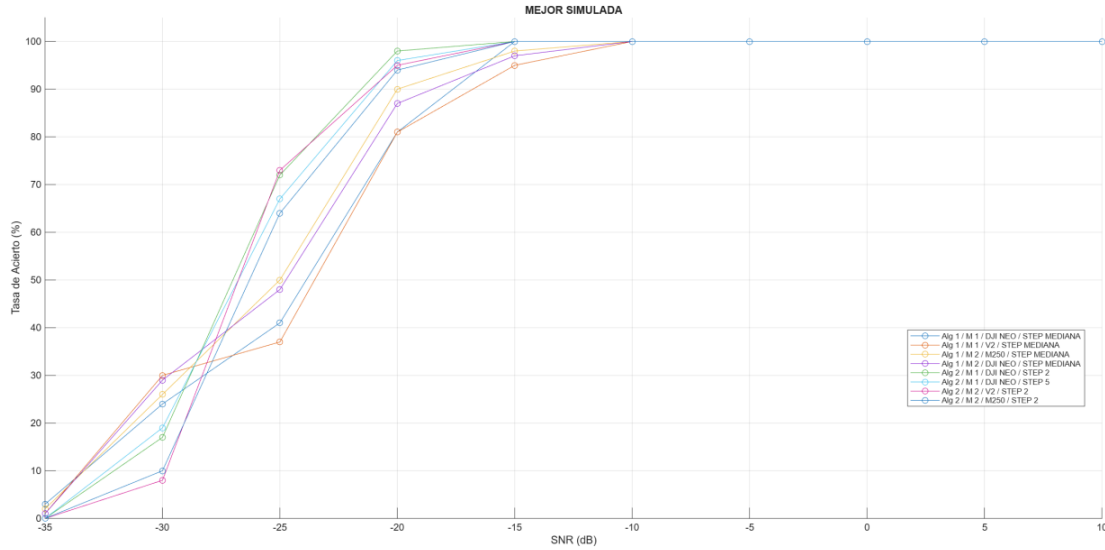


*Illustration 2: Rate of valid detections versus SNR for the best-performing configurations obtained on simulated signals.*

The results are remarkably favourable. The best configurations, for both simulated and bounded real signals, maintain 100% valid detections down to an SNR of  $-10$  dB and preserve success rates above 90% even at  $-20$  dB, a noise level at which the occupied band becomes completely invisible in the raw magnitude spectrum. Even in the least favourable cases within each signal category, the system remains operational over a wide SNR range, with the exception of a particularly low-quality real capture (M250), whose poor performance is attributed to the acquisition conditions rather than to limitations of the proposed algorithms.

## 5. Conclusions

The developed system successfully achieves the five objectives established at the beginning of the project. The generation of simulated signals through frequency-limited noise proves to be a valid approximation, as confirmed by its behaviour, which closely resembles that of bounded real signals when processed by the detection algorithms. Despite the initial difficulties associated with more ambitious physical configurations, the final SDR platform demonstrated robust performance. The two detection algorithms exhibit complementary behaviour: the first is more accurate under moderate noise conditions, whereas the second is more robust under severe noise and when processing unfiltered real signals.

Several future research directions are proposed, including solving the synchronisation problem between independent SDR units, improving the algorithms to mitigate spurious harmonics inherent to real signals, and incorporating machine learning techniques to develop a classifier capable of identifying the specific drone model from the detected RF signal.

## 6. References

- [1] Analog Devices. "ADALM-PLUTO Active Learning Module – Product Overview". Analog Devices Inc., Norwood, MA, USA, 2017.
- [2] Lichtman, M. PySDR: A Guide to SDR and DSP using Python. 2022. Disponible en: <https://pysdr.org/content/pluto.html>.
- [3] Aouladhadj, D.; Kpre, E.; Deniau, V.; Kharchouf, A.; Gransart, C.; Gaquière, C. "Drone Detection and Tracking Using RF Identification Signals". *Sensors*, vol. 23, no. 17, p. 7650, Septiembre de 2023.
- [4] Taha, B.; Shoufan, A. "Machine Learning-Based Drone Detection and Classification: State-of-the-Art in Research". *IEEE Access*, vol. 7, pp. 138669-138682, 2019.
- [5] Ochiai, H.; Imai, H. "On the distribution of the peak-to-average power ratio in OFDM signals". *IEEE Transactions on Communications*, vol. 49, no. 2, pp. 282-289, Feb. 2001.
- [6] Analog Devices, "PLUTO HDL Project — HDL Reference Design Documentation". Disponible en: <https://analogdevicesinc.github.io/hdl/projects/pluto/index.html>.

## *Índice de la memoria*

|                                                                                |           |
|--------------------------------------------------------------------------------|-----------|
| <b>Capítulo 1. Introducción .....</b>                                          | <b>9</b>  |
| 1.1 Motivación del proyecto.....                                               | 9         |
| <b>Capítulo 2. Descripción de las Tecnologías.....</b>                         | <b>11</b> |
| 2.1 Radio Definida por Software (SDR) .....                                    | 11        |
| 2.2 ADALM-PLUTO .....                                                          | 12        |
| 2.3 OFDM (Orthogonal Frequency Division Multiplexing) .....                    | 15        |
| 2.4 Ruido AWGN.....                                                            | 16        |
| 2.5 MATLAB y herramientas de procesamiento de señal.....                       | 17        |
| <b>Capítulo 3. Estado de la Cuestión .....</b>                                 | <b>18</b> |
| 3.1 Sistemas de detección de drones: panorama tecnológico.....                 | 18        |
| 3.2 Detección basada en radiofrecuencia .....                                  | 19        |
| 3.3 Plataformas SDR de bajo coste para experimentación con señales de RF ..... | 21        |
| 3.4 Estimación de ocupación espectral y detección de bordes de banda.....      | 22        |
| 3.5 Síntesis y posicionamiento de este TFG.....                                | 23        |
| <b>Capítulo 4. Definición del Trabajo .....</b>                                | <b>24</b> |
| 4.1 Justificación.....                                                         | 24        |
| 4.2 Metodología.....                                                           | 25        |
| 4.3 Planificación y Estimación Económica .....                                 | 26        |
| 4.3.1 Planificación temporal.....                                              | 26        |
| 4.3.2 Estimación económica.....                                                | 27        |
| <b>Capítulo 5. Obtención, Procesado y Preparación de Señales RF.....</b>       | <b>29</b> |
| 5.1 Caracterización espectral de las señales reales.....                       | 29        |
| 5.1.1 Fundamento teórico.....                                                  | 30        |
| 5.1.2 Representación espectral de las señales de dron .....                    | 31        |
| 5.2 Estimación de los parámetros de banda de las señales reales.....           | 38        |
| 5.2.1 Fundamento teórico.....                                                  | 38        |
| 5.2.2 Procedimiento.....                                                       | 38        |
| 5.2.3 Resultados .....                                                         | 38        |
| 5.3 Generación de señales OFDM simuladas.....                                  | 40        |

|                                                                                      |                                                                                             |           |
|--------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------|-----------|
| 5.4                                                                                  | Comparación entre señales reales y simuladas.....                                           | 42        |
| 5.5                                                                                  | Acotación espectral y de amplitudes de las señales reales .....                             | 44        |
| 5.6                                                                                  | Unificación de las señales en una estructura común .....                                    | 47        |
| 5.7                                                                                  | Remuestreo de las señales para su transmisión con el ADALM-PLUTO.....                       | 48        |
| 5.7.1                                                                                | Fundamento teórico.....                                                                     | 48        |
| 5.7.2                                                                                | Procedimiento.....                                                                          | 49        |
| 5.7.3                                                                                | Justificación del valor de destino .....                                                    | 49        |
| 5.7.4                                                                                | Validez del muestreo.....                                                                   | 49        |
| <b>Capítulo 6. Diseño e Implementación de la Plataforma SDR .....</b>                |                                                                                             | <b>50</b> |
| 6.1                                                                                  | Configuración física de la plataforma SDR.....                                              | 50        |
| 6.2                                                                                  | Validación inicial y fundamentos de operación del ADALM-PLUTO con una señal de tono puro 52 |           |
| 6.2.1                                                                                | Parámetros de la señal y de la captura.....                                                 | 52        |
| 6.2.2                                                                                | Configuración del dispositivo desde MATLAB .....                                            | 53        |
| 6.2.3                                                                                | Procedimiento.....                                                                          | 54        |
| 6.2.4                                                                                | Efecto de la frecuencia portadora: comparación entre 868 MHz y 2,4 GHz.....                 | 58        |
| 6.3                                                                                  | Señal OFDM con cable directo entre Rx y Tx .....                                            | 64        |
| 6.4                                                                                  | Señal OFDM con dos Plutos independientes .....                                              | 64        |
| 6.5                                                                                  | Señal OFDM con un Pluto y cables (antenas conectadas mediante cable SMA).....               | 67        |
| 6.6                                                                                  | Señal OFDM con antenas directas sobre un único Pluto (solución adoptada) .....              | 69        |
| 6.7                                                                                  | Barrido de parámetros y transmisión de las señales del proyecto .....                       | 70        |
| 6.7.1                                                                                | Métricas de evaluación.....                                                                 | 71        |
| 6.7.2                                                                                | Resultados del barrido .....                                                                | 72        |
| 6.8                                                                                  | Transmisión y recepción del conjunto completo de señales.....                               | 74        |
| <b>Capítulo 7. Desarrollo de los Algoritmos de Detección y Caracterización .....</b> |                                                                                             | <b>76</b> |
| 7.1                                                                                  | Introducción y planteamiento del problema .....                                             | 76        |
| 7.2                                                                                  | Elementos comunes a los dos algoritmos.....                                                 | 77        |
| 7.2.1                                                                                | Modelo de ruido .....                                                                       | 77        |
| 7.2.2                                                                                | Suavizado del espectro .....                                                                | 78        |
| 7.2.3                                                                                | Detección con múltiples steps y estimación final por mediana.....                           | 78        |
| 7.2.4                                                                                | Modos de detección .....                                                                    | 79        |
| 7.3                                                                                  | Algoritmo 1: suavizado por media móvil y detección por derivada.....                        | 80        |
| 7.3.1                                                                                | Suavizado por media móvil .....                                                             | 81        |

|                                                                                       |            |
|---------------------------------------------------------------------------------------|------------|
| 7.3.2 Cálculo de la derivada con step variable .....                                  | 82         |
| 7.3.3 Detección de bordes por umbral .....                                            | 84         |
| 7.3.4 Resultados experimentales .....                                                 | 87         |
| 7.4 Algoritmo 2: suavizado por bloques y detección por diferencias .....              | 95         |
| 7.4.1 Suavizado por bloques .....                                                     | 96         |
| 7.4.2 Detección por diferencias entre bloques .....                                   | 97         |
| 7.4.3 Resultados experimentales .....                                                 | 100        |
| 7.5 Comparación entre algoritmos .....                                                | 106        |
| <b>Capítulo 8. Resultados Experimentales y Análisis de Resultados .....</b>           | <b>109</b> |
| 8.1 Criterio de selección de los resultados presentados .....                         | 111        |
| 8.2 Resultados más favorables .....                                                   | 112        |
| 8.2.1 Señales simuladas .....                                                         | 112        |
| 8.2.2 Señales reales acotadas .....                                                   | 114        |
| 8.2.3 Señales reales .....                                                            | 116        |
| 8.3 Resultados más desfavorables .....                                                | 118        |
| 8.3.1 Señales simuladas .....                                                         | 118        |
| 8.3.2 Señales reales acotadas .....                                                   | 120        |
| 8.3.3 Señales reales .....                                                            | 123        |
| 8.4 Síntesis del capítulo .....                                                       | 125        |
| <b>Capítulo 9. Conclusiones y Trabajos Futuros .....</b>                              | <b>127</b> |
| 9.1 Cumplimiento de los objetivos del proyecto .....                                  | 127        |
| 9.2 Evolución del desarrollo: de la versión preliminar a los resultados finales ..... | 129        |
| 9.3 Conclusiones técnicas principales .....                                           | 130        |
| 9.4 Limitaciones del trabajo .....                                                    | 132        |
| 9.5 Líneas de trabajo futuro .....                                                    | 133        |
| 9.6 Conclusión final .....                                                            | 134        |
| <b>Capítulo 10. Bibliografía .....</b>                                                | <b>135</b> |
| <b>Capítulo 11. ANEXO I: ALINEACIÓN DEL PROYECTO CON LOS ODS .....</b>                | <b>138</b> |
| <b>Capítulo 12. ANEXO II .....</b>                                                    | <b>139</b> |
| 12.1 Generación y caracterización de señales (Capítulo 5. ) .....                     | 139        |
| 12.1.1 Generación de señal OFDM simulada .....                                        | 139        |

---

|                                                                        |     |
|------------------------------------------------------------------------|-----|
| 12.1.2 Caracterización espectral de las señales de referencia .....    | 139 |
| 12.2 Plataforma SDR (Capítulo 6. ) .....                               | 140 |
| 12.2.1 Remuestreo de señales para el ADALM-PLUTO .....                 | 140 |
| 12.2.2 Transmisión y recepción con el ADALM-PLUTO.....                 | 140 |
| 12.3 Algoritmos de detección espectral (Capítulo 7. ) .....            | 141 |
| 12.3.1 Algoritmo 1: media móvil y derivada .....                       | 141 |
| 12.3.2 Algoritmo 2: suavizado por bloques y detección por step.....    | 142 |
| 12.4 Validación experimental (Capítulo 8. ) .....                      | 144 |
| 12.4.1 Transmisión y recepción de todas las señales del proyecto ..... | 144 |

## *Índice de figuras*

|                                                                                                                                                                                                         |    |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----|
| Figura 1: Cadena de señal del ADALM-PLUTO, desde la antena hasta el host donde se ejecuta MATLAB.....                                                                                                   | 12 |
| Figura 2: Diagrama de bloques funcional del transceptor AD9363, mostrando las cadenas de recepción y transmisión independientes, los conversores ADC/DAC y la interfaz digital hacia la FPGA [26]. .... | 13 |
| Figura 3: Diagrama de bloques de la familia Xilinx Zynq-7000 SoC, mostrando la separación entre el sistema de procesamiento (núcleos ARM Cortex-A9) y la lógica programable [27]. ....                  | 14 |
| Figura 4: Espectros correspondientes a las tres capturas del dron Mini 2.....                                                                                                                           | 32 |
| Figura 5: Espectros correspondientes a las tres capturas del dron DJI Neo. ....                                                                                                                         | 33 |
| Figura 6: Espectros correspondientes a las tres capturas del dron M250.....                                                                                                                             | 34 |
| Figura 7: Espectros correspondientes a las tres capturas del dron OcuSync 2.....                                                                                                                        | 35 |
| Figura 8: Espectros correspondientes a las tres capturas del dron V2.....                                                                                                                               | 36 |
| Figura 9: Comparación de los espectros de magnitud de la muestra representativa seleccionada para cada una de las cinco señales analizadas.....                                                         | 37 |
| Figura 10: Máscara espectral rectangular idealizada para cada señal, a partir de los parámetros de banda estimados, con escala de magnitud individual (izquierda) y escala común (derecha). ....        | 39 |
| Figura 11: Espectros de magnitud de las señales OFDM-like simuladas para cada una de las cinco señales analizadas. ....                                                                                 | 41 |
| Figura 12: Comparación entre los espectros de las señales reales y las señales OFDM-like simuladas. ....                                                                                                | 43 |
| Figura 13: Espectro original, acotado en frecuencia y acotado en frecuencia y amplitud, para las tres capturas de la señal M250. ....                                                                   | 46 |
| Figura 14: Comparación de los espectros de magnitud de la señal real, la señal real acotada y la señal simulada, para la señal M250. ....                                                               | 48 |

|                                                                                                                                                                                                                                                     |     |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----|
| Figura 15: Espectros de magnitud de la señal transmitida y la señal recibida, en escala lineal y logarítmica, para la prueba con tono puro a $f_c = 868\text{MHz}$ ( $f_{\text{tono}} = 100\text{ kHz}$ , `txGain` = -20 dB, `rxGain` = 15 dB)..... | 55  |
| Figura 16: Diagrama de constelación de la señal transmitida y la señal recibida para la prueba con tono puro a $f_c=868$ . ....                                                                                                                     | 57  |
| Figura 17: Espectros de magnitud de la señal transmitida y la señal recibida, en escala lineal y logarítmica, para la prueba con tono puro a $f_c = 2.400\text{MHz}$ . ....                                                                         | 59  |
| Figura 18: Diagrama de constelación de la señal transmitida y la señal recibida para la prueba con tono puro a $f_c = 2.400\text{MHz}$ . ....                                                                                                       | 60  |
| Figura 19: Espectro de magnitud de la señal de prueba OFDM-like antes y después del remuestreo de 92,16 MHz a 61,44 MHz. ....                                                                                                                       | 63  |
| Figura 20: Espectros de magnitud de la señal transmitida y la señal recibida con la configuración de dos Plutos independientes, en escala lineal y logarítmica. ....                                                                                | 66  |
| Figura 21: Espectros de magnitud de la señal transmitida y la señal recibida con la configuración de un único Pluto y cables de extensión SMA, en escala lineal y logarítmica. ....                                                                 | 68  |
| Figura 22: Espectros de magnitud de la señal transmitida y la señal recibida con la configuración de antenas directas sobre un único Pluto, en escala lineal y logarítmica. ....                                                                    | 69  |
| Figura 23: Espectro de magnitud original, sin ruido añadido. ....                                                                                                                                                                                   | 91  |
| Figura 24: Espectro de magnitud tras añadir ruido AWGN con SNR = -15 dB. ....                                                                                                                                                                       | 92  |
| Figura 25: Derivada bruta del espectro suavizado, sin filtrar. ....                                                                                                                                                                                 | 92  |
| Figura 26: Derivada suavizada mediante movmean. ....                                                                                                                                                                                                | 93  |
| Figura 27: Bordes detectados para step = 12323. ....                                                                                                                                                                                                | 94  |
| Figura 28: Estimación final por mediana ( $f_c = 16,35\text{ MHz}$ ). ....                                                                                                                                                                          | 94  |
| Figura 29: Espectro de magnitud original, sin ruido añadido. ....                                                                                                                                                                                   | 103 |
| Figura 30: Espectro de magnitud tras añadir ruido AWGN con SNR = -15 dB. ....                                                                                                                                                                       | 103 |
| Figura 31: Espectro suavizado mediante promediado por bloques. ....                                                                                                                                                                                 | 104 |
| Figura 32: Bordes detectados para step = 5. ....                                                                                                                                                                                                    | 105 |
| Figura 33: Estimación final por mediana ( $f_l = 7,88\text{ MHz}$ , $f_h = 25,21\text{ MHz}$ ). ....                                                                                                                                                | 105 |

---

|                                                                                                                                                                                                                 |     |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----|
| Figura 34: Tasa de detecciones válidas frente a SNR para las configuraciones con mejor rendimiento sobre señales simuladas."                                                                                    | 112 |
| Figura 35: Tasa de detecciones válidas frente a SNR para las configuraciones con mejor rendimiento sobre señales reales acotadas.....                                                                           | 114 |
| Figura 36: Tasa de detecciones válidas frente a SNR para las configuraciones con mejor rendimiento sobre señales reales sin acotar. ....                                                                        | 116 |
| Figura 37: Tasa de detecciones válidas frente a SNR para las configuraciones con peor rendimiento sobre señales simuladas.....                                                                                  | 118 |
| Figura 38: Tasa de detecciones válidas frente a SNR para las configuraciones con peor rendimiento sobre señales reales acotadas.....                                                                            | 120 |
| Figura 39: Tasa de detecciones válidas frente a SNR para las configuraciones con peor rendimiento sobre señales reales sin acotar. ....                                                                         | 123 |
| Figura 40: Resultados de la versión preliminar del algoritmo de detección, evaluados durante el seguimiento intermedio del proyecto, con ruido añadido directamente sobre la señal simulada en banda base. .... | 129 |

## *Índice de tablas*

|                                                                                                              |     |
|--------------------------------------------------------------------------------------------------------------|-----|
| Tabla 1: Distribución temporal de horas dedicadas al proyecto por tramos. ....                               | 26  |
| Tabla 2: Estimación económica del proyecto.....                                                              | 28  |
| Tabla 3: Parámetros de banda estimados para cada una de las señales reales analizadas. ...                   | 38  |
| Tabla 4: Las diez mejores configuraciones del barrido de parámetros, ordenadas por SNR descendente.....      | 73  |
| Tabla 5: Resultados del Algoritmo 1 por step, para SNR = -15 dB. ....                                        | 88  |
| Tabla 6: Resultados del Algoritmo 1 por step, para SNR = -23 dB. ....                                        | 89  |
| Tabla 7: Resultados del Algoritmo 2 por step, para SNR = -25 dB (Modo 1).....                                | 101 |
| Tabla 8: Resultados del Algoritmo 2 por step, para SNR = -15 dB (Modo 1).....                                | 102 |
| Tabla 9: Comparación de resultados a SNR = -15 dB, único punto evaluado en común entre ambos algoritmos..... | 106 |
| Tabla 10: Mejor rendimiento sobre señales simuladas, por algoritmo, modo y dron. ....                        | 112 |
| Tabla 11: Mejor rendimiento sobre señales reales acotadas, por algoritmo, modo, dron y muestra. ....         | 115 |
| Tabla 12: Mejor rendimiento sobre señales reales sin acotar, por algoritmo, modo, dron y muestra. ....       | 117 |
| Tabla 13: Peor rendimiento sobre señales simuladas, por algoritmo, modo, dron y step. ....                   | 119 |
| Tabla 14: Peor rendimiento sobre señales reales acotadas, por algoritmo, modo, dron, muestra y step. ....    | 121 |
| Tabla 15: Peor rendimiento sobre señales reales sin acotar, por algoritmo, modo, dron, muestra y step. ....  | 123 |

## Capítulo 1. INTRODUCCIÓN

Este capítulo presenta el contexto y la motivación que dan origen a este TFG. Se expone la problemática del uso no autorizado de drones y la necesidad de sistemas de detección fiables, y se introduce el enfoque concreto que se ha seguido en este trabajo: la detección mediante radio definida por software.

### ***1.1 MOTIVACIÓN DEL PROYECTO***

En los últimos años, el uso de drones se ha extendido de forma notable, tanto en aplicaciones civiles y comerciales (fotografía aérea, logística, inspección de infraestructuras) como en el ámbito recreativo. Esta accesibilidad creciente, favorecida por la reducción de costes y la simplicidad de manejo de los modelos actuales, ha traído consigo un riesgo igualmente creciente: el uso no autorizado de drones en espacios aéreos restringidos, con implicaciones que van desde la invasión de la privacidad hasta amenazas directas a la seguridad de infraestructuras críticas.

Un caso especialmente ilustrativo de este riesgo es el incidente ocurrido en el aeropuerto de Gatwick (Reino Unido) entre el 19 y el 21 de diciembre de 2018, cuando varios avistamientos de drones no autorizados obligaron a cerrar la única pista del aeropuerto de forma intermitente durante más de un día. El incidente provocó la cancelación o el desvío de aproximadamente 1.000 vuelos, afectando a cerca de 140.000 pasajeros, y puso de manifiesto la falta de sistemas de detección de drones suficientemente fiables en infraestructuras de esta importancia [25]. Episodios como este, lejos de ser aislados, han impulsado en los últimos años un interés creciente por parte de la industria y la investigación en el desarrollo de sistemas de detección de drones eficaces.

Entre las distintas estrategias de detección disponibles (acústica, óptica, radar, RF), la detección basada en la señal de radiofrecuencia que el propio dron emite para comunicarse con su mando de control resulta especialmente atractiva. A diferencia de los métodos ópticos o acústicos, no depende de condiciones de visibilidad, distancia o ruido ambiental, y a diferencia del radar, no requiere infraestructura dedicada de alto coste, ya que puede implementarse mediante un receptor de radio definida por software (SDR) relativamente accesible. Esta combinación de eficacia y bajo coste de despliegue es lo que motiva, en última instancia, el enfoque de este TFG: explorar hasta qué punto un sistema basado en un único dispositivo SDR de gama de entrada, como el ADALM-PLUTO, es capaz de detectar y caracterizar señales de drones reales de forma fiable.

A este interés técnico se suma una motivación personal: la oportunidad de combinar contenidos propios del procesamiento digital de señal y las comunicaciones por radiofrecuencia, estudiados a lo largo del grado, con un trabajo eminentemente práctico que involucra hardware real, señales reales y un problema de ingeniería de actualidad, en lugar de quedarse en un desarrollo puramente teórico o simulado.

## Capítulo 2. DESCRIPCIÓN DE LAS TECNOLOGÍAS

### 2.1 RADIO DEFINIDA POR SOFTWARE (SDR)

La radio definida por software, o SDR (*Software Defined Radio*), es un paradigma de diseño de sistemas de radiocomunicaciones en el que funciones que tradicionalmente se implementaban mediante hardware dedicado (mezcladores, filtros, moduladores y demoduladores, detectores) se trasladan al dominio del software, ejecutándose sobre un procesador de propósito general o sobre un dispositivo programable como una FPGA [5]. En un sistema SDR, el hardware se limita, en gran medida, a la etapa de radiofrecuencia más básica: la antena, una etapa de amplificación y filtrado analógico, y un convertidor analógico-digital (en recepción) o digital-analógico (en transmisión). A partir de ahí, toda la cadena de procesamiento (down-conversion a banda base, filtrado, modulación o demodulación, detección) se realiza de forma digital, mediante software.

Esta arquitectura ofrece una ventaja determinante frente a un receptor de radiofrecuencia convencional: la flexibilidad. Un mismo equipo SDR puede reconfigurarse, mediante software, para operar en una frecuencia portadora distinta, con un ancho de banda distinto, o incluso para implementar un esquema de modulación completamente diferente, sin necesidad de modificar el hardware. Esta característica es la que hace que un dispositivo SDR resulte especialmente adecuado para un proyecto como este, en el que es necesario trabajar con varias señales de dron distintas, cada una con su propia frecuencia central y ancho de banda, sin tener que recurrir a un receptor dedicado para cada una de ellas.

Para los propósitos de este TFG, el aspecto más relevante de un sistema SDR es que proporciona acceso directo a las muestras IQ (en fase y cuadratura) de la señal recibida, es decir, a su representación digital en banda base. Es precisamente sobre estas muestras IQ sobre las que se aplican, a lo largo de los capítulos 5 (Capítulo 5. ) a 8 (Capítulo 8. ) de esta memoria, tanto el proceso de generación y acotación de señales como los algoritmos de detección espectral desarrollados.

## 2.2 ADALM-PLUTO

El ADALM-PLUTO (también conocido como PlutoSDR) es un módulo de aprendizaje activo desarrollado por Analog Devices, concebido como una plataforma SDR portátil, autónoma y de bajo coste para la experimentación con sistemas de radiofrecuencia [1].

Aunque se trata de un único dispositivo físico, formado por una sola placa de circuito impreso (PCB), sobre ella se integran dos chips principales, cada uno con una función distinta, de forma análoga a como un smartphone integra en una sola placa un chip de procesador, un chip de memoria y un chip de radio, cada uno especializado en una tarea concreta. En el caso del ADALM-PLUTO, estos dos chips son un transceptor de radiofrecuencia AD9363 y una FPGA Xilinx Zynq Z-7010, que integra además un procesador ARM junto con su lógica programable[26]. La Figura 1 resume la cadena de señal completa del dispositivo, desde la antena hasta el host donde se ejecuta MATLAB.

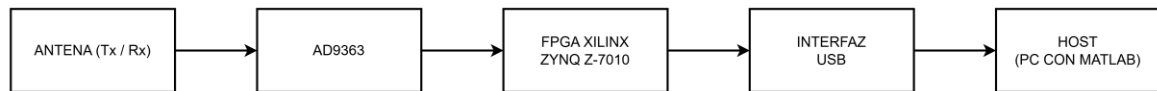


Figura 1: Cadena de señal del ADALM-PLUTO, desde la antena hasta el host donde se ejecuta MATLAB.

El AD9363 es el único de los dos chips que interactúa directamente con la señal de radiofrecuencia. Es un transceptor de altas prestaciones, con un canal de transmisión y un canal de recepción independientes, capaz de operar en un rango de frecuencia de 325 MHz a 3,8 GHz, con un ancho de banda instantáneo configurable de hasta 20 MHz[26]. Integra internamente los conversores analógico-digital y digital-analógico, los osciladores locales y los mezcladores necesarios para realizar la conversión directa entre la señal de radiofrecuencia y su representación en banda base, de modo que entrega ya digitalizadas las muestras en fase y cuadratura (IQ). La Figura 2 muestra el diagrama de bloques funcional de este chip, donde puede apreciarse, entre otros elementos, el límite de 61,44 millones de muestras por segundo (MSPS) en sus cadenas de recepción y transmisión, cifra que, como se justificó en el apartado 5.7, determina la necesidad de remuestrear las señales del proyecto (originalmente a 92,16 MHz) antes de poder transmitir y recibirlas con este dispositivo.

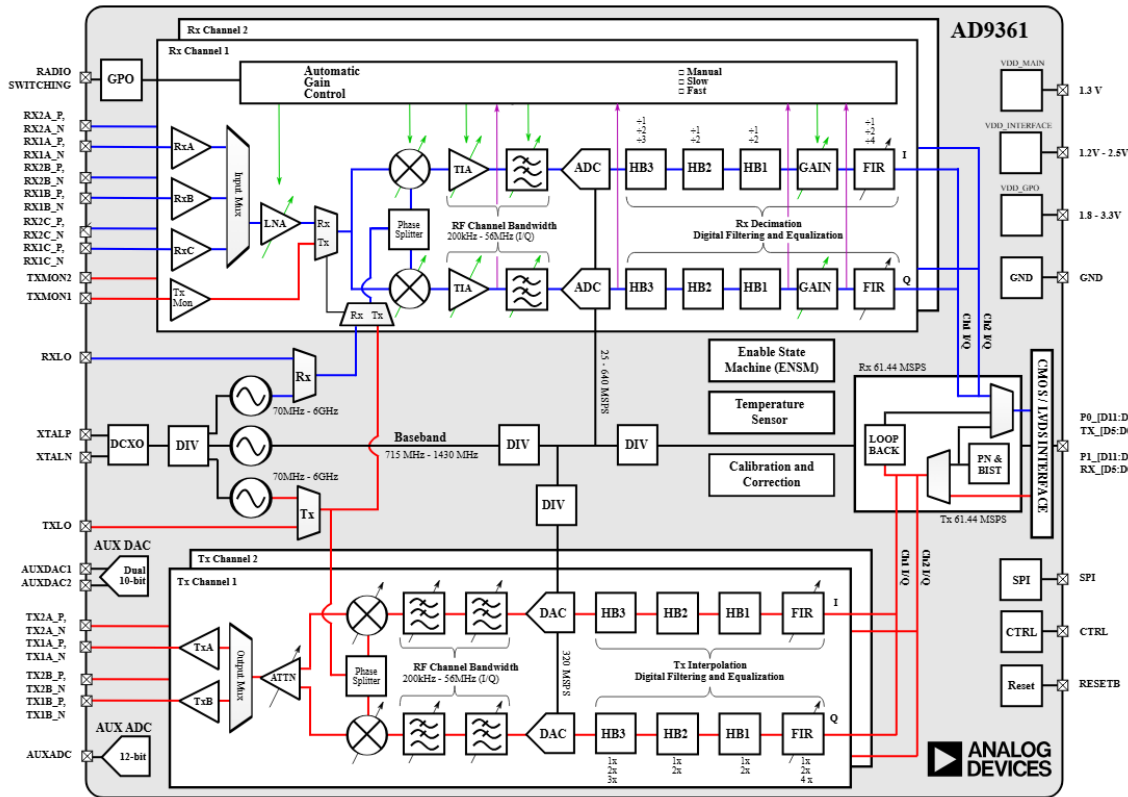


Figura 2: Diagrama de bloques funcional del transceptor AD9363, mostrando las cadenas de recepción y transmisión independientes, los conversores ADC/DAC y la interfaz digital hacia la FPGA [26].

La FPGA Zynq Z-7010, por su parte, no interactúa con la señal de radiofrecuencia en ningún momento; su función es la de actuar como puente digital entre el AD9363 y el host. Recibe las muestras IQ ya digitalizadas, gestiona su transferencia hacia el ordenador a través de la interfaz USB, y se encarga del control del transceptor (sintonía de frecuencia, ganancia, etc.) mediante el firmware estándar del dispositivo. La Figura 3 muestra el diagrama de bloques de la familia Zynq-7000, donde se distingue claramente la separación entre el sistema de procesamiento (los núcleos ARM Cortex-A9, sus cachés y las interfaces de E/S, entre ellas el USB que emplea el Pluto para comunicarse con el host) y la lógica programable. En este TFG no se ha programado ni modificado la lógica interna de esta FPGA, limitándose su uso al firmware estándar que proporciona Analog Devices, por lo que su arquitectura se incluye aquí únicamente con fines ilustrativos, sin entrar en mayor detalle sobre cada uno de sus bloques.

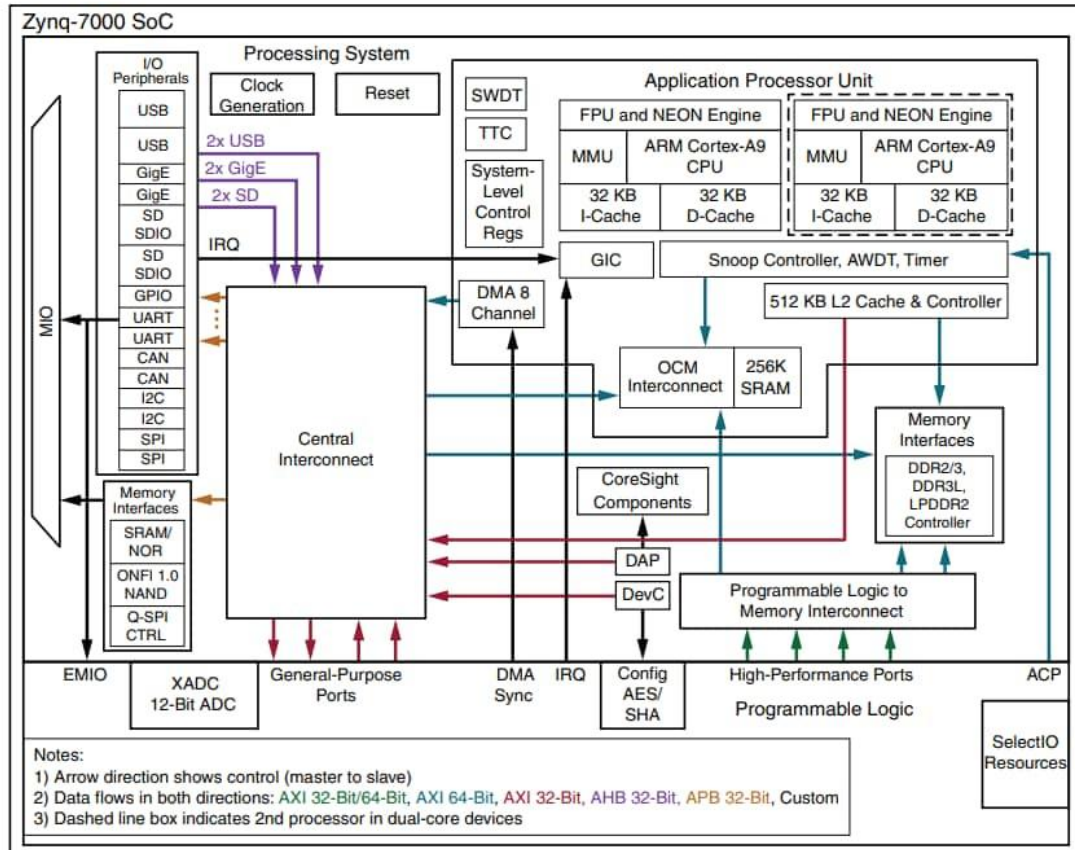


Figura 3: Diagrama de bloques de la familia Xilinx Zynq-7000 SoC, mostrando la separación entre el sistema de procesamiento (núcleos ARM Cortex-A9) y la lógica programable [27].

El ADALM-PLUTO admite, además, modo full-duplex (transmisión y recepción simultáneas) y cuenta con compatibilidad nativa con MATLAB y Simulink a través del paquete de soporte de hardware correspondiente [12], lo que ha permitido integrarlo de forma directa en el flujo de trabajo de este TFG, sin necesidad de herramientas de programación adicionales.

## **2.3 OFDM (ORTHOGONAL FREQUENCY DIVISION MULTIPLEXING)**

OFDM es una técnica de modulación digital. Divide el ancho de banda disponible en muchas subportadoras estrechas. Cada subportadora se modula de forma independiente con un símbolo de datos. Esta técnica se propuso para combatir el multitrayecto en canales móviles [28]. Hoy en día es la base de la mayoría de estándares inalámbricos modernos: Wi-Fi, LTE, 5G, y también muchos enlaces de vídeo y control de drones.

La clave de OFDM es que las subportadoras son ortogonales entre sí. Esto significa que, aunque sus espectros se solapan parcialmente, no interfieren unas con otras. Gracias a esto, se puede aprovechar el ancho de banda de forma muy eficiente, sin necesidad de dejar espacio de separación entre subportadoras, como sí haría falta en otras técnicas de multiplexación más simples.

En la práctica, la señal OFDM se genera mediante una transformada inversa de Fourier, aplicada sobre el conjunto de símbolos que se quieren transmitir en cada subportadora. Además, se añade un breve prefijo cíclico al inicio de cada símbolo, que sirve para evitar que el canal de transmisión introduzca interferencia entre símbolos consecutivos.

Para este TFG, lo más relevante de una señal OFDM es su comportamiento estadístico cuando hay muchas subportadoras. La señal final resulta de sumar muchas contribuciones independientes, una por cada subportadora. Cuantas más subportadoras hay, más se parece esta suma, en términos estadísticos, a ruido gaussiano [15]. Esta propiedad es clave para el apartado 5.3 de esta memoria. Allí se genera una señal simulada sin construir una cadena OFDM completa. En vez de eso, se genera directamente ruido gaussiano acotado en frecuencia. Así se reproduce el comportamiento estadístico de una señal OFDM real, sin tener que modelar toda su estructura interna.

## 2.4 RUIDO AWGN

AWGN son las siglas de *Additive White Gaussian Noise*, es decir, ruido blanco gaussiano aditivo. Es el modelo de ruido más utilizado en el estudio de sistemas de comunicaciones, y es el que se emplea en los capítulos 7 (Capítulo 7. ) y 8 (Capítulo 8. ) de esta memoria para evaluar el comportamiento de los algoritmos de detección.

El nombre describe tres propiedades del ruido. Es aditivo porque se suma directamente a la señal, sin alterar su forma de ninguna otra manera. Es blanco porque su potencia se reparte de forma uniforme en todas las frecuencias, igual que la luz blanca contiene por igual todos los colores del espectro visible. Y es gaussiano porque sus valores siguen una distribución normal, con media cero.

La señal recibida con ruido se puede expresar de forma sencilla como:

$$r(t) = s(t) + n(t)$$

donde  $s(t)$  es la señal original y  $n(t)$  es el ruido, modelado como un proceso aleatorio gaussiano.

Este modelo es muy habitual en ingeniería de comunicaciones porque representa bien el ruido térmico que aparece de forma natural en cualquier receptor de radiofrecuencia, además de otras fuentes de ruido que, al sumarse, también tienden a comportarse de forma gaussiana. Por eso es el modelo elegido para el análisis del capítulo 8 (Capítulo 8. ): añadiendo ruido AWGN de forma controlada sobre la señal recibida, se puede fijar con precisión la relación señal-ruido (SNR) en cada prueba, y así estudiar cómo se comportan los algoritmos de detección a medida que las condiciones empeoran.

## **2.5 MATLAB Y HERRAMIENTAS DE PROCESAMIENTO DE SEÑAL**

Todo el desarrollo de este TFG se ha realizado en MATLAB, un entorno de cálculo numérico muy extendido en ingeniería y, por tanto, asumido como conocido. Sí conviene mencionar, en cambio, dos herramientas concretas de MATLAB empleadas de forma intensiva a lo largo del proyecto.

La primera es el Communications Toolbox, que proporciona funciones específicas para el procesamiento de señales de comunicaciones. De este toolbox proceden, entre otras, las funciones empleadas para el remuestreo de señales (apartado 5.7) y para el suavizado de espectros mediante media móvil (apartado 7.3.1).

La segunda es el paquete de soporte de hardware para el ADALM-PLUTO (*Communications Toolbox Support Package for ADALM-PLUTO Radio*), que proporciona los objetos `comm.SDRTxPluto` y `comm.SDRRxPluto` empleados en el capítulo 6 (Capítulo 6. ) para transmitir y recibir señales con el dispositivo. Este paquete es el que permite controlar desde MATLAB parámetros como la frecuencia central, la ganancia o la frecuencia de muestreo del Pluto, sin necesidad de programar directamente la FPGA ni de usar otras herramientas de bajo nivel [12].

## Capítulo 3. ESTADO DE LA CUESTIÓN

Antes de abordar el diseño del sistema propuesto en este Trabajo de Fin de Grado, resulta necesario revisar el contexto tecnológico en el que se inscribe. Este capítulo repasa, en primer lugar, las distintas familias de sistemas de detección de drones existentes y la posición que ocupa la detección por radiofrecuencia dentro de ese panorama. A continuación, se revisa el papel de las plataformas de Radio Definida por Software como herramienta de experimentación con señales de RF, y se sitúa el trabajo previo relacionado con la estimación de ocupación espectral, que constituye el fundamento conceptual de los algoritmos de detección desarrollados en el capítulo 7 (Capítulo 7. ). El capítulo se cierra con una síntesis que sitúa este TFG dentro del panorama revisado, dando pie a la justificación detallada del trabajo en el capítulo 4 (Capítulo 4. ).

### ***3.1 SISTEMAS DE DETECCIÓN DE DRONES: PANORAMA TECNOLÓGICO***

El uso de aeronaves no tripuladas (UAV, *Unmanned Aerial Vehicle*) se ha extendido de forma notable en aplicaciones civiles, logísticas y de defensa, lo que ha incrementado en paralelo la preocupación por su detección, identificación y seguimiento en tiempo real [9]. Para abordar este problema se han desarrollado distintas familias de sistemas de detección, clasificables según el principio físico que explotan: sistemas ópticos basados en cámaras de vídeo, sistemas acústicos que analizan la firma sonora de motores y hélices, sistemas térmicos o infrarrojos sensibles al calor emitido por los componentes electrónicos y de propulsión, sistemas radar que iluminan el objetivo con una señal propia y analizan el eco reflejado, y sistemas basados en radiofrecuencia (RF) que interceptan de forma pasiva las emisiones del propio dron.

Cada una de estas tecnologías presenta limitaciones propias. Los sistemas ópticos dependen de condiciones de visibilidad e iluminación favorables, y su alcance se reduce considerablemente en condiciones meteorológicas adversas o durante la noche. Los sistemas acústicos son muy sensibles al ruido ambiental, lo que reduce su fiabilidad en entornos urbanos o industriales. Los sistemas térmicos solo resultan eficaces frente a drones con motores de combustión, una configuración minoritaria entre los multicopteros civiles actuales. Y el radar, aunque consolidado para aeronaves de gran tamaño, encuentra dificultades para detectar plataformas pequeñas fabricadas en materiales plásticos o compuestos, además de requerir equipos de coste e infraestructura considerables. Una revisión más amplia de estas distintas familias de detección, incluyendo enfoques recientes basados en aprendizaje automático, puede encontrarse en [12].

Frente a estas limitaciones, la detección por RF se ha consolidado como una de las alternativas más prácticas, y es la que se adopta como punto de partida en este TFG, según se desarrolla en el apartado siguiente.

### **3.2 DETECCIÓN BASADA EN RADIOFRECUENCIA**

La detección por RF se fundamenta en una idea sencilla: prácticamente la totalidad de los drones comerciales necesita un enlace bidireccional de radiofrecuencia para operar, compuesto típicamente por un canal de mando y control (C2, *Command and Control*) en sentido estación-dron y un canal de telemetría y/o vídeo en sentido dron-estación. Ambos enlaces deben emitir energía electromagnética de forma continua o periódica durante el vuelo, lo que convierte al propio dron en una fuente identificable [8], [9]. A diferencia del radar o de los sistemas ópticos, la detección por RF no requiere iluminar al objetivo ni disponer de línea de visión directa, ni depende de las dimensiones físicas o del material de la aeronave, lo que explica su idoneidad como técnica de bajo coste y amplia cobertura [8].

La mayoría de los fabricantes emplea bandas de uso libre, fundamentalmente la banda ISM de 2,4 GHz y, en menor medida, la de 5,8 GHz. Recurren a esquemas de modulación y acceso al medio diversos, entre ellos técnicas de salto en frecuencia (FHSS, *Frequency Hopping Spread Spectrum*) y esquemas de modulación multiportadora de tipo OFDM, según el fabricante y el modelo concreto [7], [8]. En el caso particular de los drones estudiados en este TFG (M250, V2, OcuSync 2, Mini 2 y DJI Neo), la bibliografía consultada confirma la existencia de protocolos propietarios de mando y control poco o nada cifrados [7] y vulnerables a su caracterización mediante SDR [8]. Trabajos previos, como el de Bello [11], han desarrollado herramientas de detección y clasificación de drones basadas en RF, aunque sin profundizar en la caracterización experimental de las formas de onda concretas de cada protocolo (separación entre subportadoras, número de puntos de FFT, prefijo cíclico, patrón de salto, etc.). La determinación experimental de estas características a partir de las capturas reales facilitadas por el director del proyecto constituye, precisamente, uno de los objetivos centrales de este trabajo, y se aborda mediante análisis espectral y comparación con señales OFDM generadas de forma sintética en los capítulos 5 (Capítulo 5. ) y 8 ( Capítulo 8. ) de esta memoria.

Desde el punto de vista del procesamiento digital de señal, la detección por RF se apoya generalmente en la estimación de la densidad espectral de potencia (PSD) de la señal recibida y en la comparación de la energía presente en una banda de interés frente a un umbral de decisión, principio básico de los detectores de energía empleados habitualmente en sistemas de monitorización espectral [13]. Este principio inspira directamente los algoritmos de detección desarrollados en este TFG, descritos con detalle en el apartado 3.4 y en el capítulo 7 (Capítulo 7. ).

### ***3.3 PLATAFORMAS SDR DE BAJO COSTE PARA EXPERIMENTACIÓN CON SEÑALES DE RF***

Cualquier sistema de detección de drones basado en RF necesita, durante su desarrollo, exponerse a señales de referencia conocidas y reproducibles, ya sea para ajustar sus parámetros, comparar distintos algoritmos de detección entre sí, o simplemente comprobar su comportamiento ante condiciones de ruido controladas. Tradicionalmente, este tipo de pruebas se ha apoyado en generadores de señal y analizadores de espectro de banco, instrumentos fiables pero de coste elevado y limitada flexibilidad para reproducir formas de onda complejas o realistas [2].

La Radio Definida por Software ha abierto una alternativa considerablemente más económica y configurable para esta misma función. El paradigma SDR traslada al dominio software funciones que tradicionalmente se implementaban en hardware dedicado, de manera que basta modificar el código de control para generar, reproducir o analizar formas de onda completamente distintas sin alterar el hardware subyacente [4], [5]. Esta flexibilidad ha impulsado el uso de plataformas SDR de bajo coste en proyectos de investigación y docencia, permitiendo replicar funcionalidades antes exclusivas de equipos de laboratorio de gama alta [2].

Entre las plataformas SDR de bajo coste orientadas a uso académico, el módulo ADALM-PLUTO de Analog Devices, descrito en detalle en el apartado 2.2, destaca por su sencillez de integración y por su compatibilidad nativa con MATLAB y Simulink [1]. Diversos trabajos previos han empleado plataformas SDR de bajo coste para construir bancos de pruebas reconfigurables en otros ámbitos de las comunicaciones, como la caracterización de prestaciones del propio ADALM-PLUTO frente a otros estándares de comunicaciones, por ejemplo IEEE 802.15.4 [10], estudios de propagación radioeléctrica [2], o la construcción de bancos de pruebas reconfigurables para investigación en seguridad [3]. Estos antecedentes respaldan la viabilidad de extender este mismo enfoque al ámbito específico de la generación, transmisión y recepción de señales de dron, como se hace en el capítulo 6 (Capítulo 6. ) de esta memoria.

### ***3.4 ESTIMACIÓN DE OCUPACIÓN ESPECTRAL Y DETECCIÓN DE BORDES DE BANDA***

El área de investigación más cercana a los algoritmos de este TFG es la del *spectrum sensing*, estudiada en el contexto de la radio cognitiva para determinar qué partes del espectro están ocupadas [29]. La técnica más extendida es la detección de energía, propuesta por Urkowitz [31], que compara la energía de una banda frente a un umbral de decisión sin conocer a priori la forma de onda de la señal.

Más relevante aún es el trabajo de Tian y Giannakis [30], que localiza los bordes de una banda ocupada a partir de los picos de la derivada de la densidad espectral de potencia suavizada mediante transformada wavelet. Esta idea, derivar un espectro suavizado y localizar sus picos como bordes, es el principio en el que se basa el Algoritmo 1 del capítulo 7 (Capítulo 7. ), aunque aquí se aplica mediante una media móvil en lugar de wavelets, y se orienta a localizar los dos límites de frecuencia de una señal conocida, no a decidir la ocupación de múltiples subbandas.

La mayoría de los trabajos de *spectrum sensing* [29] abordan una decisión binaria (banda ocupada o libre), no la estimación precisa de los bordes de una señal que se sabe presente. Esta diferencia de enfoque se retoma en el apartado siguiente.

### 3.5 SÍNTESIS Y POSICIONAMIENTO DE ESTE TFG

De la revisión anterior se desprenden varias conclusiones que sitúan el enfoque de este Trabajo de Fin de Grado. En primer lugar, la detección por RF es, dentro del panorama tecnológico disponible, la alternativa más adecuada para identificar drones comerciales de pequeño tamaño, al no depender de condiciones ambientales favorables ni de equipos de coste elevado (apartados 3.1 y 3.2). En segundo lugar, las plataformas SDR de bajo coste, y en particular el ADALM-PLUTO, constituyen una herramienta consolidada para generar, transmitir y analizar señales de radiofrecuencia en proyectos de investigación, lo que las hace especialmente adecuadas para experimentar con señales de dron de forma controlada y reproducible (apartado 3.3). En tercer lugar, la literatura de *spectrum sensing* aporta el fundamento teórico de los algoritmos de detección espectral, pero se centra mayoritariamente en la decisión binaria de ocupación de banda, dejando un hueco en la estimación precisa de los bordes de frecuencia de una señal conocida (apartado 3.4).

Este TFG se sitúa precisamente en ese espacio. Combina la caracterización espectral de señales reales de drones con la generación de señales OFDM simuladas comparables, desarrolla dos algoritmos propios de detección de bordes de banda inspirados en los principios de detección de energía y de localización de bordes mediante derivadas, y valida experimentalmente todo el conjunto mediante transmisión y recepción real con un dispositivo ADALM-PLUTO. El resto de esta memoria desarrolla este planteamiento: el capítulo 4 (Capítulo 4. ) detalla los objetivos y la metodología del trabajo, y los capítulos 5 (Capítulo 5. ) a 8 (Capítulo 8. ) describen, respectivamente, la generación de señales, la plataforma SDR, los algoritmos de detección y los resultados experimentales obtenidos.

## Capítulo 4. DEFINICIÓN DEL TRABAJO

Este capítulo concreta, a partir de la revisión realizada en el capítulo 3 (Capítulo 3. ), las razones que justifican este Trabajo de Fin de Grado, los objetivos concretos que persigue, la metodología seguida para alcanzarlos y la planificación temporal y económica del proyecto.

### 4.1 JUSTIFICACIÓN

El capítulo 3 (Capítulo 3. ) ha puesto de manifiesto tres carencias concretas en el estado actual de la cuestión. Primero, aunque la detección por RF es la alternativa más práctica para identificar drones comerciales de pequeño tamaño, la bibliografía consultada sobre los drones estudiados en este TFG no proporciona una caracterización pública y detallada de los parámetros de forma de onda de sus enlaces de mando, control y vídeo. Segundo, la literatura de *spectrum sensing* aporta un marco teórico sólido para la detección de energía espectral, pero se centra mayoritariamente en decidir si una banda está ocupada o libre, sin abordar la estimación precisa de los bordes de frecuencia de una señal que se sabe presente. Tercero, no se ha encontrado documentación que combine, dentro de un mismo flujo de trabajo reproducible, la caracterización de señales reales de drones, la generación de señales simuladas comparables y la validación experimental de algoritmos de detección mediante hardware SDR de bajo coste.

Estas carencias justifican técnicamente este proyecto. Frente a la opción de depender de equipos de laboratorio especializados y costosos, o de asumir sin verificación experimental que las señales de drones se comportan como señales OFDM ideales, este TFG propone una alternativa accesible: emplear un dispositivo ADALM-PLUTO, de coste reducido y plenamente integrado en MATLAB, para caracterizar espectralmente señales reales de drones DJI, compararlas con señales OFDM generadas de forma sintética, y desarrollar y validar experimentalmente algoritmos propios capaces de estimar los bordes de banda de una señal conocida en presencia de ruido.

Desde un punto de vista más amplio, el contexto de proliferación de drones comerciales descrito en el capítulo 1 (Capítulo 1. ), junto con la creciente necesidad de sistemas de detección accesibles que no dependan de infraestructuras costosas, refuerza el interés práctico de demostrar que una plataforma SDR de bajo coste puede emplearse no solo con fines docentes, sino para abordar un problema de detección y caracterización con aplicación real.

## **4.2 METODOLOGÍA**

El objetivo general de este Trabajo de Fin de Grado es desarrollar un sistema de detección y caracterización de señales RF de drones mediante Radio Definida por Software, que combine el análisis de señales reales y simuladas con algoritmos propios de estimación espectral, validado experimentalmente mediante un dispositivo ADALM-PLUTO.

Este objetivo general se concreta en los siguientes objetivos específicos:

1. Analizar y caracterizar espectralmente las señales reales de los cinco drones de referencia empleados en este trabajo (M250, V2, OcuSync 2, Mini 2 y DJI Neo), a partir de las capturas facilitadas por el director del proyecto.
2. Generar y estudiar señales OFDM simuladas comparables a las señales reales, justificando los parámetros de generación elegidos (número de subportadoras, ancho de banda, frecuencia de muestreo, entre otros).
3. Desarrollar dos algoritmos propios de detección espectral, capaces de estimar los bordes de banda (frecuencias inferior y superior) de una señal en presencia de ruido controlado.
4. Comparar cuantitativamente las señales simuladas y las señales reales, evaluando su similitud estadística y las diferencias relevantes para el comportamiento de los algoritmos de detección.

5. Validar experimentalmente el sistema completo, transmitiendo y recibiendo señales reales y simuladas mediante el dispositivo ADALM-PLUTO, y evaluando el comportamiento de los algoritmos de detección desarrollados bajo distintas condiciones de relación señal-ruido (SNR).

### 4.3 PLANIFICACIÓN Y ESTIMACIÓN ECONÓMICA

#### 4.3.1 PLANIFICACIÓN TEMPORAL

El proyecto se ha desarrollado a lo largo del curso académico 2025-2026, desde el 1 de septiembre de 2025 hasta la defensa final, prevista para el 30 de junio de 2026, compaginándose con el resto de asignaturas del curso y con los periodos de exámenes de diciembre, mayo y la convocatoria de recuperación de junio. La siguiente tabla resume las fases en las que se ha organizado el trabajo, con una estimación de horas semanales de dedicación ajustada a la carga real de cada periodo:

| <i>Tramo</i> | <i>Periodo</i>         | <i>Horas</i> |
|--------------|------------------------|--------------|
| T1           | Septiembre - noviembre | 60 h         |
| T2           | Diciembre              | 8 h          |
| T3           | Enero - marzo          | 96 h         |
| T4           | 1 - 7 de abril         | 30 h         |
| T5           | 7 - 30 de abril        | 50 h         |
| T6           | Mayo                   | 40 h         |
| T7           | 1 - 24 de junio        | 75 h         |
| T8           | 24 - 30 de junio       | 15 h         |
| <b>Total</b> |                        | <b>374 h</b> |

*Tabla 1: Distribución temporal de horas dedicadas al proyecto por tramos.*

Descripción de cada tramo:

- **T1:** reuniones con el director, revisión de las señales de referencia, búsqueda bibliográfica e investigación sobre señales OFDM, primeras pruebas con el ADALM-PLUTO.
- **T2:** periodo de exámenes, dedicación mínima al proyecto.
- **T3:** desarrollo del primer algoritmo de detección, pruebas iniciales con el ADALM-PLUTO en corto (sin transmisión real por RF).
- **T4:** cierre del primer algoritmo, resultados para la presentación de seguimiento.
- **T5:** desarrollo del segundo algoritmo, depuración y mejora del primero, inicio de la transmisión real de señales OFDM con el ADALM-PLUTO.
- **T6:** periodo de exámenes, manteniendo en paralelo pruebas experimentales e inicio de la redacción de la memoria.
- **T7:** pruebas finales con los dos ADALM-PLUTO y los cables SMA, obtención de resultados, redacción intensiva de la memoria (coincide con la recuperación de junio).
- **T8:** preparación de la presentación para la defensa.

*Nota: estimación reconstruida a partir de la descripción real del desarrollo del proyecto, ajustando la intensidad de trabajo a los periodos de exámenes y a los hitos marcados (seguimiento del 7 de abril, entrega de memoria el 24 de junio, defensa el 30 de junio), al no disponer de un registro horario exacto.*

#### 4.3.2 ESTIMACIÓN ECONÓMICA

El presupuesto de este TFG se ha calculado siguiendo la práctica habitual en proyectos de ingeniería, incluyendo todos los recursos empleados a su valor de mercado de referencia, independientemente de si han supuesto un coste directo para el alumno. Esto permite reflejar el coste real que tendría replicar el proyecto fuera de un contexto universitario, aunque en este caso concreto el hardware y el software hayan sido proporcionados por la universidad y por el director del proyecto.

| Concepto                                 | Coste de referencia | Coste real  |
|------------------------------------------|---------------------|-------------|
| 2 × ADALM-PLUTO                          | 400 €               | 0 €         |
| Cables SMA                               | 15 €                | 15 €        |
| Ordenador                                | No se imputa        | 0 €         |
| Licencia MATLAB + Communications Toolbox | 1.700 €/año         | 0 €         |
| Horas de ingeniería (374 h × 15 €/h)     | 5.610 €             | No aplica   |
| <b>Total</b>                             | <b>≈ 7.725 €</b>    | <b>15 €</b> |

Tabla 2: Estimación económica del proyecto.

Notas:

- Los dos ADALM-PLUTO fueron cedidos por el director del proyecto; el coste de referencia (200 €/unidad) corresponde al precio oficial de mercado.
- El ordenador empleado ya estaba disponible, por lo que no se imputa coste adicional.
- La licencia de MATLAB y Communications Toolbox corresponde a la licencia Campus de la universidad; el coste de referencia es el precio individual anual de mercado.
- La tarifa de ingeniería (15 €/h) se basa en el sueldo medio de un ingeniero junior de telecomunicaciones en España (~23.250 €/año, fuente: Glassdoor), aplicando el recargo estándar de coste de empresa.
- El único coste real desembolsado para este TFG ha sido el de los cables SMA (15 €).

Como puede observarse, el coste real desembolsado para la realización de este TFG ha sido mínimo (15 €, correspondientes a los cables SMA), gracias a la disponibilidad de hardware y software facilitados por la universidad y el director del proyecto. El coste de referencia total (≈ 7.725 €), que incluye el valor de mercado del hardware, la licencia de software y las horas de ingeniería dedicadas, refleja el coste que tendría desarrollar un proyecto equivalente en un contexto profesional sin acceso a estos recursos institucionales.

## **Capítulo 5. OBTENCIÓN, PROCESADO Y**

### **PREPARACIÓN DE SEÑALES RF**

Este capítulo recoge la primera etapa del desarrollo del sistema de detección y caracterización de señales RF de drones objeto de este Trabajo de Fin de Grado. Partiendo de las señales reales facilitadas por el director del proyecto, se realiza inicialmente un análisis espectral que permite caracterizar el comportamiento en frecuencia de cada una de ellas. A partir de dicha caracterización, se generan señales simuladas de banda limitada que reproducen de forma aproximada las propiedades espectrales observadas, permitiendo comparar el comportamiento de las señales reales y las señales simuladas.

Posteriormente, se describe el procesado aplicado a las señales reales con el fin de acotar su contenido espectral y adaptar su amplitud para facilitar su análisis. Asimismo, se presenta el procedimiento seguido para unificar todas las señales, tanto reales como simuladas, en una estructura de datos común que simplifica su tratamiento posterior.

Finalmente, se aborda la adaptación de la frecuencia de muestreo de las señales al valor soportado por el ADALM-PLUTO, paso necesario para su posterior transmisión y recepción mediante este dispositivo, aspectos que se desarrollan en el capítulo siguiente.

#### **5.1 CARACTERIZACIÓN ESPECTRAL DE LAS SEÑALES REALES**

Como punto de partida del desarrollo experimental, se dispuso de un conjunto de capturas reales correspondientes a cinco tipos de señales de dron, facilitadas por el director del proyecto. Para cada tipo de señal se proporcionaron tres capturas independientes, dando lugar a un total de quince registros empleados durante las distintas fases de análisis, simulación y validación del sistema desarrollado.

Cuatro de los conjuntos de datos presentan una estructura homogénea formada por los campos *fc*, *Fs* y *signal*, donde *fc* representa la frecuencia central de captura, *Fs* la frecuencia de muestreo, igual a 92,16 MHz, y *signal* un vector de  $N = 1.441.792$  muestras complejas en banda base.

El quinto conjunto de datos, correspondiente al dron M250, presenta un formato diferente, compuesto por los campos *fc*, *pxxden* y *raw\_data*. En este caso, *pxxden* contiene una estimación previa de la densidad espectral de potencia, mientras que *raw\_data* almacena la señal compleja en bruto y posee la misma longitud que las restantes capturas, es decir,  $N = 1.441.792$  muestras.

Con el fin de simplificar el procesamiento posterior y disponer de una estructura de datos homogénea para todas las señales, las capturas correspondientes al M250 fueron adaptadas al mismo formato empleado por el resto de drones. Para ello, la señal almacenada en el campo *raw\_data* se reasignó al campo *signal* y se incorporó explícitamente el campo *Fs*, fijando su valor en 92,16 MHz. De este modo, todas las capturas quedaron representadas mediante los campos *fc*, *Fs* y *signal*, permitiendo el desarrollo de algoritmos de análisis, simulación y caracterización independientes del origen de los datos.

El campo *fc*, presente en todos los conjuntos de datos, no se emplea en esta primera etapa del análisis. Su utilización se reserva para fases posteriores del trabajo en las que sea necesario expresar los resultados en términos de frecuencia absoluta.

### 5.1.1 FUNDAMENTO TEÓRICO

Para visualizar el contenido en frecuencia de las señales de dron se empleó la Transformada Discreta de Fourier (DFT), calculada mediante el algoritmo rápido de Fourier (FFT):

$$X[k] = \sum_{n=0}^{N-1} x[n] e^{-j2\pi kn/N}, k = 0, 1, \dots, N-1$$

La FFT transforma la señal desde el dominio temporal al dominio frecuencial, permitiendo observar cómo se distribuye su energía en las distintas frecuencias.

Posteriormente, se aplicó la función `fftshift`, que reordena las componentes espectrales de forma que la frecuencia nula quede situada en el centro de la representación. De este modo, el eje de frecuencias asociado a cada bin viene dado por

$$f[k] = \left(-\frac{N}{2} + k\right) \frac{f_s}{N}.$$

El espectro representado corresponde al módulo de la FFT,

$$S[k] = |X[k]|$$

obtenido a partir del registro completo de la señal. En esta primera etapa no se aplicaron técnicas de estimación espectral más avanzadas, como el promediado de Welch o el cálculo de densidades espectrales de potencia. El objetivo de este análisis es únicamente disponer de una primera representación del espectro de las señales y observar de forma cualitativa su ocupación en frecuencia y su forma espectral.

### **5.1.2 REPRESENTACIÓN ESPECTRAL DE LAS SEÑALES DE DRON**

Con el fin de obtener una primera visión del comportamiento espectral de las señales empleadas en el proyecto, se calculó el espectro de magnitud de las tres capturas disponibles para cada tipo de dron. A continuación, se presentan las representaciones espectrales obtenidas para cada una de las señales analizadas.

### 5.1.2.1 Señal Mini 2

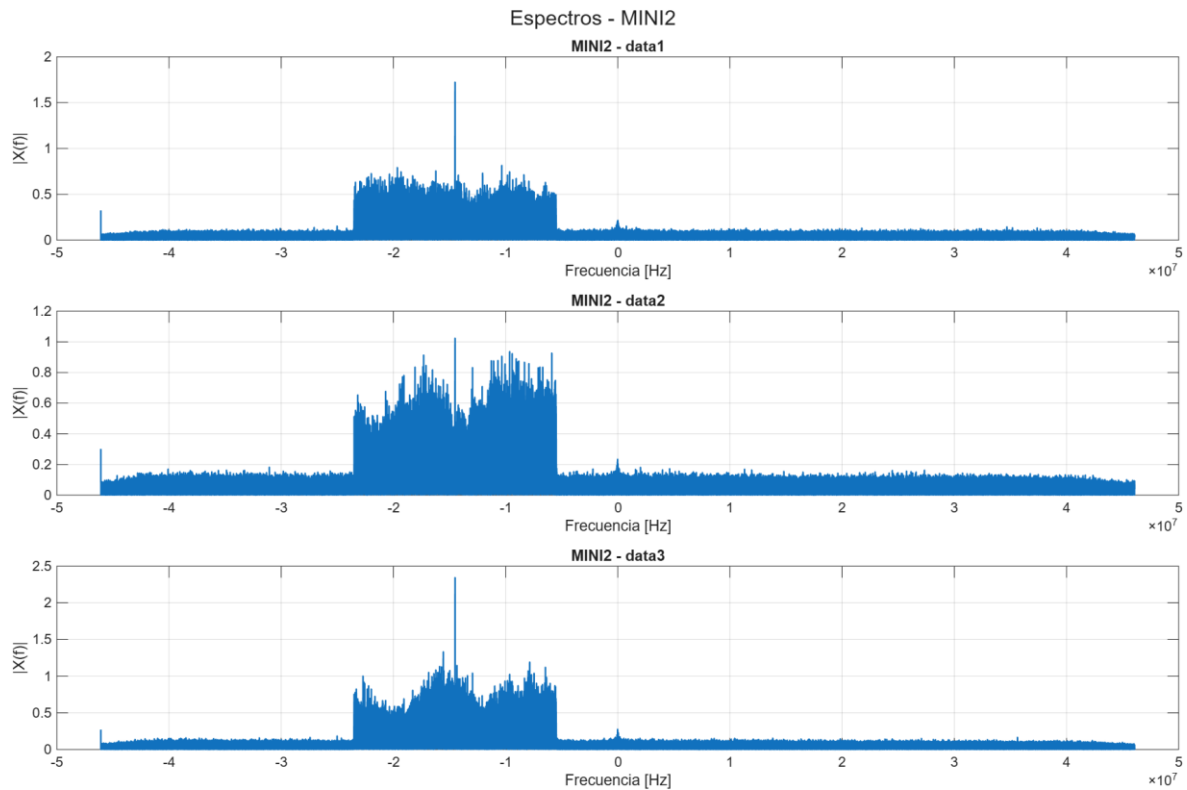


*Figura 4: Espectros correspondientes a las tres capturas del dron Mini 2.*

La banda ocupada se mantiene consistente en las tres capturas, situándose aproximadamente entre -23 y -5,5 MHz. En los tres casos se observa un pico estrecho recurrente en torno a -15 MHz, lo que sugiere que dicha componente podría corresponder a un elemento estructural de la señal, como una subportadora o un tono de referencia, y no a una variación aleatoria entre capturas. Se aprecia asimismo un pequeño repunte de magnitud en torno a 0 Hz en las tres capturas, compatible con una componente residual de continua o con una posible fuga del oscilador local en la cadena de recepción, así como un pico puntual cerca del extremo del eje de frecuencias que podría asociarse a un artefacto de filtrado próximo a la frecuencia de Nyquist. La segunda captura (data2) presenta un nivel de ruido de fondo superior al de las otras dos, lo que indica una relación señal-ruido más desfavorable en dicha adquisición.

### 5.1.2.2 Señal DJI Neo

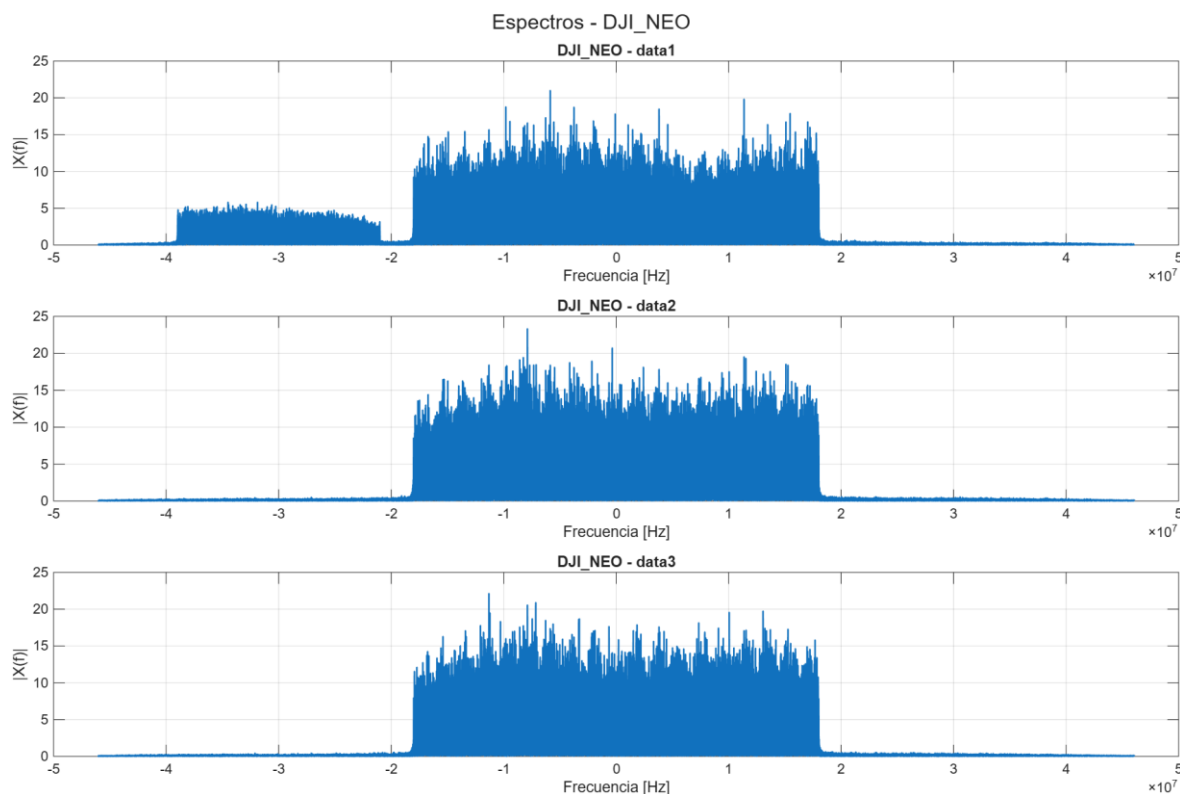


*Figura 5: Espectros correspondientes a las tres capturas del dron DJI Neo.*

Las tres capturas presentan una banda principal consistente, aproximadamente entre -18 y +18 MHz, con una forma espectral densa e irregular, propia de una señal digital de banda ancha. No obstante, una de las tres capturas presenta además una segunda banda de menor amplitud, en torno a -39 a -21 MHz, que no aparece en las otras dos. Esta discrepancia entre capturas de la misma señal constituye una evidencia directa de la necesidad del criterio de selección de muestra representativa descrito anteriormente, dado que la inclusión de dicha banda adicional en un análisis conjunto introduciría una componente no representativa del comportamiento habitual de la señal.

### 5.1.2.3 Señal M250

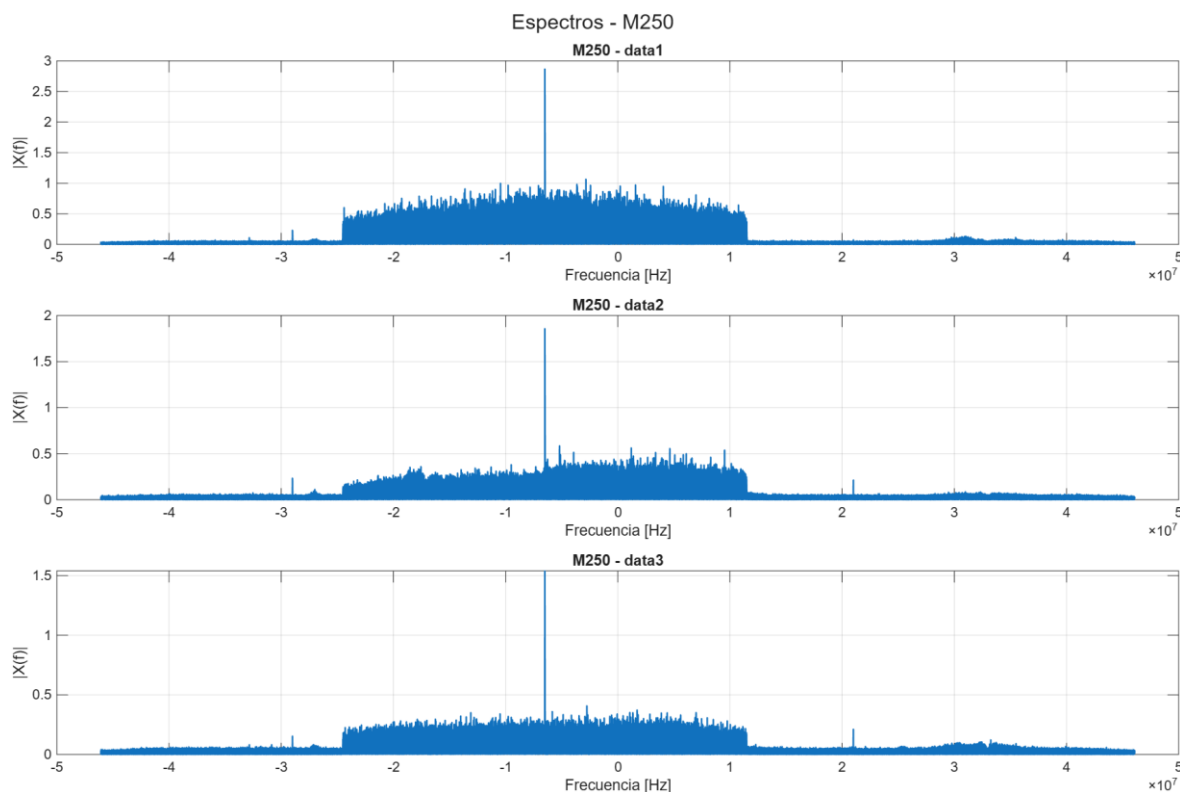


*Figura 6: Espectros correspondientes a las tres capturas del dron M250.*

Las tres capturas comparten una misma forma espectral: un pico estrecho y dominante situado en torno a -6,5 MHz, superpuesto a un nivel de ruido de banda ancha que se extiende aproximadamente entre -25 y +11,5 MHz. La amplitud de dicho pico decrece de forma progresiva entre las tres capturas, lo que resulta compatible con una variación en la relación señal-ruido de la adquisición más que con una diferencia en el contenido espectral de la señal.

### 5.1.2.4 Señal OcuSync 2

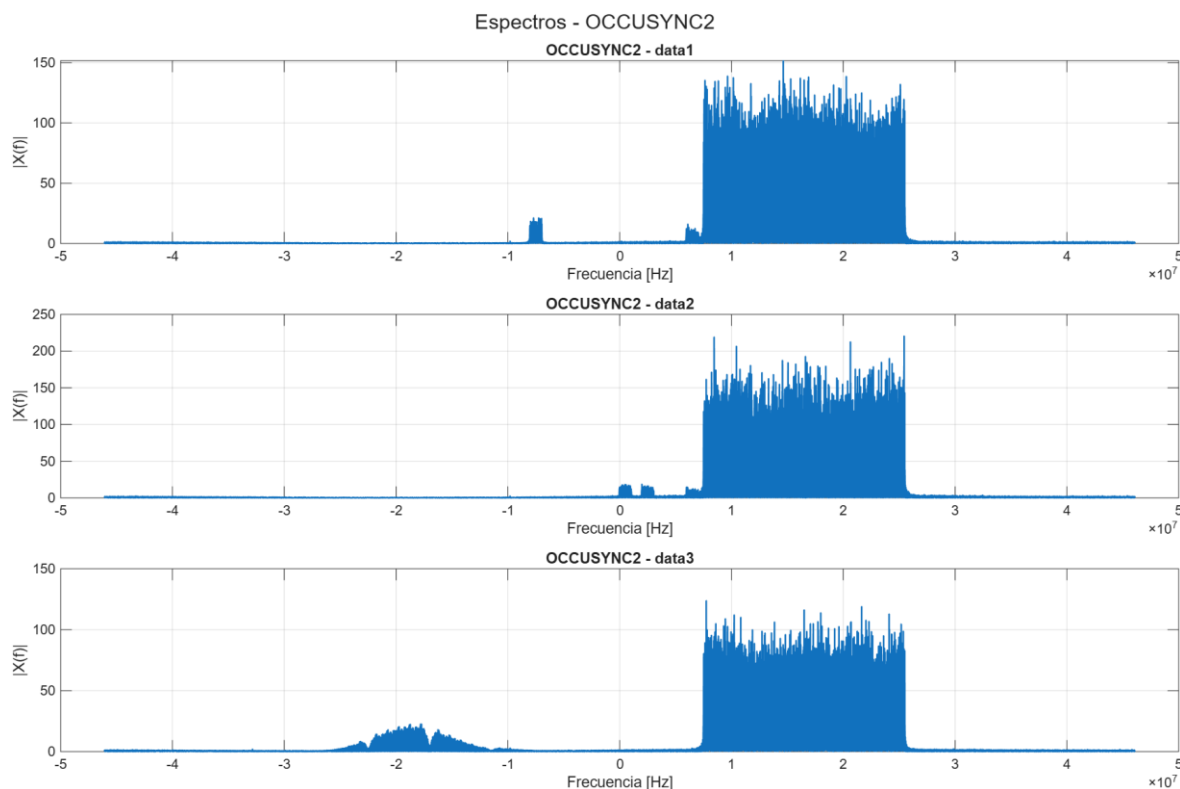


*Figura 7: Espectros correspondientes a las tres capturas del dron OcuSync 2.*

La banda principal, situada aproximadamente entre +7,5 y +25,5 MHz, resulta prácticamente idéntica en forma y amplitud relativa en las tres capturas, y presenta además el nivel de magnitud más elevado de las cinco señales analizadas, lo que es indicativo de una recepción con una relación señal-ruido favorable. Cada captura muestra, no obstante, una componente secundaria de baja amplitud situada en una posición distinta del espectro, sin que dicha posición se repita entre capturas, lo que sugiere un origen ajeno a la propia señal del dron, probablemente asociado a interferencias del entorno de captura.

### 5.1.2.5 Señal V2

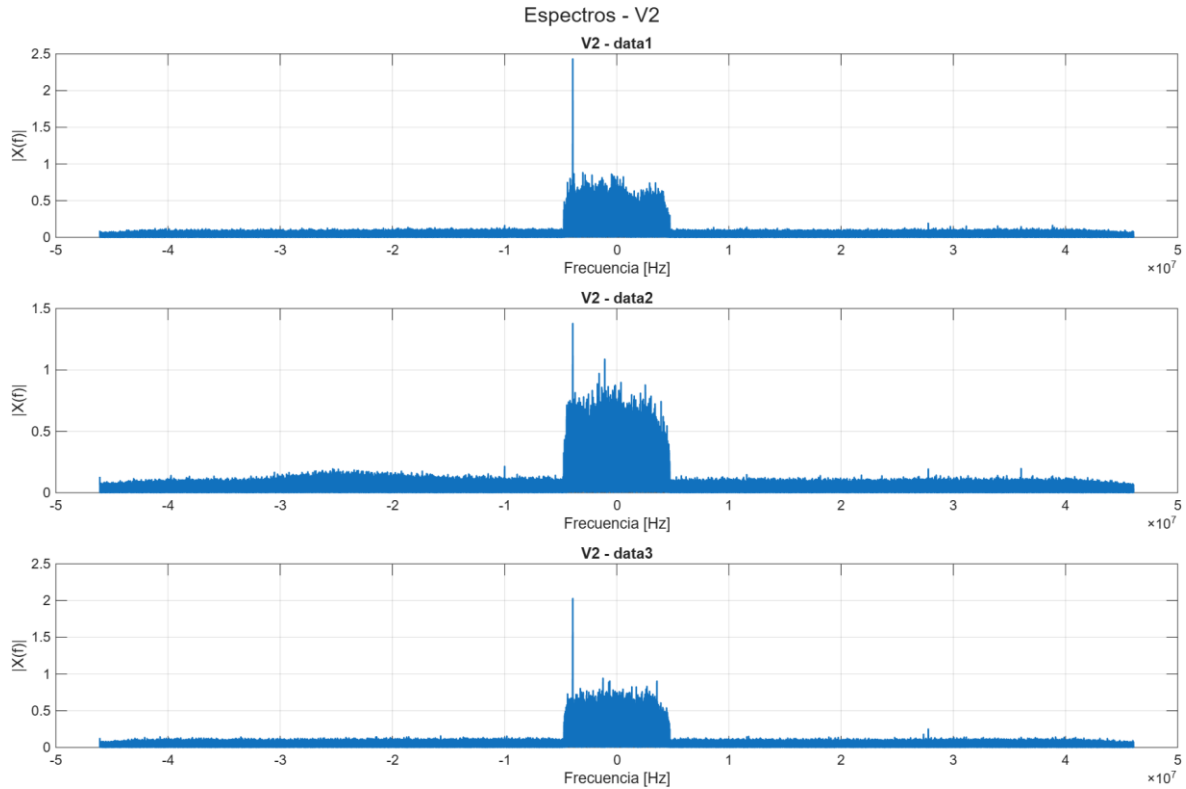


*Figura 8: Espectros correspondientes a las tres capturas del dron V2.*

Las tres capturas presentan una banda estrecha y consistente en torno a  $\pm 5$  MHz, con un tono dominante claramente marcado en el centro de la banda en los tres casos. Se trata de la señal con menor ancho de banda ocupado de las cinco analizadas. Al igual que en la señal OcuSync 2, se observan picos puntuales aislados fuera de la banda principal que varían de posición entre capturas, lo que apunta de nuevo a un origen externo a la propia señal más que a una característica intrínseca de esta.

Tras el análisis individual de las tres capturas disponibles para cada señal, se seleccionó una de ellas como muestra representativa para su empleo en las comparaciones conjuntas de los apartados siguientes. El criterio de selección fue cualitativo, basado en la inspección visual de las tres capturas de cada señal y la elección de aquella que presentaba un espectro más limpio, sin artefactos evidentes ni componentes ajenas a la banda principal identificada; no se empleó una métrica cuantitativa, como la relación señal-ruido, para fundamentar esta elección, lo que se señala como una limitación de esta etapa exploratoria del análisis. La Figura 9 recoge, de forma conjunta, el espectro de la muestra seleccionada para cada una de las cinco señales analizadas.

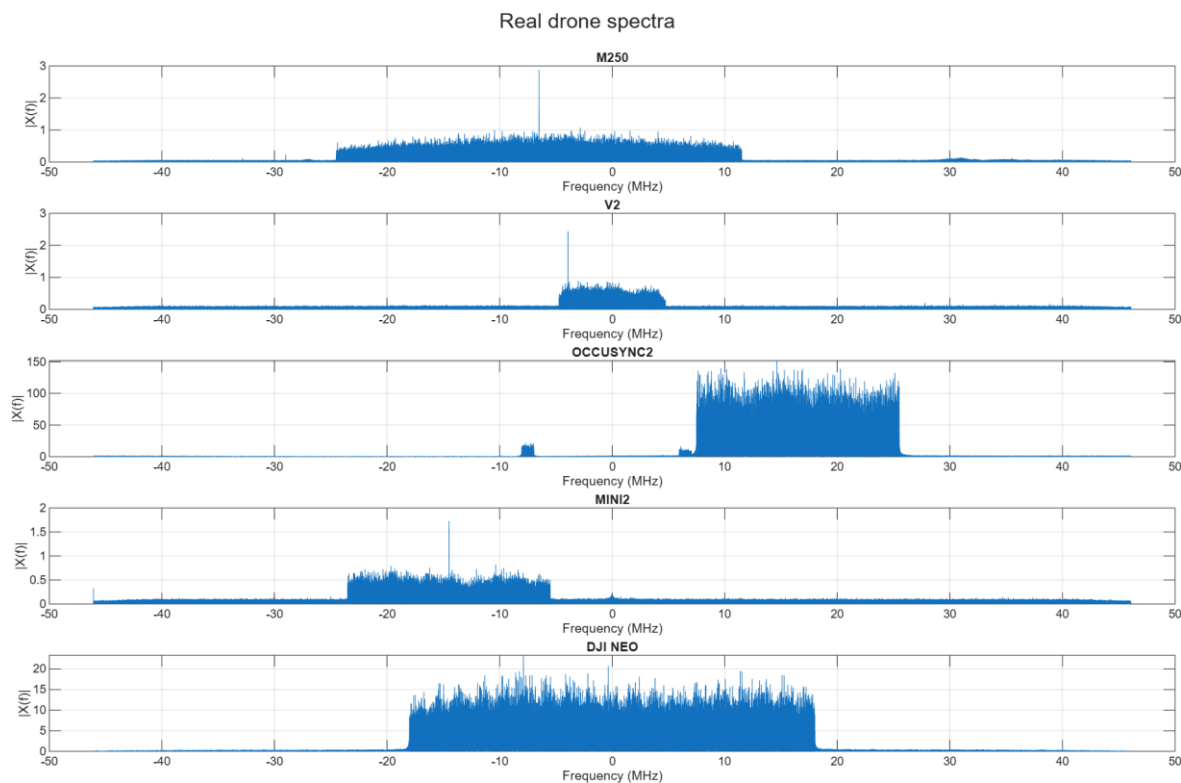


*Figura 9: Comparación de los espectros de magnitud de la muestra representativa seleccionada para cada una de las cinco señales analizadas.*

## 5.2 ESTIMACIÓN DE LOS PARÁMETROS DE BANDA DE LAS SEÑALES REALES

### 5.2.1 FUNDAMENTO TEÓRICO

A partir de los límites de banda inferior y superior identificados para cada señal, se calculan dos parámetros característicos: la frecuencia central y el ancho de banda ocupado.

$$f_c = \frac{f_l + f_h}{2}$$

$$BW = f_h - f_l$$

### 5.2.2 PROCEDIMIENTO

Los límites de banda ( $f_l$ ,  $f_h$ ) y el nivel de amplitud característico de cada señal se determinaron mediante inspección visual de los espectros presentados en el apartado anterior, identificando para cada una el rango de frecuencias en el que se concentra de forma clara su energía, así como un nivel de magnitud representativo dentro de dicho rango. No se empleó, por tanto, un criterio automático o un umbral cuantitativo para esta estimación, lo que se señala como una limitación de esta etapa exploratoria.

### 5.2.3 RESULTADOS

| <i>Señal</i>     | <i><math>f_l</math>(MHz)</i> | <i><math>f_h</math>(MHz)</i> | <i><math>f_c</math>(MHz)</i> | <i>BW (MHz)</i> | <i>Amplitud</i> |
|------------------|------------------------------|------------------------------|------------------------------|-----------------|-----------------|
| <b>M250</b>      | -24,50                       | 11,51                        | -6,50                        | 36,01           | 0,7             |
| <b>V2</b>        | -4,70                        | 4,73                         | 0,02                         | 9,43            | 0,5             |
| <b>OcuSync 2</b> | 7,50                         | 25,50                        | 16,50                        | 18,00           | 80              |
| <b>Mini 2</b>    | -23,51                       | -5,50                        | -14,51                       | 18,01           | 0,5             |
| <b>DJI Neo</b>   | -18,00                       | 18,00                        | 0,00                         | 36,00           | 10              |

*Tabla 3: Parámetros de banda estimados para cada una de las señales reales analizadas.*

Los valores de  $f_c$  obtenidos coinciden, en los cinco casos, con la posición del pico o tono dominante descrito cualitativamente en el apartado anterior, lo que respalda la coherencia entre ambos análisis. En cuanto al ancho de banda, se observan diferencias notables entre señales: M250 y DJI Neo ocupan una banda considerablemente más amplia (36 MHz) que OcuSync 2 y Mini 2 (18 MHz), mientras que V2 presenta el ancho de banda más estrecho (9,43 MHz). Estas diferencias son coherentes con lo observado visualmente y sugieren que cada señal corresponde a un esquema de transmisión con requisitos de ancho de banda distintos.

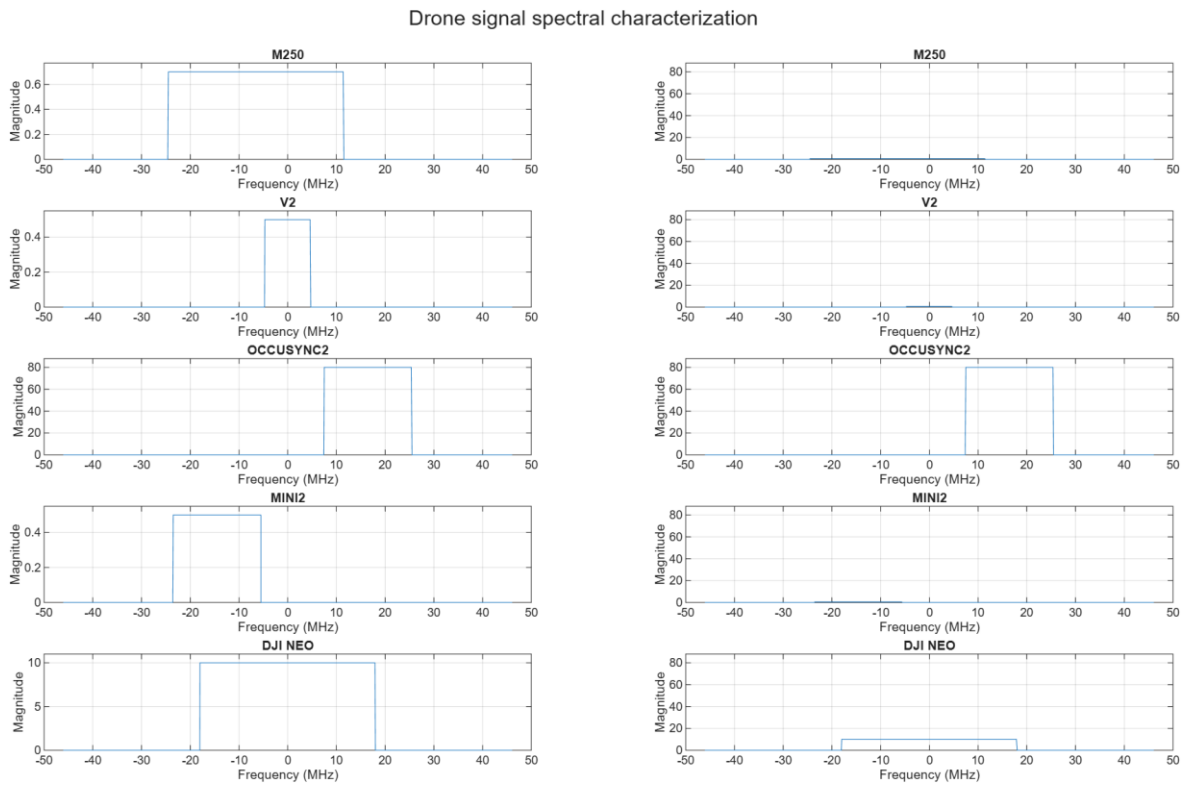


*Figura 10: Máscara espectral rectangular idealizada para cada señal, a partir de los parámetros de banda estimados, con escala de magnitud individual (izquierda) y escala común (derecha).*

A partir de estos parámetros se construye, para cada señal, una máscara espectral rectangular idealizada: un bloque de magnitud constante igual a la amplitud característica, confinado al intervalo  $[f_l, f_h]$  y nulo fuera de él. Esta representación se muestra en dos versiones complementarias: una con escala de magnitud individual para cada señal, que permite apreciar con claridad la forma y anchura relativa de cada banda, y otra con una escala común a las cinco señales (fijada a partir de la amplitud máxima, correspondiente a OcuSync 2), que permite comparar visualmente sus niveles de amplitud relativos. Conviene insistir en que esta máscara rectangular es un modelo simplificado con fines de comparación y de generación de señales simuladas (apartado siguiente), y no una representación fiel del espectro real, que, como se mostró anteriormente, presenta estructura interna, picos y componentes adicionales dentro de cada banda.

### **5.3 GENERACIÓN DE SEÑALES OFDM SIMULADAS**

En el apartado anterior se observó que las cinco señales reales analizadas presentan, dentro de su banda ocupada, un espectro denso y continuo, sin huecos significativos, más parecido al de una señal de banda ancha que al de un tono aislado. Este tipo de comportamiento espectral es característico de las técnicas de modulación digital habituales en enlaces de comunicación de banda ISM (2,4 GHz), entre las que se encuentran los esquemas multiportadora como OFDM. En una señal OFDM real, la información se reparte entre muchas subportadoras moduladas de forma independiente; el resultado de combinarlas es una señal cuya energía aparece distribuida de forma bastante uniforme dentro de la banda ocupada, sin un patrón evidente, en lugar de concentrarse en una única frecuencia. De hecho, está demostrado que, al combinar un número elevado de subportadoras moduladas de forma independiente, la señal resultante en el dominio temporal tiende, por el teorema central del límite, a una distribución estadística gaussiana, comportándose de forma muy similar a un ruido aleatorio limitado en banda [15].

Esta similitud es la que justifica el enfoque empleado en este TFG para generar señales simuladas que aproximen el comportamiento espectral observado en las señales reales: en lugar de construir una señal OFDM completa, con todas sus subportadoras moduladas individualmente, se genera directamente una señal aleatoria limitada a la misma banda de frecuencias, reproduciendo el aspecto espectral que resultaría de combinar muchas subportadoras, sin necesidad de modelar cada una de ellas por separado. Esta aproximación resulta adecuada para los fines de este trabajo, centrados en comparar la ocupación espectral (posición, anchura y forma general de la banda) entre señales reales y simuladas, y no en reproducir con fidelidad la estructura interna de una OFDM real.

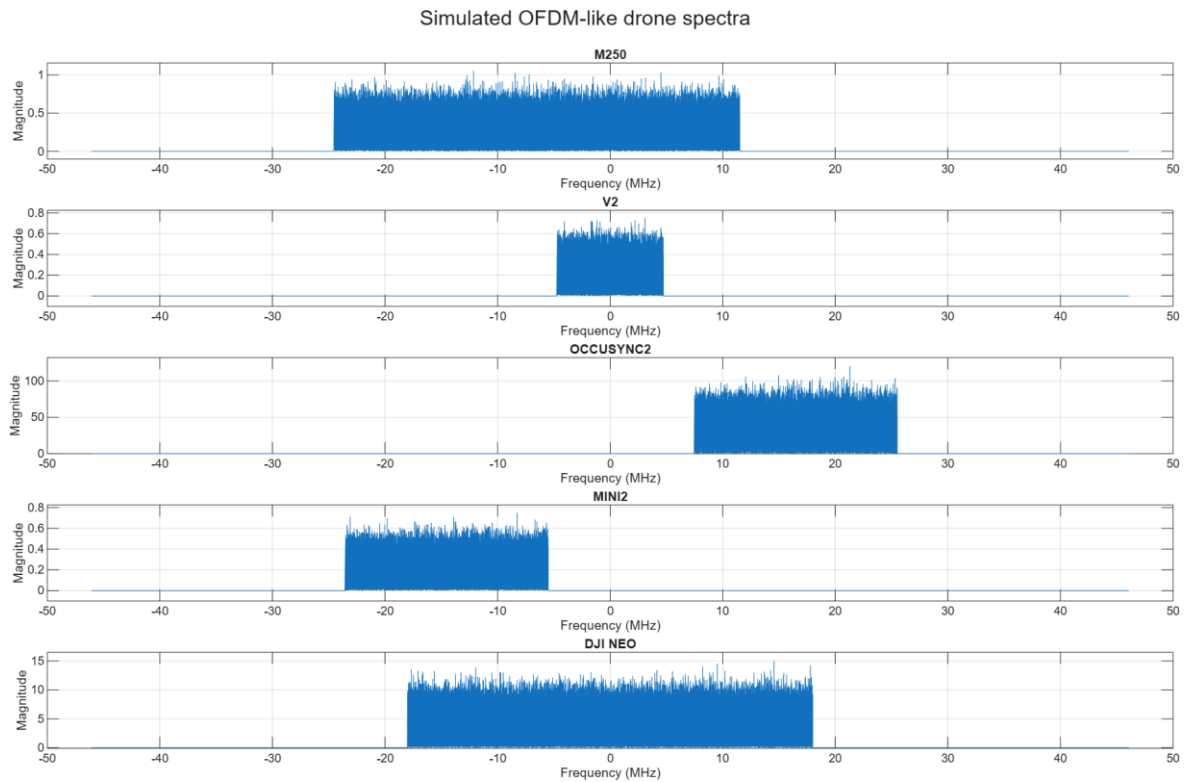


*Figura 11: Espectros de magnitud de las señales OFDM-like simuladas para cada una de las cinco señales analizadas.*

La Figura 11 muestra los espectros de magnitud resultantes para las cinco señales simuladas. En todos los casos, la energía se concentra de forma prácticamente uniforme dentro del intervalo de banda asignado a cada señal (los mismos límites  $f_l$  y  $f_h$  de la Tabla 3), sin componentes apreciables fuera de dicho intervalo, lo que confirma que el procedimiento de generación descrito reproduce correctamente la banda objetivo en cada caso. El nivel de magnitud alcanzado dentro de cada banda es, asimismo, coherente con el valor de amplitud característico de la Tabla 3, una vez aplicado el factor de escalado de 1,5 comentado anteriormente.

A diferencia de las señales reales analizadas en el apartado 5.2.2, en las que se observaban picos dominantes, tonos puntuales o componentes secundarias dentro de la banda, las señales simuladas presentan una distribución de energía considerablemente más homogénea dentro de su banda ocupada, sin estructura interna apreciable. Esta diferencia es consistente con el procedimiento de generación empleado, que asigna ruido aleatorio de forma uniforme a toda la banda sin reproducir las características específicas de cada señal real, y se retomará con más detalle en el apartado siguiente, dedicado a la comparación entre señales reales y simuladas.

## **5.4 COMPARACIÓN ENTRE SEÑALES REALES Y SIMULADAS**

Con el fin de evaluar visualmente en qué medida las señales simuladas reproducen el comportamiento espectral de las señales reales, se representan conjuntamente, para cada una de las cinco señales, el espectro de la muestra real seleccionada y el de la señal simulada correspondiente. En cada caso, ambos espectros se representan con la misma escala de magnitud (fijada a partir del valor máximo entre ambos), de forma que la comparación de niveles de amplitud entre la señal real y la simulada sea visualmente directa y no esté distorsionada por escalas independientes.

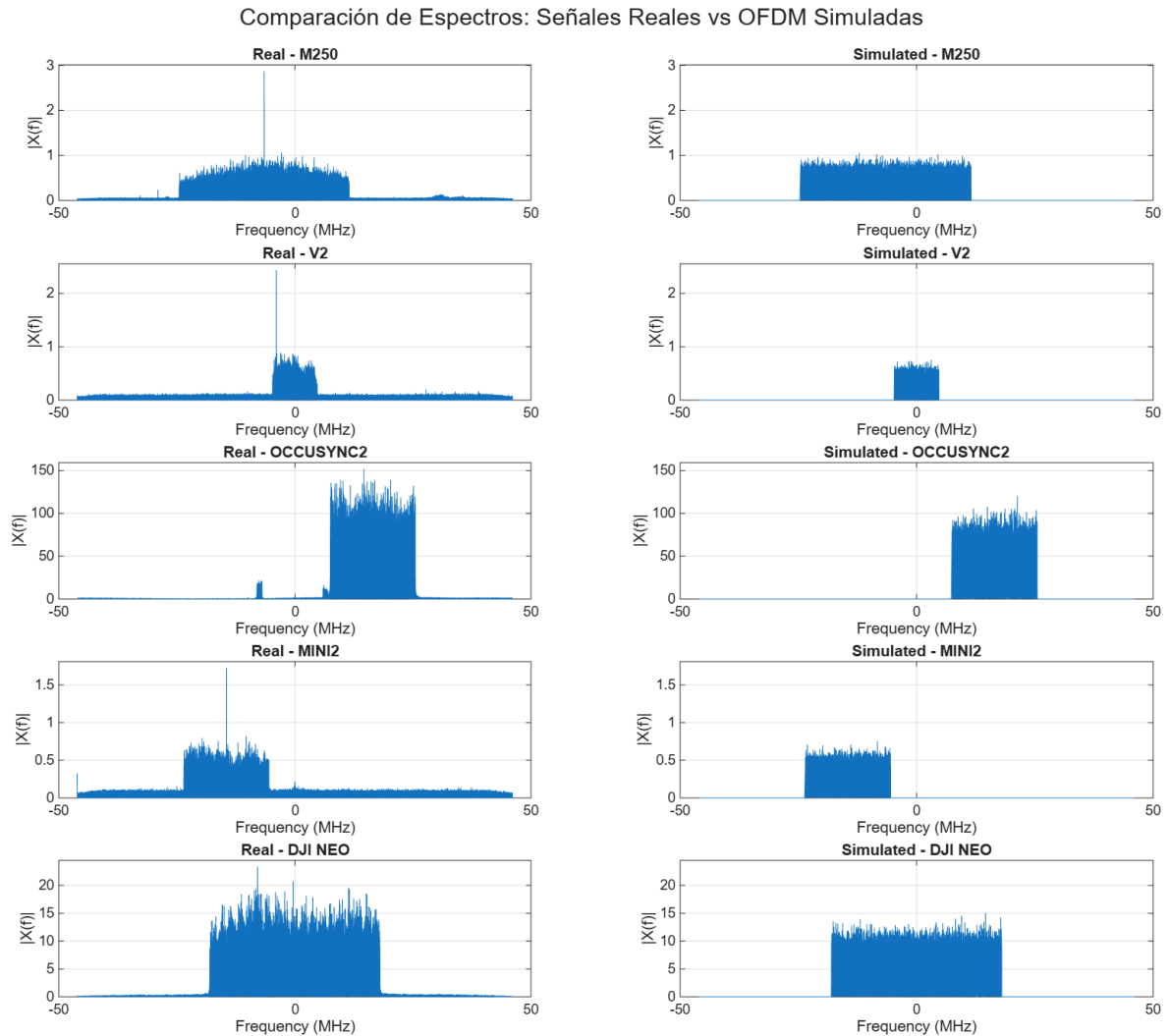


Figura 12: Comparación entre los espectros de las señales reales y las señales OFDM-like simuladas.

De la comparación mostrada en la Figura 12 se extraen dos observaciones principales. En primer lugar, las señales simuladas reproducen adecuadamente la ocupación espectral de las señales reales: tanto la posición como la anchura de la banda coinciden en los cinco casos, y el nivel de magnitud alcanzado por la señal simulada resulta del mismo orden de magnitud que el de la señal real correspondiente, lo que confirma la validez de los parámetros de banda estimados en la Tabla 3 y del procedimiento de generación descrito en el apartado 5.3.

En segundo lugar, se observa una diferencia sistemática en la estructura interna del espectro dentro de la banda ocupada. Las señales reales presentan, en varios casos, picos estrechos y dominantes claramente por encima del nivel general de la banda (especialmente visibles en M250, V2 y Mini 2), así como componentes secundarias fuera de la banda principal (como la observada en OcuSync 2). Las señales simuladas, en cambio, muestran una distribución de energía considerablemente más uniforme dentro de su banda, sin picos destacados ni componentes adicionales. Esta diferencia es coherente con el procedimiento de generación empleado, que asigna ruido aleatorio de forma homogénea a toda la banda sin modelar características específicas de cada señal real, como posibles tonos de referencia, subportadoras piloto o patrones de modulación particulares. En consecuencia, las señales simuladas resultan adecuadas para reproducir la ocupación espectral global de cada señal, pero no para reproducir su forma espectral detallada.

## ***5.5 ACOTACIÓN ESPECTRAL Y DE AMPLITUDES DE LAS SEÑALES REALES***

Las señales reales capturadas incluyen, además de la emisión propia de cada dron, componentes adicionales atribuibles al entorno de captura: interferencias de otros equipos, picos puntuales aislados, y en algunos casos bandas secundarias no presentes de forma consistente entre las tres muestras de la misma señal, como se describió en el apartado 5.2.2. Dado que el objetivo de este TFG es controlar de forma explícita los efectos de transmisión que se desean estudiar (ruido, atenuación, interferencia) en etapas posteriores del proyecto, resulta conveniente disponer también de una representación de cada señal real que aproxime, en la medida de lo posible, la emisión intrínseca del dron, sin los efectos incidentales introducidos por el entorno de captura concreto en el que se obtuvo cada muestra.

En una primera versión, la acotación se aplicó únicamente sobre el módulo del espectro,  $|X[k]|$ , obtenido mediante la función abs. El procedimiento constaba de dos pasos: en primer lugar, se eliminaban (poniendo a cero) los componentes situados fuera de la banda característica de cada señal,  $[f_l, f_h]$ , obtenida en la Tabla 3; en segundo lugar, se acotaba la magnitud máxima dentro de la banda mediante un límite fijado de forma individual para cada captura, con el fin de atenuar picos puntuales anómalos como el tono dominante observado en M250, V2 o Mini 2.

Esta primera versión presentaba una limitación de fondo: al descartar la fase de la señal mediante abs, el resultado solo podía emplearse como una representación visual del espectro, útil para el análisis, pero no permitía reconstruir una versión filtrada de la señal en el dominio temporal. La fase no es un dato accesorio, sino la que determina cómo se combinan en el tiempo las distintas componentes de frecuencia para dar lugar a la forma de onda original; aplicar la transformada inversa directamente sobre el módulo acotado no habría reconstruido la señal real filtrada, sino una señal completamente distinta, sin relación con la forma de onda original del dron. Por este motivo, en esta primera versión la señal en el dominio temporal se mantenía sin modificar, y la introducción controlada de ruido en etapas posteriores habría recaído, en la práctica, sobre la captura original sin filtrar.

Para resolver esta limitación, el procedimiento se reformuló trabajando sobre el espectro complejo,  $X[k] = \text{FFT}\{x[n]\}$ , conservando la fase en todo momento. El recorte en frecuencia se aplica directamente sobre el valor complejo de cada bin fuera de banda (poniéndolo a cero, lo que sigue siendo una operación válida sobre un número complejo), y el recorte de amplitud se realiza limitando únicamente el módulo de los bins que superan el límite establecido, manteniendo su fase original mediante  $X[k] = A_{max} \cdot e^{j\angle X[k]}$ . A partir de este espectro complejo corregido se obtiene, mediante la transformada inversa de Fourier, una señal en el dominio temporal genuinamente filtrada en banda y en amplitud.

Esta versión final es la que tiene sentido emplear en el resto del proyecto, ya que, a diferencia de la primera implementación, sí proporciona una señal temporal que conserva la estructura real de la emisión del dron, con las componentes ajenas a la banda eliminadas y los picos anómalos atenuados, lista para emplearse como referencia limpia sobre la que añadir, en etapas posteriores, los efectos de transmisión que se deseen estudiar de forma controlada.

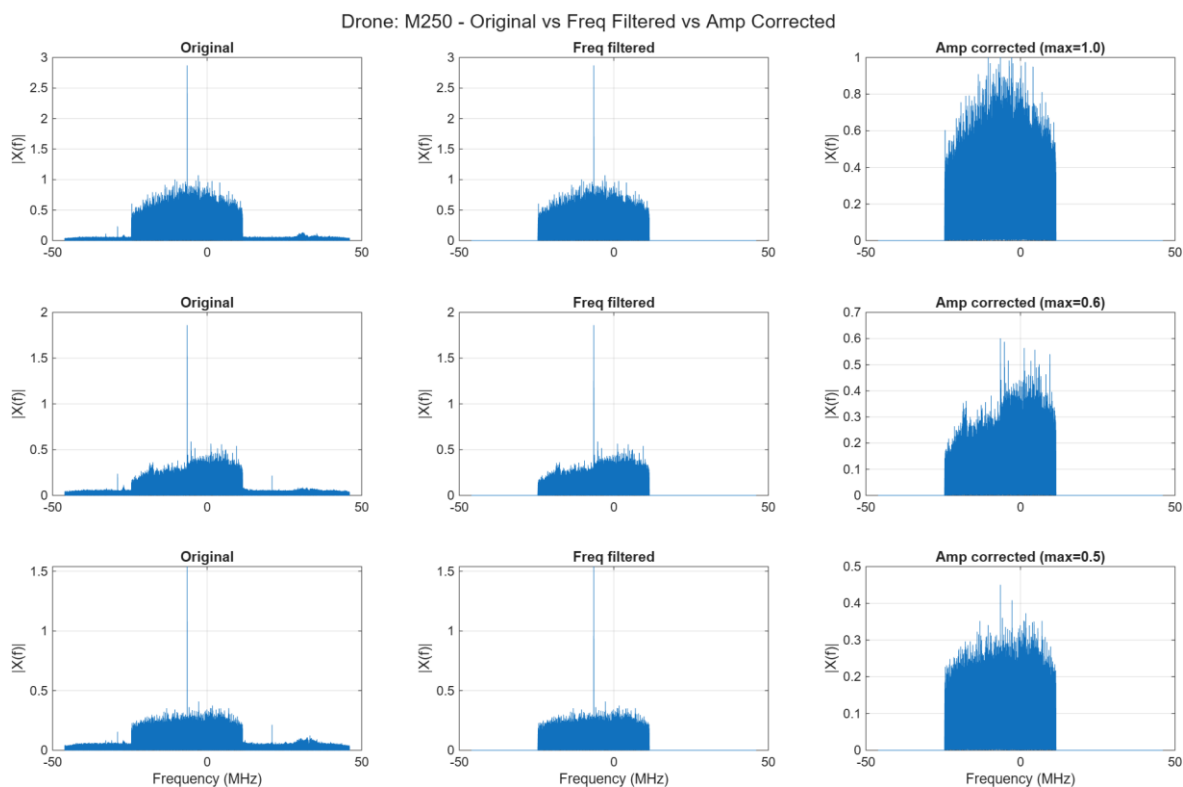


*Figura 13: Espectro original, acotado en frecuencia y acotado en frecuencia y amplitud, para las tres capturas de la señal M250.*

## **5.6 UNIFICACIÓN DE LAS SEÑALES EN UNA ESTRUCTURA COMÚN**

Las distintas etapas del desarrollo (caracterización, generación de señales simuladas, acotación de señales reales) han producido tres tipos de señal con orígenes y formatos propios. Para que las fases posteriores del proyecto, en particular el remuestreo y la transmisión mediante el ADALM-PLUTO, puedan tratar cualquier señal de forma genérica, sin necesidad de código específico para cada origen, se unifican las tres categorías de señal (simulada, real y real acotada) en una estructura de datos común, en la que todas comparten los mismos campos: la señal en el dominio temporal (signal), su frecuencia de muestreo (fs), su espectro de magnitud (spectrum), el eje de frecuencias correspondiente (freq) y el número de muestras (N).

Cada campo se obtiene de una fuente distinta según la categoría. Para las señales simuladas, los campos provienen directamente de la generación descrita en el apartado 5.3, con una única señal por dron. Para las señales reales, los campos provienen de la caracterización del apartado 5.1, manteniéndose las tres capturas disponibles por dron. Para las señales reales acotadas, el campo spectrum toma el espectro acotado en frecuencia y amplitud descrito en el apartado 5.5, mientras que el campo signal toma la señal temporal obtenida tras aplicar esa misma acotación sobre el espectro complejo, es decir, la versión genuinamente filtrada en el dominio temporal, y no la captura original sin modificar; esta distinción es relevante porque garantiza que, en las fases posteriores donde se introduzcan efectos de transmisión de forma controlada, dichos efectos se añadan sobre la versión limpia de la señal y no sobre la captura original con sus interferencias propias.

A modo de ejemplo, la Figura 14 muestra esta comparación para una de las cinco señales (M250), empleando en los tres casos la primera muestra disponible (data1); el mismo procedimiento se aplicó de forma análoga al resto de señales para verificar la coherencia de la unificación en todos los casos.

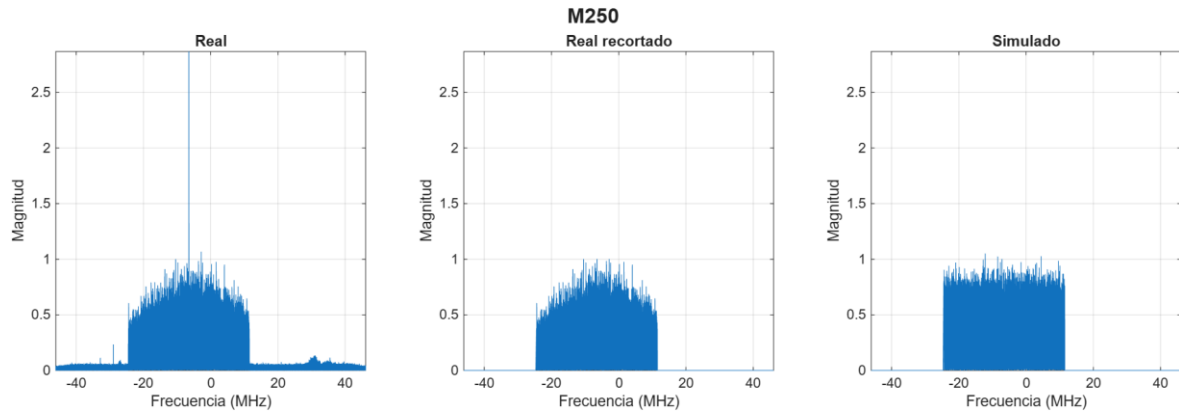


*Figura 14: Comparación de los espectros de magnitud de la señal real, la señal real acotada y la señal simulada, para la señal M250.*

## 5.7 REMUESTREO DE LAS SEÑALES PARA SU TRANSMISIÓN CON EL ADALM-PLUTO

### 5.7.1 FUNDAMENTO TEÓRICO

La conversión de la frecuencia de muestreo de una señal discreta, de  $f_{s,1}$  a una nueva frecuencia  $f_{s,2}$  puede expresarse como una relación racional

$$f_{s,2} = f_{s,1} \cdot \frac{P}{Q}$$

, donde P y Q son enteros. Esta conversión se realiza en dos etapas: una interpolación por factor P (inserción de P – 1 ceros entre muestras, seguida de un filtro paso bajo que elimina las réplicas espectrales introducidas) y una decimación por factor Q (filtrado paso bajo para evitar solapamiento espectral, seguido de la retención de una de cada Q muestras). La función *resample* de MATLAB implementa ambas etapas de forma conjunta mediante un filtro polifásico, diseñado automáticamente con una frecuencia de corte ajustada a la nueva frecuencia de Nyquist [17].

### 5.7.2 PROCEDIMIENTO

Las señales del proyecto, muestreadas originalmente a  $f_{s,1} = 92,16 \text{ MHz}$ , se convierten a una nueva frecuencia mediante  $P = 2$ ,  $Q = 3$ , lo que da lugar a:

$$f_{s,2} = 92,16 \text{ MHz} \times \frac{2}{3} = 61,44 \text{ MHz}$$

Esta conversión es exacta y no introduce error de aproximación racional, ya que  $92,16/61,44 = 3/2$  de forma exacta.

### 5.7.3 JUSTIFICACIÓN DEL VALOR DE DESTINO

El valor de 61,44 MHz no es arbitrario: corresponde a la frecuencia de muestreo máxima soportada por el transceptor AD9361/AD9363 sobre el que se construye el ADALM-PLUTO, según la documentación oficial de Analog Devices[16]. El remuestreo realizado adapta, por tanto, las señales del proyecto al límite operativo real del dispositivo con el que se realizarán las transmisiones en el capítulo 6 (Capítulo 6. ).

### 5.7.4 VALIDEZ DEL MUESTREO

Para que esta reducción de la frecuencia de muestreo no degrade la información de interés, es necesario que el contenido espectral de cada señal quede dentro de la nueva frecuencia de Nyquist,  $f_{s,2}/2 = 30,72 \text{ MHz}$ . Revisando los límites de banda obtenidos en la Tabla 3, las cinco señales ocupan como máximo el intervalo  $[-24,5, 25,5] \text{ MHz}$ , ampliamente dentro del nuevo límite de Nyquist, por lo que el remuestreo no introduce solapamiento espectral ni pérdida de las componentes de interés en ninguno de los cinco casos.

El procedimiento se aplica de forma idéntica a las tres categorías de señal de la estructura unificada (simulada, real y real acotada), recalculando en cada caso el espectro de magnitud y el eje de frecuencias a la nueva frecuencia de muestreo.

## **Capítulo 6. DISEÑO E IMPLEMENTACIÓN DE LA PLATAFORMA SDR**

### **6.1 CONFIGURACIÓN FÍSICA DE LA PLATAFORMA SDR**

La configuración física de la plataforma SDR empleada en este TFG se definió de forma progresiva, mediante un proceso de prueba y error que permitió descartar configuraciones inviables antes de fijar la configuración final empleada en el resto del proyecto.

Las primeras pruebas se realizaron transmitiendo una señal de tono puro, por su sencillez de interpretación tanto en el dominio temporal como en el frecuencial, lo que permitió validar el funcionamiento básico del ADALM-PLUTO y familiarizarse con sus parámetros de configuración antes de introducir señales más complejas.

A continuación, ya con la señal de prueba OFDM-like presentada en el capítulo 5 (Capítulo 5. ), se realizó una prueba de bucle (loopback): conectando directamente el puerto de transmisión (Tx) y el puerto de recepción (Rx) de un único ADALM-PLUTO mediante un cable SMA. Esta es la configuración recomendada habitualmente como primer paso al trabajar con esta plataforma, antes de introducir un canal de radiofrecuencia real, ya que permite verificar el correcto funcionamiento del dispositivo y del código de control sin las incertidumbres añadidas por la propagación por el aire [5].

Con el objetivo de poder realizar pruebas con una separación física mayor entre las antenas de transmisión y recepción, se intentó a continuación emplear dos unidades independientes de ADALM-PLUTO, una configurada como transmisor y otra como receptor. Esta configuración resultó inviable: al tratarse de dos dispositivos independientes, cada uno dispone de su propio oscilador de referencia interno, y al no compartir entre ambos una referencia de reloj común, sus relojes derivan de forma independiente con el tiempo. Esto se traduce en un desplazamiento de frecuencia portadora (CFO, Carrier Frequency Offset) y en un deslizamiento progresivo entre las muestras transmitidas y las recibidas, lo que impidió obtener resultados coherentes con esta configuración.

Como alternativa para conseguir mayor separación sin recurrir a dos unidades independientes, se adquirieron cables de extensión SMA macho-hembra con el fin de alejar las antenas del propio dispositivo manteniendo un único ADALM-PLUTO. Esta solución tampoco funcionó: a pesar de que los cables tenían la combinación de género correcta (macho-hembra) y el conector correspondía nominalmente al estándar SMA, el contacto eléctrico con el puerto del ADALM-PLUTO no resultó fiable, probablemente por una tolerancia mecánica insuficiente del pin o la rosca del cable concreto adquirido, lo que se tradujo en mediciones anómalas e inconsistentes.

Tras consultar esta dificultad con el director del proyecto, se decidió simplificar la configuración, renunciando al objetivo de separar físicamente las antenas, y conectando ambas directamente sobre un único ADALM-PLUTO, configuración que se mantuvo durante el resto del desarrollo experimental.

## **6.2 VALIDACIÓN INICIAL Y FUNDAMENTOS DE OPERACIÓN DEL ADALM-PLUTO CON UNA SEÑAL DE TONO PURO**

Una vez fijada la configuración física definitiva (las dos antenas incluidas con el dispositivo, conectadas directamente sobre un único ADALM-PLUTO), se realizó una primera validación funcional transmitiendo y recibiendo una señal de tono puro. Esta señal, al concentrar toda su energía en una única frecuencia, resulta especialmente adecuada para verificar de forma sencilla e inequívoca el correcto funcionamiento del dispositivo, tanto en el dominio temporal como en el frecuencial, y para introducir los parámetros de configuración que se emplean en el resto del capítulo.

### **6.2.1 PARÁMETROS DE LA SEÑAL Y DE LA CAPTURA**

La frecuencia de muestreo se fija en  $f_s = 61,44 \text{ MHz}$ , el valor máximo soportado por el transceptor AD9361/AD9363 del ADALM-PLUTO, ya justificado en el apartado 5.7 [16]. El número de muestras por captura,  $N = 961.196$ , se hizo coincidir con la longitud de las señales reales y simuladas remuestreadas en el capítulo 5 (Capítulo 5. ), de forma que las pruebas con el Pluto trabajen desde el principio con tramas de la misma duración que las que se emplearán más adelante con las señales del proyecto; este valor debe ser, además, par, requisito impuesto por el propio dispositivo para el envío y recepción de muestras IQ.

El tono se genera como una exponencial compleja de amplitud  $A = 0,2$  y frecuencia  $f_{\text{tono}} = 100 \text{ kHz}$  respecto a la frecuencia central:

$$x[n] = A \cdot e^{j2\pi f_{\text{tono}}n/f_s}$$

Esta frecuencia de 100 kHz se eligió lo suficientemente alejada de 0 Hz para poder distinguir visualmente el tono de posibles artefactos de continua (fuga del oscilador local) en torno a la frecuencia central, sin necesidad de alejarse mucho de banda base.

## 6.2.2 CONFIGURACIÓN DEL DISPOSITIVO DESDE MATLAB

El control del ADALM-PLUTO se realiza mediante los objetos `comm.SDRTxPluto` y `comm.SDRRxPluto` del Communications Toolbox Support Package correspondiente [18]:

```
tx = comm.SDRTxPluto('CenterFrequency', fc, ...  
    'BasebandSampleRate', fs, ...  
    'Gain', txGain);  
  
rx = comm.SDRRxPluto('CenterFrequency', fc, ...  
    'BasebandSampleRate', fs, ...  
    'GainSource', 'Manual', ...  
    'Gain', rxGain, ...  
    'SamplesPerFrame', N, ...  
    'OutputDataType', 'double');
```

`CenterFrequency` fija la frecuencia central ( $f_c$ ) del mezclador, tanto en transmisión como en recepción. `BasebandSampleRate` fija la frecuencia de muestreo, que según la documentación oficial admite valores entre 65.105 Hz y 61,44 MHz [18], coherente con el límite ya empleado en el apartado 5.7. `Gain`, en el objeto transmisor, controla la ganancia de la cadena de RF de transmisión (`txGain`), con un rango admitido de -89,75 a 0 dB [18]. En el objeto receptor, `GainSource` se fija a 'Manual' para desactivar el control automático de ganancia (AGC) que el dispositivo aplicaría por defecto, y así poder fijar manualmente la ganancia de recepción (`rxGain`) mediante la propiedad `Gain`, cuyo rango admitido depende de la frecuencia central elegida [18]. Por último, `SamplesPerFrame` fija el número de muestras que se capturan en cada llamada al objeto receptor, y `OutputDataType` determina que las muestras recibidas se devuelvan en formato de coma flotante de doble precisión en lugar del formato nativo del dispositivo.

### 6.2.3 PROCEDIMIENTO

La señal se transmite de forma continua mediante la función `transmitRepeat`, de modo que el tono se repite sin interrupción mientras el receptor captura una única trama de  $N$  muestras. Tanto la señal transmitida como la recibida se representan en frecuencia mediante su espectro de magnitud, en dos versiones: una en escala lineal y otra en escala logarítmica (dB), esta última útil para apreciar con claridad componentes de baja amplitud que en la escala lineal apenas serían visibles. Ambas representaciones se limitan a un margen de  $\pm 1$  MHz alrededor de la frecuencia del tono, donde se concentra toda la información relevante de esta prueba. Como primera comprobación cuantitativa, se compara la amplitud de pico de la señal transmitida con la de la señal recibida, expresando la diferencia en dB; este valor ofrece una primera idea de cuánta señal se pierde entre la antena de transmisión y la de recepción en esta configuración.

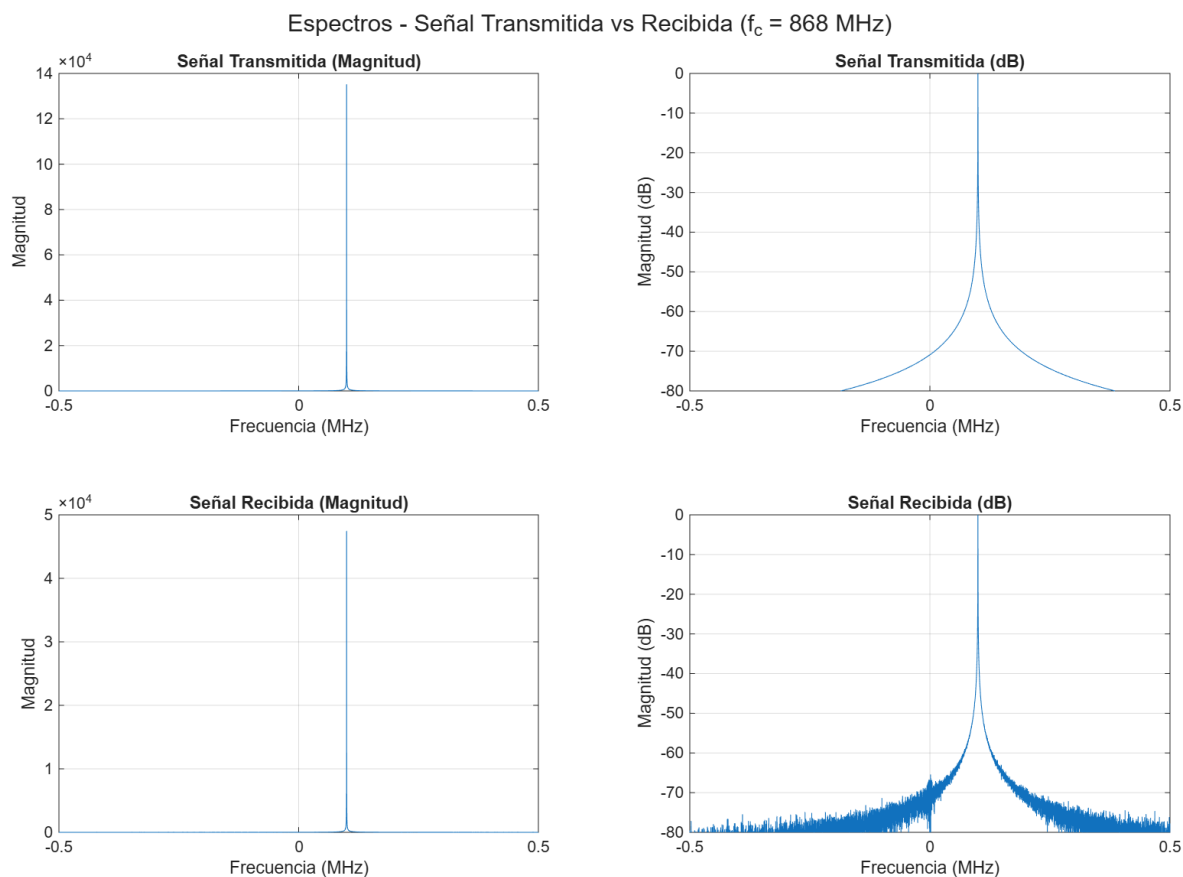


*Figura 15: Espectros de magnitud de la señal transmitida y la señal recibida, en escala lineal y logarítmica, para la prueba con tono puro a  $f_c = 868 \text{ MHz}$  ( $f_{\text{tono}} = 100 \text{ kHz}$ ,  $\text{`txGain`} = -20 \text{ dB}$ ,  $\text{`rxGain`} = 15 \text{ dB}$ ).*

La Figura 15 compara los espectros de la señal transmitida y la señal recibida, en magnitud lineal y en escala logarítmica (dB), para la prueba con tono puro a  $f_c = 868 \text{ MHz}$ .

En la **señal transmitida** se observa un único pico claramente dominante en torno a  $100 \text{ kHz}$ , como corresponde a un tono puro. En la representación en dB se aprecia una caída suave y simétrica alrededor del pico, visible entre aproximadamente  $-60$  y  $-80 \text{ dB}$ . Este efecto no proviene de la señal en sí, sino del propio cálculo de la FFT sobre un número finito de muestras: trabajar con un tramo finito de la señal equivale a multiplicarla por una ventana rectangular, lo que en el dominio de la frecuencia genera la denominada fuga espectral (spectral leakage), cuya forma sigue un perfil de tipo sinc [14]. Se trata, por tanto, de un artefacto esperado del procesamiento digital.

En la **señal recibida**, el tono aparece exactamente en la misma frecuencia que en transmisión, sin desplazamiento apreciable, lo cual confirma el correcto funcionamiento del dispositivo y es coherente con que transmisor y receptor compartan el mismo oscilador de referencia al tratarse de un único ADALM-PLUTO. En la representación en dB de la señal recibida aparece, además, un nivel de ruido de fondo visible en torno a -75 a -80 dB, resultado de la suma del ruido térmico y del ruido cuantificación del conversor analógico-digital del dispositivo.

La representación en dB de la señal recibida puede resultar llamativa a primera vista, ya que el espectro parece considerablemente más "sucio" que el de la señal transmitida, con un nivel de ruido de fondo visible y fluctuante en torno a -75 a -80 dB. Sin embargo, esta percepción es en gran medida un efecto de la propia escala logarítmica, y no refleja necesariamente una degradación severa de la señal. Para entenderlo, basta recordar la definición de la escala en dB:

$$X_{dB} = 20 \log_{10} \left( \frac{|X|}{|X|_{max}} \right)$$

Una componente situada a -75 dB del pico principal representa, en términos de amplitud, una fracción de  $10^{-75/20} \approx 0,006$  respecto al pico, es decir, menos del 1% de su amplitud. En la escala lineal, esta diferencia entre el pico y el nivel de ruido de fondo es tan grande que el ruido resulta prácticamente invisible (se confunde con el cero), mientras que en la escala logarítmica esa misma diferencia ocupa tres cuartas partes del rango vertical de la gráfica, lo que hace que el ruido parezca mucho más prominente de lo que realmente es en términos de energía.

Este efecto de amplificación visual de las componentes de bajo nivel es precisamente la razón por la que la representación en dB resulta tan útil en análisis de señales RF: permite detectar señales espurias, armónicos o niveles de ruido que en una escala lineal pasarían completamente desapercibidos. Pero también exige interpretar las gráficas con cuidado, entendiendo que una representación "visualmente sucia" en dB puede corresponder a una señal perfectamente limpia en términos prácticos, cuyo ruido de fondo está decenas de decibelios por debajo de la señal de interés.

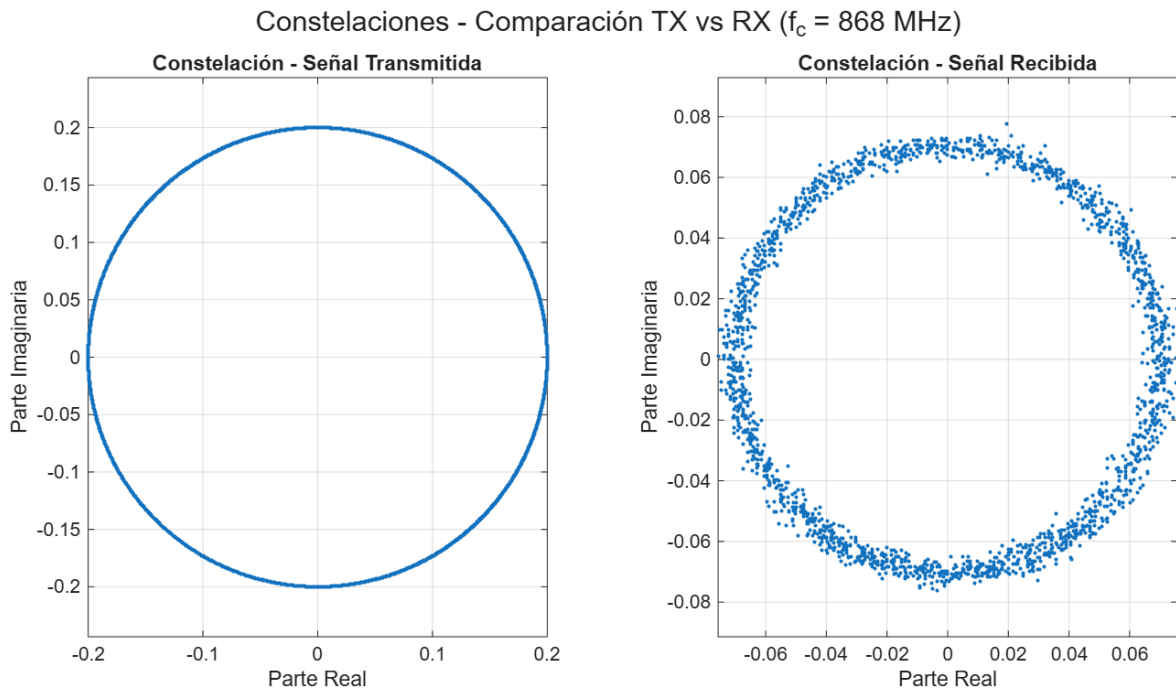


*Figura 16: Diagrama de constelación de la señal transmitida y la señal recibida para la prueba con tono puro a  $f_c=868$ .*

El diagrama de constelación (Figura 16) permite observar la señal en el plano IQ (parte real frente a parte imaginaria). La señal transmitida describe una circunferencia perfecta de radio  $A = 0,2$ , comportamiento esperado de una exponencial compleja de amplitud y frecuencia constantes: a medida que avanza en el tiempo, su fase rota de forma uniforme y su módulo permanece constante, generando el círculo característico. En la señal recibida, la circunferencia se mantiene reconocible pero con dos diferencias notables: el radio es considerablemente menor (en torno a 0,07), reflejo de las pérdidas de propagación entre las dos antenas en esta configuración; y los puntos no forman una línea continua y perfecta sino un anillo con dispersión, cuyo grosor refleja la presencia del ruido añadido por la cadena de recepción descrito anteriormente.

#### **6.2.4 EFECTO DE LA FRECUENCIA PORTADORA: COMPARACIÓN ENTRE 868 MHz Y 2,4 GHz**

Con el fin de evaluar el comportamiento del dispositivo a la frecuencia de operación real de las señales de dron objeto de este TFG (banda ISM de 2,4 GHz), se repitió la misma prueba modificando únicamente la frecuencia portadora, de 868 MHz a 2.400 MHz, manteniendo el resto de parámetros idénticos.

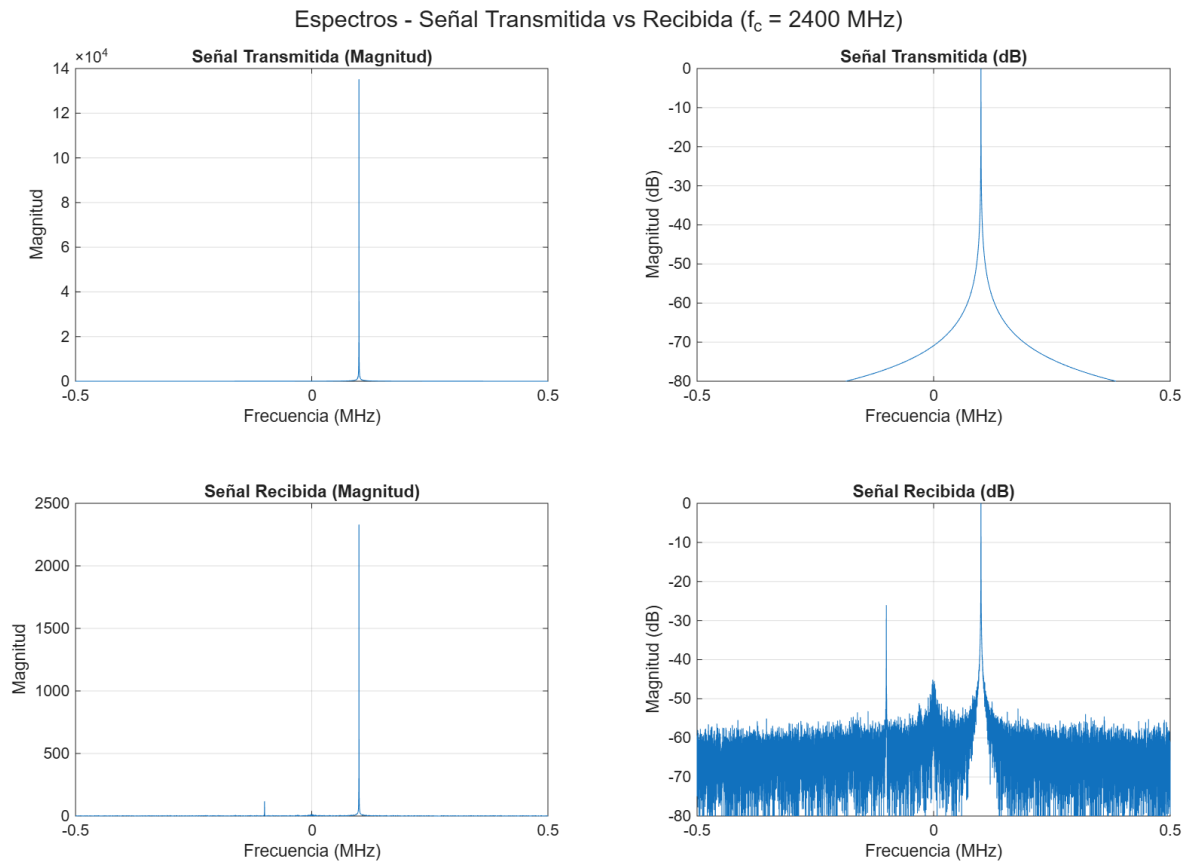


*Figura 17: Espectros de magnitud de la señal transmitida y la señal recibida, en escala lineal y logarítmica, para la prueba con tono puro a  $f_c = 2.400$  MHz.*

La comparación con los resultados obtenidos a 868 MHz revela diferencias significativas. En primer lugar, la amplitud de pico de la señal recibida es notablemente inferior: mientras que a 868 MHz el pico recibido alcanzaba un valor del orden de  $5 \times 10^4$ , a 2.400 MHz apenas supera 2.300, una diferencia de más de un orden de magnitud. Esto es consistente con un aumento de las pérdidas de propagación con la frecuencia, que siguen la ley de Friis: para una misma distancia y ganancia de antena, las pérdidas en espacio libre crecen con el cuadrado de la frecuencia, lo que implica que a 2,4 GHz las pérdidas son aproximadamente  $(2400/868)^2 \approx 7,6$  veces mayores que a 868 MHz, es decir, unos 8,8 dB adicionales [19].

En segundo lugar, la representación en dB de la señal recibida a 2,4 GHz muestra un nivel de ruido de fondo considerablemente más elevado (en torno a -55 a -60 dB frente a los -75 a -80 dB de la prueba a 868 MHz), así como la presencia de componentes adicionales cerca de la frecuencia central (en torno a 0 Hz) que no aparecían a 868 MHz. Estas componentes son atribuibles a una mayor fuga del oscilador local a frecuencias más altas, un fenómeno conocido (LO leakage) que afecta especialmente a los transceptores de conversión directa como el AD9361 del ADALM-PLUTO [1].

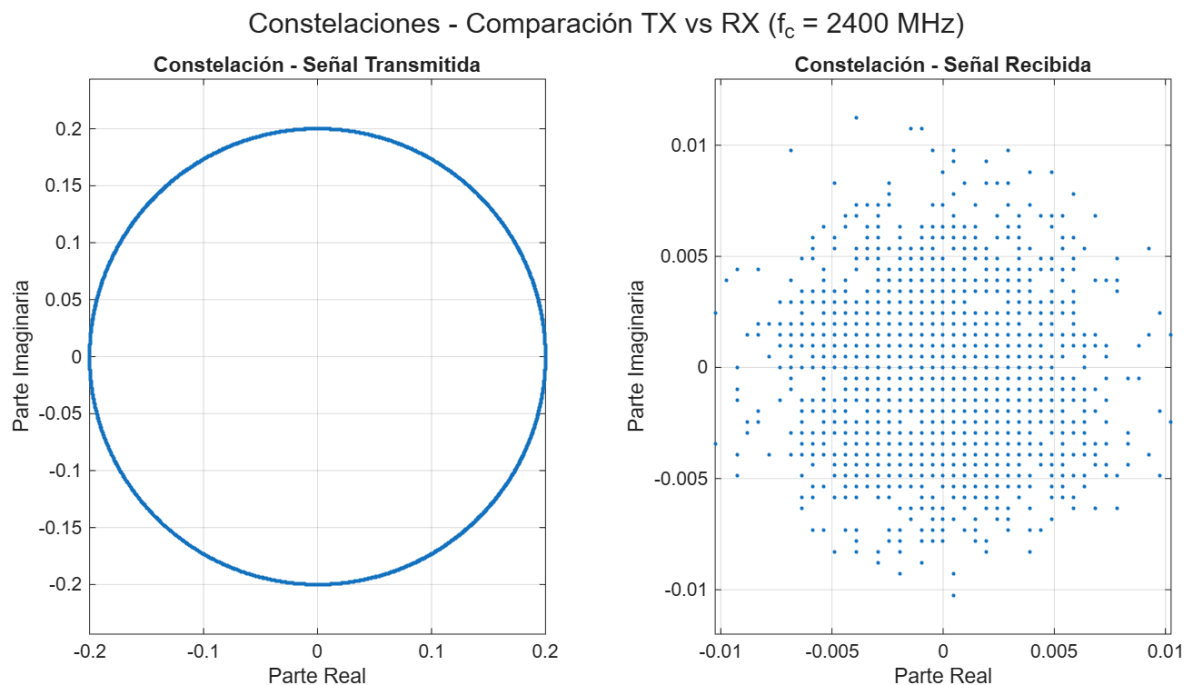


*Figura 18: Diagrama de constelación de la señal transmitida y la señal recibida para la prueba con tono puro a  $f_c = 2.400$  MHz.*

El diagrama de constelación a 2,4 GHz confirma esta degradación de forma aún más visual. La señal transmitida sigue describiendo la circunferencia perfecta esperada de una exponencial compleja. Sin embargo, la señal recibida ya no muestra un anillo reconocible, sino una nube de puntos dispersos sin estructura circular apreciable. Esto indica que la relación señal-ruido en recepción es insuficiente para recuperar el tono con un margen de fase y amplitud razonable con estos parámetros de ganancia: la energía del tono recibido es tan pequeña frente al ruido de fondo que la forma circular queda completamente enmascarada.

Esta comparación entre 868 MHz y 2,4 GHz justifica la elección de 868 MHz como frecuencia de trabajo para las pruebas iniciales de validación del dispositivo: al operar en una frecuencia más baja, las pérdidas de propagación son menores y la señal recibida presenta una relación señal-ruido significativamente más favorable, lo que facilita la verificación del correcto funcionamiento del sistema antes de abordar condiciones de prueba más exigentes.

Existe además un factor que agrava especialmente la degradación observada a 2,4 GHz más allá de las pérdidas de propagación: esta frecuencia coincide con la banda ISM utilizada por numerosos dispositivos de uso cotidiano, entre los que destaca el estándar Wi-Fi (IEEE 802.11 b/g/n), así como Bluetooth y otros sistemas de comunicación inalámbrica de corto alcance [20], [21]. En un entorno de laboratorio o doméstico, esta banda se encuentra habitualmente ocupada por múltiples transmisores activos de forma simultánea, lo que introduce interferencias externas que elevan el nivel de ruido aparente en recepción y contribuyen a la degradación espectral observada. La banda de 868 MHz, en cambio, corresponde a un uso mucho más restringido, con una ocupación espectral considerablemente menor en entornos interiores, lo que explica en parte por qué las pruebas realizadas a esa frecuencia mostraron un espectro recibido notablemente más limpio.

Para las pruebas de transmisión y recepción con señal OFDM descritas en los apartados siguientes, se empleó una señal de prueba generada artificialmente siguiendo el procedimiento de síntesis descrito en el apartado 5.3. Concretamente, la señal utilizada reproduce los parámetros espectrales de la señal OcuSync 2, con una banda ocupada entre 7,5 y 25,5 MHz, una frecuencia de muestreo original de 92,16 MHz y una amplitud característica de 120 (equivalente al valor de la Tabla 3 multiplicado por el factor 1,5). Antes de su transmisión, la señal se remuestrea de 92,16 MHz a 61,44 MHz mediante el mismo procedimiento de interpolación y diezmado descrito en el apartado 5.7 ( $P = 2$ ,  $Q = 3$ ), obteniendo un vector de  $N_{res}$  muestras listo para su uso con el ADALM-PLUTO.

La elección de una señal sintética para estas pruebas, en lugar de una de las capturas reales, responde a un criterio de control experimental: al tratarse de una señal generada con parámetros exactamente conocidos, cualquier diferencia entre la señal transmitida y la recibida puede atribuirse inequívocamente a los efectos de la cadena de transmisión y recepción, y no a variaciones propias de la señal real. El objetivo de estas pruebas no era reproducir fielmente las condiciones de operación de un dron real, sino evaluar el comportamiento de cada configuración física de la plataforma SDR de forma controlada y reproducible.

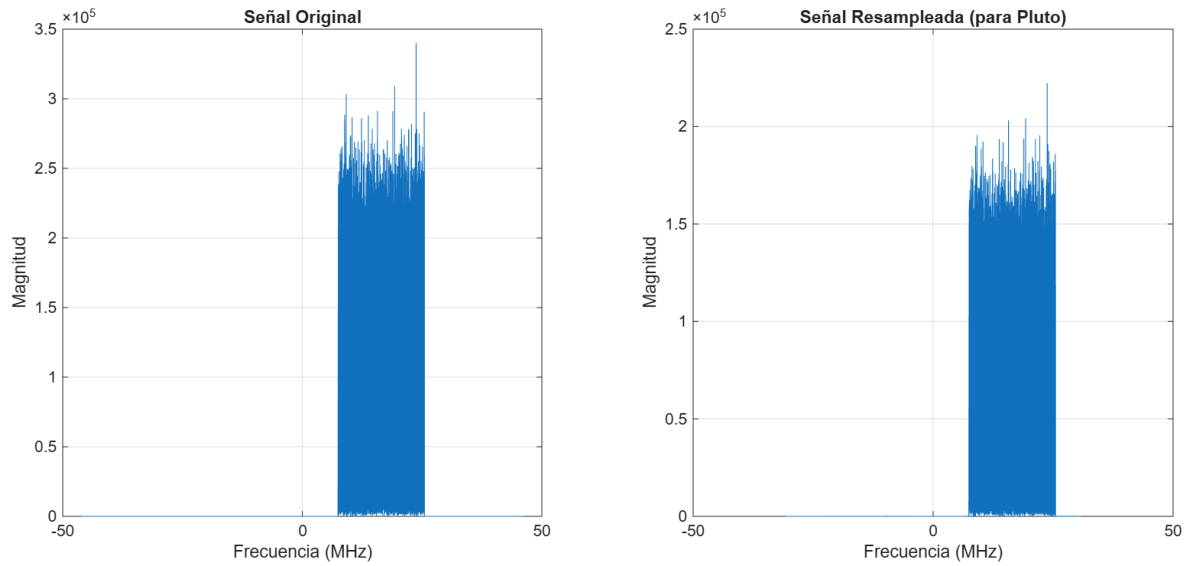


*Figura 19: Espectro de magnitud de la señal de prueba OFDM-like antes y después del remuestreo de 92,16 MHz a 61,44 MHz.*

La Figura 19 muestra el efecto del remuestreo sobre el espectro de la señal de prueba. La banda ocupada, comprendida entre 7,5 y 25,5 MHz, se conserva íntegramente en ambas representaciones, confirmando que la reducción de la frecuencia de muestreo no introduce pérdida de información espectral, en coherencia con lo argumentado en el apartado 5.7: el contenido de la señal queda bien por debajo de la nueva frecuencia de Nyquist de 30,72 MHz. La diferencia más visible entre ambas gráficas es la reducción de la magnitud absoluta tras el remuestreo, que se explica por la normalización implícita del filtro polifásico aplicado durante el proceso[17]. La posición, anchura y forma general de la banda ocupada permanecen inalteradas, lo que valida la idoneidad del factor de conversión  $P/Q = 2/3$  elegido para adaptar la señal al ADALM-PLUTO.

### **6.3 SEÑAL OFDM CON CABLE DIRECTO ENTRE RX Y TX**

Una vez validado el funcionamiento básico del ADALM-PLUTO con la señal de tono puro, se repitió una comprobación equivalente empleando la señal de prueba OFDM-like descrita anteriormente. Para ello se conectaron directamente los puertos de transmisión (Tx) y recepción (Rx) del dispositivo mediante un cable SMA, configuración conocida como bucle o loopback, que elimina las incertidumbres asociadas a la propagación por el aire y permite verificar el correcto funcionamiento de la cadena de transmisión y recepción con una señal más compleja antes de introducir un canal radioeléctrico real [5].

Esta prueba confirmó que la plataforma SDR era capaz de transmitir y recibir correctamente la señal OFDM-like, reproduciéndose en recepción la banda ocupada entre 7,5 y 25,5 MHz con una forma espectral coherente con la señal transmitida. Los resultados de esta comprobación no se conservaron de forma sistemática, al tratarse de una validación preliminar cuyo único objetivo era confirmar el correcto funcionamiento del sistema antes de pasar a evaluar configuraciones físicas más complejas.

### **6.4 SEÑAL OFDM CON DOS PLUTOS INDEPENDIENTES**

Con el objetivo de ampliar la separación física entre las antenas de transmisión y recepción, se intentó emplear dos unidades independientes de ADALM-PLUTO, una configurada como transmisor y otra como receptor, identificadas desde MATLAB mediante los identificadores de dispositivo usb:0 y usb:1. Es importante señalar que esta configuración requiere disponer de dos puertos USB libres en el equipo host, así como que el sistema operativo reconozca ambos dispositivos de forma simultánea, condición que no siempre se cumple de forma inmediata y puede requerir ajustes en los drivers o en la asignación de identificadores.

La prueba no proporcionó resultados utilizables, como se aprecia claramente en la Figura 20: la señal recibida no muestra ninguna relación reconocible con la señal transmitida. En lugar de reproducir la banda ocupada entre 7,5 y 25,5 MHz, el espectro recibido presenta energía distribuida de forma caótica por toda la banda, con dos lóbulos simétricos pronunciados en los extremos y un nivel de magnitud entre 4 y 25, frente a los valores de 50 a 350 que se observan en la señal transmitida.

La causa de este resultado es el problema de sincronización de relojes entre dos dispositivos SDR independientes. Cada unidad de ADALM-PLUTO dispone de su propio oscilador de cristal interno de 40 MHz, y al no existir ningún mecanismo de sincronización entre ambos, sus frecuencias de referencia derivan de forma independiente con el tiempo. Esto se traduce en un desplazamiento de frecuencia portadora (CFO, Carrier Frequency Offset) entre transmisor y receptor que varía progresivamente, haciendo que la señal recibida no se corresponda con la transmitida. Este problema es bien conocido en el ámbito de las plataformas SDR de bajo coste: el foro de soporte oficial de Analog Devices para el ADALM-PLUTO recoge explícitamente que la sincronización de reloj entre dos unidades independientes no está soportada en este modelo, y que las desviaciones de frecuencia observadas entre dos dispositivos pueden superar los 7 kHz [23].

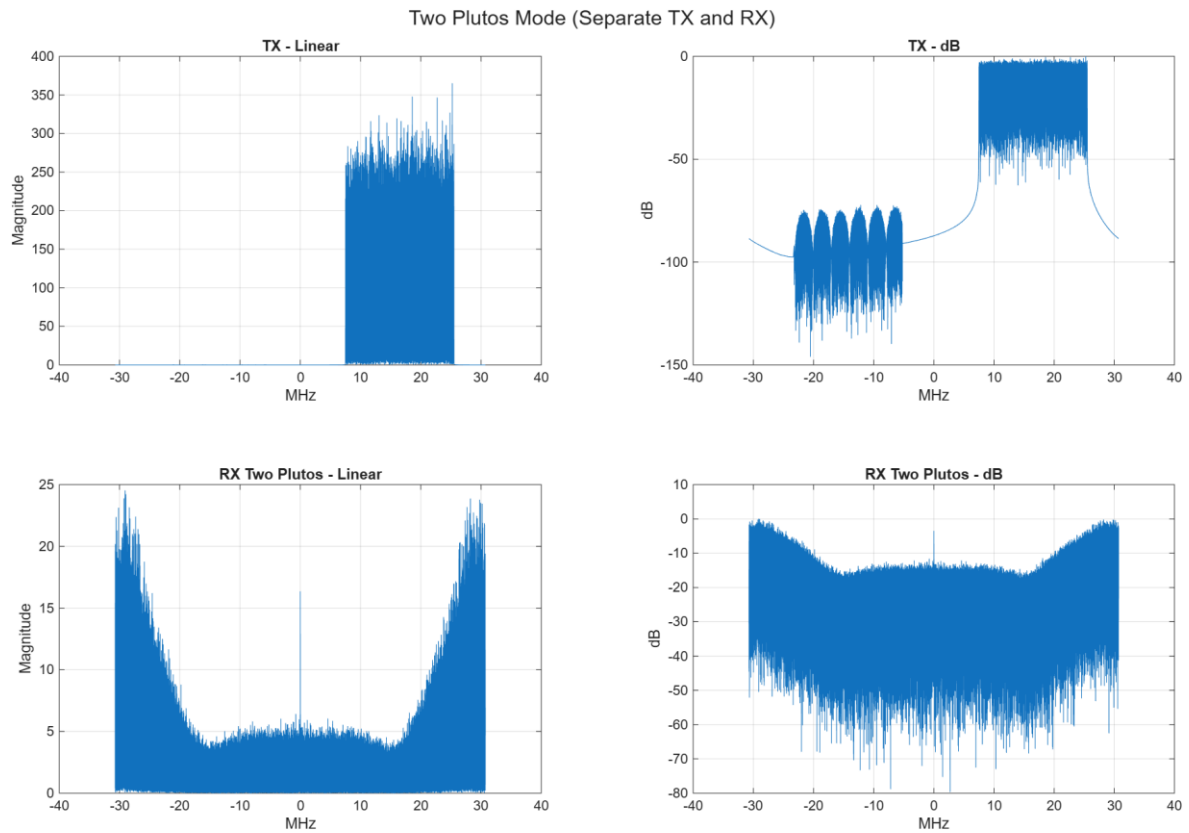


*Figura 20: Espectros de magnitud de la señal transmitida y la señal recibida con la configuración de dos Plutos independientes, en escala lineal y logarítmica.*

La resolución de este problema de sincronización (por ejemplo, mediante el uso de una referencia de reloj externa compartida entre ambos dispositivos, o mediante técnicas de estimación y corrección software del CFO) se alejaba del objetivo de este TFG. Por ello, como alternativa para intentar mantener cierta separación física entre las antenas sin recurrir a dos unidades independientes, se optó por el uso de cables de extensión SMA, descrito en el apartado siguiente.

## ***6.5 SEÑAL OFDM CON UN PLUTO Y CABLES (ANTENAS CONECTADAS MEDIANTE CABLE SMA)***

Ante la inviabilidad de la configuración con dos unidades independientes de ADALM-PLUTO, se planteó una solución alternativa para mantener cierta separación física entre las antenas de transmisión y recepción sin necesidad de un segundo dispositivo: el uso de cables de extensión SMA que permitieran alejar las antenas del propio módulo. Para ello se adquirieron dos cables de extensión SMA macho-hembra de 50 cm, modelo Boobrie RG316, de impedancia nominal 50 ohmios y conector chapado en oro, disponibles comercialmente para aplicaciones de RF y SDR.

Las pruebas se realizaron con distancias entre antenas de 10, 20, 50, 75 y 100 cm, manteniendo un único ADALM-PLUTO como dispositivo transceptor. Los resultados, sin embargo, no fueron satisfactorios en ninguna de las distancias evaluadas, como se aprecia en la Figura 21: el espectro recibido no reproduce la banda transmitida (7,5-25,5 MHz), sino que presenta un pico dominante desplazado en torno a 7 MHz, un nivel de energía elevado y desestructurado fuera de la banda de interés, y una forma espectral que no guarda relación con la señal transmitida. Además, los resultados no variaban de forma apreciable entre las distintas distancias evaluadas, lo que indica que la distancia entre antenas no era el parámetro relevante que determinaba el comportamiento del sistema, sino que el problema tenía un origen más fundamental.

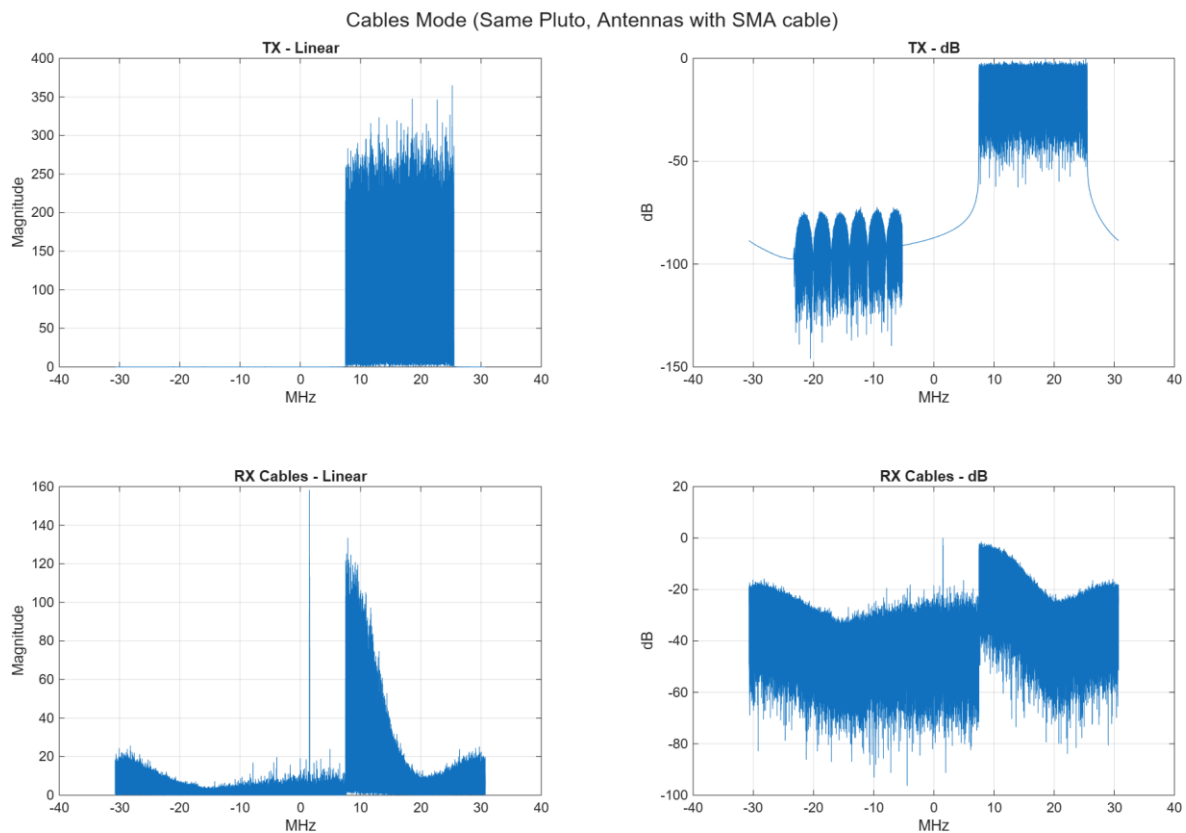


*Figura 21: Espectros de magnitud de la señal transmitida y la señal recibida con la configuración de un único Pluto y cables de extensión SMA, en escala lineal y logarítmica.*

Tras analizar la situación con el director del proyecto, se concluyó que la causa más probable era una tolerancia mecánica insuficiente del conector del cable adquirido: aunque el estándar nominal del conector era SMA macho-hembra, el pin de contacto o la rosca no encajaba con la precisión suficiente en el puerto del ADALM-PLUTO, impidiendo un contacto eléctrico fiable. Esta hipótesis es consistente con que los resultados fueran igualmente deficientes a todas las distancias, ya que un mal contacto en el punto de conexión con el dispositivo introduciría distorsiones independientemente de la longitud del cable empleado.

Dado que la resolución de este problema de compatibilidad mecánica (por ejemplo, mediante la adquisición de cables certificados específicamente para el conector del ADALM-PLUTO) se alejaba del objetivo del TFG, se decidió prescindir de los cables de extensión y adoptar la configuración más sencilla disponible, descrita en el apartado siguiente.

## **6.6 SEÑAL OFDM CON ANTENAS DIRECTAS SOBRE UN ÚNICO PLUTO (SOLUCIÓN ADOPTADA)**

Tras descartar las configuraciones anteriores, se adoptó la configuración más sencilla disponible: las dos antenas incluidas con el ADALM-PLUTO conectadas directamente a sus puertos de transmisión y recepción, sin ningún cable de extensión intermedio. Esta configuración, aunque no permite controlar la distancia entre antenas, elimina cualquier fuente de error asociada a problemas de sincronización entre dispositivos o de contacto eléctrico, y garantiza la máxima reproducibilidad de las condiciones de prueba.

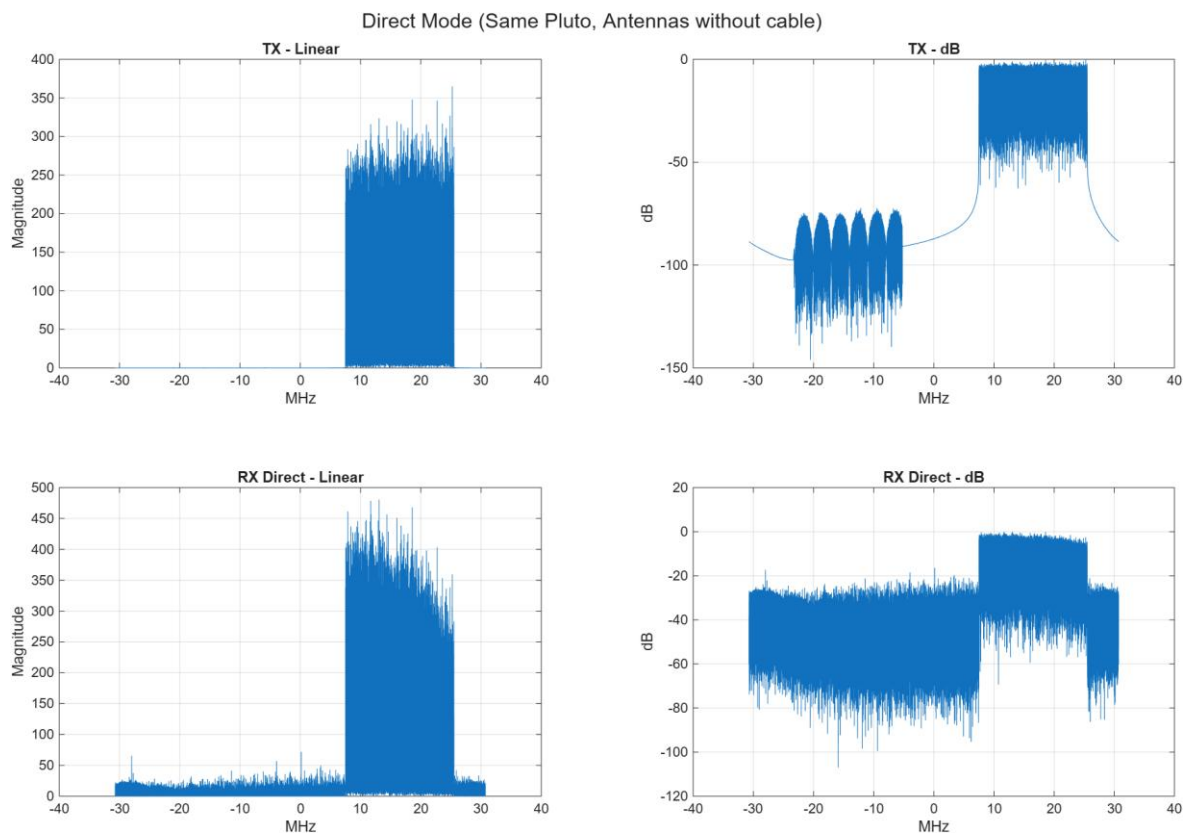


*Figura 22: Espectros de magnitud de la señal transmitida y la señal recibida con la configuración de antenas directas sobre un único Pluto, en escala lineal y logarítmica.*

Los resultados obtenidos con esta configuración son claramente satisfactorios, como se aprecia en la Figura 22. A diferencia de las configuraciones anteriores, la señal recibida reproduce con fidelidad la banda transmitida: en la representación lineal se observa un bloque de energía comprendido entre aproximadamente 7,5 y 25,5 MHz, con un nivel de magnitud que alcanza valores en torno a 400-475, claramente distinguible del nivel de ruido de fondo, que se mantiene por debajo de 50. En la representación en dB, la banda transmitida aparece en torno a 0 dB, con una caída brusca y bien definida en sus bordes, mientras que el nivel de ruido fuera de la banda se sitúa entre -20 y -30 dB, lo que supone un margen de al menos 20 dB entre la señal de interés y el ruido de fondo.

Esta relación señal-ruido, junto con la correcta reproducción de la posición y anchura de la banda transmitida, confirma que la configuración de antenas directas sobre un único ADALM-PLUTO proporciona resultados coherentes y con suficiente margen dinámico para las etapas siguientes del proyecto: la introducción controlada de ruido sobre las señales recibidas y la aplicación de los algoritmos de detección desarrollados en el capítulo 7 (Capítulo 7. ).

## ***6.7 BARRIDO DE PARÁMETROS Y TRANSMISIÓN DE LAS SEÑALES DEL PROYECTO***

Una vez fijada la configuración física definitiva, el siguiente paso fue determinar qué combinación de parámetros de operación del ADALM-PLUTO proporcionaba los mejores resultados en recepción. En lugar de ajustar los parámetros a mano por prueba y error, se optó por un barrido sistemático: se probaron todas las combinaciones posibles de los cuatro parámetros principales y se evaluó cada una de forma objetiva mediante tres métricas calculadas automáticamente.

Los cuatro parámetros barridos y los valores evaluados para cada uno fueron:

- Frecuencia portadora ( $f_c$ ): 433 MHz, 868 MHz y 2.450 MHz. Este parámetro determina en qué punto del espectro de RF se transmite la señal. Se probaron tres frecuencias distintas para comprobar en cuál de ellas el dispositivo ofrecía mejores resultados en este entorno concreto.
- Ganancia de transmisión (txGain): -20, -10 y 0 dB. Controla cuánta potencia se envía por la antena transmisora. Valores más altos implican más potencia enviada, pero también pueden saturar la cadena de recepción si son excesivos.
- Ganancia de recepción (rxGain): 10, 20 y 30 dB. Controla cuánto se amplifica la señal captada por la antena receptora antes de digitalizarla. Valores demasiado bajos dejan la señal cerca del nivel de ruido, y valores demasiado altos pueden saturar el conversor analógico-digital.
- Amplitud de transmisión ( $A_{tx}$ ): 0,2, 0,5 y 0,7. Es un factor de escala aplicado en software a la señal antes de transmitirla, independiente de la ganancia de RF. Ajusta el nivel de la señal digital enviada al dispositivo.

El número total de combinaciones evaluadas fue  $3 \times 3 \times 3 \times 3 = 81$ . Para cada una de ellas se realizaron 10 capturas consecutivas, lo que permite calcular no solo el resultado de una captura aislada, sino promediar sobre varias para obtener una estimación más robusta.

### 6.7.1 MÉTRICAS DE EVALUACIÓN

Cada configuración se evaluó mediante tres métricas:

La primera es la **potencia media recibida**, que indica cuánta energía llega al receptor en promedio:

$$P_{rx} = 10 \log_{10} \left( \frac{1}{K} \sum_{k=1}^K \overline{|r_k[n]^2|} \right) [dB]$$

La segunda es la **SNR espectral**, que mide la calidad de la señal recibida comparando la energía dentro de la banda de la señal ( $[f_l, f_h] = [7,5,25,5]$  MHz) con la energía fuera de ella, en una zona que debería ser solo ruido (con un margen de guarda de 2 MHz en cada extremo para evitar contabilizar los bordes de la banda):

$$\text{SNR} = 10 \log_{10} \left( \frac{\sum_{k \in \mathcal{B}_s} |X[k]|^2}{\sum_{k \in \mathcal{B}_n} |X[k]|^2} \right) [\text{dB}]$$

Cuanto mayor es la SNR, más claramente destaca la señal del dron frente al ruido de fondo, lo que es directamente relevante para los algoritmos de detección que se aplicarán en el capítulo 7 (Capítulo 7. ).

La tercera es la **estabilidad**, definida como la desviación típica de la potencia recibida entre las 10 capturas:

$$\sigma_p = \text{std}\{10 \log_{10}(|r_k[\bar{n}]|^2), k = 1, \dots, K\} [\text{dB}]$$

Un valor de  $\sigma_p$  bajo indica que la potencia recibida no varía mucho de una captura a otra, lo que es deseable: si el sistema es muy inestable, los algoritmos de detección tendrían que enfrentarse a una señal cuyo nivel cambia de forma impredecible entre capturas, lo que dificultaría su funcionamiento.

## 6.7.2 RESULTADOS DEL BARRIDO

La Tabla 4 recoge las diez mejores configuraciones ordenadas por SNR, junto con la configuración óptima global.

| (MHz) | txGain (dB) | rxGain (dB) | A_tx | P <sub>rx</sub> (dB) | SNR (dB) | $\sigma_P$ (dB) |
|-------|-------------|-------------|------|----------------------|----------|-----------------|
| 868   | 0           | 20          | 0,5  | -9,04                | 42,73    | 0,0113          |
| 868   | -10         | 30          | 0,5  | -8,91                | 41,88    | 0,0052          |
| 868   | 0           | 10          | 0,7  | -15,42               | 40,38    | 0,0029          |
| 868   | -10         | 20          | 0,7  | -15,71               | 40,32    | 0,0097          |
| 868   | -20         | 30          | 0,7  | -15,61               | 39,13    | 0,0030          |
| 868   | 0           | 10          | 0,5  | -18,31               | 38,75    | 0,0020          |
| 868   | 0           | 20          | 0,2  | -16,91               | 38,33    | 0,0045          |
| 868   | -10         | 20          | 0,5  | -18,69               | 38,32    | 0,0048          |
| 868   | -10         | 30          | 0,2  | -16,81               | 37,34    | 0,0058          |
| 868   | -20         | 30          | 0,5  | -18,64               | 36,78    | 0,0014          |

*Tabla 4: Las diez mejores configuraciones del barrido de parámetros, ordenadas por SNR descendente.*

El primer resultado destacable es que las diez mejores configuraciones corresponden en su totalidad a la frecuencia portadora de 868 MHz, sin que ninguna configuración a 433 MHz ni a 2.450 MHz aparezca entre ellas. Esto no es sorprendente a la vista de los resultados del apartado 6.2: la comparación con señal de tono puro ya evidenciaba que a 2,4 GHz las pérdidas de propagación son mayores [19] y el nivel de interferencias del entorno es considerablemente más elevado [20], [21]. Los resultados del barrido confirman que este comportamiento se mantiene también con señales OFDM, y que 868 MHz es la frecuencia de trabajo más adecuada en el entorno de laboratorio en el que se han realizado las pruebas.

En cuanto a los parámetros de ganancia y amplitud, la configuración que ofrece la mayor SNR (42,73 dB) combina txGain = 0 dB, rxGain = 20 dB y A\_tx = 0,5. Es interesante observar que no es la combinación con mayor potencia recibida (ese lugar lo ocupa la segunda configuración, con rxGain = 30 dB y una potencia 0,1 dB superior), sino la que mejor equilibra potencia y ruido. Subir rxGain de 20 a 30 dB amplifica la señal, pero amplifica también el ruido de la cadena de recepción en la misma proporción, de modo que la SNR resultante es ligeramente inferior. Este comportamiento ilustra bien por qué la SNR es una métrica más informativa que la potencia recibida a la hora de evaluar la calidad de una configuración.

La estabilidad de la configuración óptima ( $\sigma_p = 0,011$  dB) es la más baja entre las tres con mayor SNR, lo que indica que la potencia recibida se mantiene prácticamente constante entre las 10 capturas realizadas. Esta combinación de alta SNR y buena estabilidad la convierte en la elección natural para la transmisión del conjunto completo de señales del proyecto, y sus parámetros se adoptan directamente en el siguiente apartado.

## **6.8 TRANSMISIÓN Y RECEPCIÓN DEL CONJUNTO COMPLETO DE SEÑALES**

Con la configuración física y los parámetros de operación ya fijados, se procedió a transmitir y recibir mediante el ADALM-PLUTO el conjunto completo de señales preparado en el Capítulo 5. Las tres categorías de señal (simuladas, reales y reales acotadas) y los cinco tipos de dron (M250, V2, OcuSync 2, Mini 2 y DJI Neo) se procesaron de forma sistemática, con la única diferencia de que las señales simuladas disponen de una única muestra por dron, mientras que las reales y las reales acotadas disponen de tres.

Los parámetros de transmisión empleados son los obtenidos del barrido del apartado 6.7:  $f_c = 868\text{MHz}$ , txGain = 0 dB, rxGain = 20 dB y  $A_{tx} = 0,5$ . Para cada señal se realizaron 10 capturas consecutivas, almacenadas como una matriz de  $N \times 10$  muestras complejas, lo que permite disponer de varias realizaciones independientes de la misma señal recibida para su uso posterior en los algoritmos de detección.

El procedimiento seguido para cada señal es el siguiente: se normaliza la señal al valor de pico y se escala por  $A_{tx}$ ; se verifica que el número de muestras sea par, requisito del dispositivo; se configura el transmisor y el receptor con los parámetros fijados; se inicia la transmisión continua mediante transmitRepeat; y se capturan las 10 tramas en recepción. Una vez completada la captura, se liberan los objetos Tx y Rx antes de procesar la siguiente señal, para evitar conflictos de acceso al dispositivo. Los resultados se guardan en ficheros individuales por señal y categoría, con los campos rxSig, txSig, fs\_pluto, fc, txGain, rxGain e iterations, que serán cargados directamente por los algoritmos del capítulo 7 (Capítulo 7. ).

Con esto queda completo el proceso de transmisión y recepción: el ADALM-PLUTO ha actuado como canal RF controlado entre una señal generada o capturada previamente y su versión recibida, lista para ser analizada por los algoritmos de detección y caracterización que se describen en el capítulo siguiente.

## Capítulo 7. DESARROLLO DE LOS ALGORITMOS DE DETECCIÓN Y CARACTERIZACIÓN

### 7.1 INTRODUCCIÓN Y PLANTEAMIENTO DEL PROBLEMA

El objetivo de los algoritmos desarrollados en este capítulo es estimar, a partir del espectro de magnitud de una señal recibida, los límites inferior y superior de la banda ocupada por dicha señal,  $\hat{f}_l$  y  $\hat{f}_h$ , y evaluar en qué medida esas estimaciones se corresponden con los valores de referencia obtenidos en el capítulo 5 (Capítulo 5. ).

En condiciones ideales, sin ruido y con un espectro perfectamente plano dentro de la banda, los bordes de la señal son transiciones abruptas y fácilmente identificables. En condiciones reales, sin embargo, el espectro presenta irregularidades dentro de la banda (picos, estructura interna, variaciones de amplitud), y fuera de ella existe un nivel de ruido de fondo que difumina los bordes y complica su localización. Esta dificultad se agrava a medida que la relación señal-ruido disminuye: para SNR elevadas, los bordes de la banda destacan claramente sobre el ruido; para SNR bajas, la transición entre la banda y el ruido puede ser tan gradual que resulta difícil determinar con precisión en qué frecuencia se produce.

Para evaluar el comportamiento de los algoritmos bajo distintas condiciones de ruido, se añade de forma controlada ruido gaussiano blanco aditivo (AWGN, Additive White Gaussian Noise) a la señal recibida mediante la función `awgn` de MATLAB, ajustando la SNR a distintos valores. De este modo, es posible estudiar cómo varía la precisión de la estimación en función del nivel de ruido, sin depender de las condiciones ambientales del entorno de captura.

La calidad de cada estimación se evalúa comparando los valores estimados  $\hat{f}_l$  y  $\hat{f}_h$  con los valores de referencia  $f_l^{ref}$  y  $f_h^{ref}$  de la Tabla 3, aplicando un margen de tolerancia porcentual  $\epsilon$  directamente sobre cada frecuencia de corte. Una estimación se considera válida si se cumplen simultáneamente las dos condiciones siguientes:

$$f_l^{ref} \cdot (1 - \epsilon) \leq \hat{f}_l \leq f_l^{ref} \cdot (1 + \epsilon)$$
$$f_h^{ref} \cdot (1 - \epsilon) \leq \hat{f}_h \leq f_h^{ref} \cdot (1 + \epsilon)$$

El porcentaje de estimaciones válidas sobre el total de muestras evaluadas constituye la métrica principal de rendimiento de cada algoritmo. Este criterio permite cuantificar de forma objetiva la robustez del estimador ante distintos niveles de ruido y distintas señales, y comparar directamente el rendimiento de los dos algoritmos desarrollados.

## **7.2 ELEMENTOS COMUNES A LOS DOS ALGORITMOS**

Antes de describir cada algoritmo de forma individual, conviene presentar los elementos que ambos comparten, ya que constituyen la base sobre la que se construyen.

### **7.2.1 MODELO DE RUIDO**

Con el fin de evaluar el comportamiento de los algoritmos bajo distintas condiciones de canal, se añade ruido gaussiano blanco aditivo (AWGN, Additive White Gaussian Noise) a la señal recibida mediante la función `awgn` de MATLAB, que permite fijar la SNR de la señal resultante a un valor determinado tomando como referencia la potencia de la propia señal. Este enfoque permite estudiar de forma controlada y reproducible cómo varía la precisión de la estimación en función del nivel de ruido, sin depender de las condiciones ambientales del entorno de captura. El nivel de SNR es un parámetro configurable del sistema, de forma que pueden realizarse pruebas a distintos niveles de degradación para caracterizar los límites de operación de cada algoritmo.

### **7.2.2 SUAVIZADO DEL ESPECTRO**

Ambos algoritmos operan sobre el espectro de magnitud de la señal recibida, no directamente sobre la señal temporal. Sin embargo, el espectro crudo presenta una variabilidad muestra a muestra elevada, incluso en ausencia de ruido añadido: al tratarse de una señal OFDM-like con coeficientes aleatorios, su espectro de magnitud fluctúa rápidamente de bin a bin dentro de la banda, lo que dificultaría la detección de los bordes si se trabajara directamente sobre él. Por este motivo, antes de aplicar ningún criterio de detección, ambos algoritmos aplican una etapa de suavizado espectral cuyo objetivo es obtener una representación más regular del espectro, en la que los bordes de la banda sean más fácilmente identificables. Los dos algoritmos difieren en el método de suavizado empleado.

### **7.2.3 DETECCIÓN CON MÚLTIPLES STEPS Y ESTIMACIÓN FINAL POR MEDIANA**

Tanto el Algoritmo 1 como el Algoritmo 2 basan su detección en el cálculo de diferencias entre valores del espectro suavizado separados por una distancia determinada, denominada step. El valor del step influye directamente en la sensibilidad del detector: un step pequeño es más sensible a variaciones locales pero también más susceptible al ruido residual; un step grande promedia más información pero puede perder precisión en la localización exacta del borde.

Dado que no existe a priori un valor de step óptimo universal para todas las señales y todos los niveles de ruido, ambos algoritmos se ejecutan de forma repetida con un conjunto de valores de step distintos, obteniendo una estimación de  $\hat{f}_l$  y  $\hat{f}_h$  para cada uno de ellos. La estimación final se obtiene como la mediana de todos los valores estimados:

$$\hat{f}_l^{final} = \text{mediana}(\hat{f}_l^{(1)}, \hat{f}_l^{(2)}, \dots, \hat{f}_l^{(S)})$$
$$\hat{f}_h^{final} = \text{mediana}(\hat{f}_h^{(1)}, \hat{f}_h^{(2)}, \dots, \hat{f}_h^{(S)})$$

donde  $S$  es el número de valores de step evaluados. La elección de la mediana como estimador de consenso responde a su conocida robustez frente a valores atípicos: si algún step concreto produce una estimación muy desviada del valor real (por ejemplo, por coincidir con un pico de ruido en una zona crítica del espectro), ese valor extremo queda automáticamente neutralizado por la mediana, sin necesidad de identificar ni descartar manualmente qué step fue el problemático. Esta estrategia fue desarrollada como parte del diseño propio de este TFG, como mecanismo sencillo y efectivo para mejorar la robustez de la estimación frente a la variabilidad introducida por el ruido.

#### 7.2.4 MODOS DE DETECCIÓN

Ambos algoritmos implementan dos modos de operación, que determinan el orden en que se buscan los bordes de la banda. En el **Modo 1**, se busca primero el borde de subida (el punto donde la energía del espectro aumenta bruscamente, correspondiente a  $f_l$ ) y, a partir de ese punto, se busca el borde de bajada (correspondiente a  $f_h$ ) en la región posterior. En el **Modo 2**, el orden se invierte: se busca primero el borde de bajada y, a partir de ahí, se retrocede para encontrar el borde de subida. La existencia de dos modos permite adaptar el algoritmo a señales cuya estructura espectral favorece una u otra dirección de búsqueda, y comparar si el orden de detección afecta a la precisión de la estimación.

### ***7.3 ALGORITMO 1: SUAVIZADO POR MEDIA MÓVIL Y DETECCIÓN POR DERIVADA***

El primer algoritmo desarrollado basa su funcionamiento en dos ideas principales: suavizar el espectro mediante una media móvil para reducir la variabilidad bin a bin, y calcular la derivada de ese espectro suavizado para identificar los puntos donde se producen los cambios bruscos de energía que corresponden a los bordes de la banda.

### 7.3.1 SUAVIZADO POR MEDIA MÓVIL

La primera etapa del algoritmo aplica una media móvil (función `movmean` de MATLAB [24]) sobre el espectro de magnitud de la señal ruidosa. Según la documentación oficial, `movmean(S, W)` devuelve un array de medias locales calculadas sobre una ventana deslizante de longitud  $W$ , centrada en cada elemento; cuando la ventana llega a los extremos del array, se trunca automáticamente para no sobrepasar los límites [24].

El tamaño de ventana  $W$  es un parámetro configurable del algoritmo, y puede especificarse de dos formas distintas según las necesidades del usuario. La primera opción consiste en fijar directamente el número de muestras de la ventana como un valor absoluto, lo que resulta conveniente cuando se conoce de antemano la resolución frecuencial de la señal y se quiere un control preciso sobre el suavizado aplicado. La segunda opción consiste en expresar el tamaño de ventana como un porcentaje del número total de muestras  $N$ , de forma que:

$$W = \text{round}\left(\frac{p}{100} \cdot N\right)$$

donde  $p$  es el porcentaje deseado. Esta segunda opción tiene la ventaja de que el nivel de suavizado se adapta automáticamente a señales de distinta longitud, manteniendo una proporción constante entre el tamaño de ventana y el total de muestras. En cualquiera de los dos casos, se impone un valor mínimo de 10 muestras para evitar ventanas degeneradas. En las pruebas realizadas en este TFG se empleó la opción porcentual con  $p = 1\%$ , lo que para  $N = 961.196$  muestras resulta en una ventana de aproximadamente 9.600 muestras.

Una vez determinado el tamaño de ventana  $W$ , el suavizado se aplica sobre el espectro de magnitud  $S[k]$ , donde  $k = 0, 1, \dots, N - 1$  es el índice de bin frecuencial. El espectro suavizado  $\tilde{S}[k]$  se obtiene sustituyendo cada bin por la media aritmética de los bins contenidos en una ventana centrada en él:

$$\tilde{S}[k] = \frac{1}{W} \sum_{m=-W/2}^{\lfloor W/2 \rfloor} S[k + m]$$

donde los índices fuera del rango  $[0, N - 1]$  se tratan truncando la ventana en los extremos, tal y como implementa la función `movmean` [24]. El resultado es un espectro de la misma longitud que el original, en el que las fluctuaciones rápidas bin a bin quedan atenuadas, preservando las transiciones lentas como los bordes de la banda ocupada, que son precisamente las que el algoritmo necesita detectar en la etapa siguiente.

### 7.3.2 CÁLCULO DE LA DERIVADA CON STEP VARIABLE

La función de cálculo de la derivada recibe como entradas el espectro suavizado  $\tilde{S}[k]$ , el valor de `step`, el tamaño de ventana principal  $W$  y el porcentaje de suavizado de la derivada  $p_{deriv}$ .

Lo primero que hace es inicializar un vector de ceros de la misma longitud que el espectro:

```
derivada = zeros(length(S_smooth), 1);
```

Esto es importante: el vector de la derivada tiene exactamente  $N$  elementos, igual que el espectro. Los últimos `step` elementos quedarán a cero porque no tienen muestra futura con la que calcular la diferencia, ya que el bucle solo llega hasta  $N - \text{step}$ . Esto evita acceder a posiciones fuera del vector, pero implica que la derivada no está definida en esa región final del espectro.

A continuación, el bucle calcula la diferencia hacia adelante para cada bin:

```
for i = 1:length(S_smooth) - step
    derivada(i) = S_smooth(i + step) - S_smooth(i);
end
```

Para entender qué hace esto geoméricamente: si el espectro  $\tilde{S}[k]$  es plano (dentro del ruido o dentro de la banda), la diferencia entre dos puntos separados por  $\text{step}$  bins será pequeña y próxima a cero. En cambio, en el borde de subida de la banda, donde el espectro pasa de un nivel bajo (ruido) a un nivel alto (señal) de forma relativamente brusca,  $\tilde{S}[k + \text{step}]$  será notablemente mayor que  $\tilde{S}[k]$ , produciendo un valor positivo grande. En el borde de bajada ocurre lo contrario:  $\tilde{S}[k + \text{step}]$  será menor que  $\tilde{S}[k]$ , produciendo un valor negativo grande. Por tanto, el vector derivada tendrá un pico positivo en torno al borde de subida y un pico negativo en torno al borde de bajada, siendo aproximadamente nulo en el resto.

El efecto del parámetro  $\text{step}$  sobre esta representación es el siguiente: con  $\text{step}=1$ , las diferencias son muy locales y la derivada puede presentar mucha variabilidad debida al ruido residual del espectro; con  $\text{step}$  grande, cada diferencia "salta" por encima de más bins y el pico de la derivada queda más suavizado pero también más desplazado respecto al borde real, ya que la diferencia se evalúa entre un punto ya dentro de la banda y otro todavía fuera, o viceversa. Este desplazamiento es el precio a pagar por la mayor robustez frente al ruido que ofrece un  $\text{step}$  grande.

Una vez calculada la derivada bruta, se aplica una segunda etapa de suavizado sobre ella mediante  $\text{movmean}$ . El tamaño de esa ventana se calcula como:

```
ventana_deriv = round(ventana / porc_suavizado_deriv);  
ventana_deriv = max(ventana_deriv, 3);
```

Es decir, la ventana de suavizado de la derivada es una fracción de la ventana principal: si la ventana principal era  $W$  y el porcentaje es  $p_{\text{deriv}} = 10$ , la ventana de la derivada será  $W_{\text{deriv}} = W/10$ . El  $\max(\dots, 3)$  garantiza que, aunque la ventana principal sea muy pequeña, la ventana de la derivada nunca sea inferior a 3 muestras, evitando un suavizado degenerado que no tuviera ningún efecto real.

El motivo de aplicar este segundo suavizado es que la derivada bruta, aunque ya se calcula sobre el espectro suavizado  $\tilde{S}[k]$ , puede seguir presentando picos espurios debidos a irregularidades residuales del espectro dentro de la banda. Si se buscara el máximo de la derivada bruta directamente, uno de esos picos espurios podría ser detectado erróneamente como el borde de la banda. Al suavizar también la derivada, estos picos quedan atenuados y el pico dominante que corresponde al borde real de la banda resulta más fácilmente identificable:

```
derivada_smooth = movmean(derivada, ventana_deriv);
```

El resultado final, `derivada_smooth`, es la derivada doblemente suavizada  $\tilde{d}[k]$ : primero porque se calculó sobre el espectro ya suavizado  $\tilde{S}[k]$ , y segundo porque se le aplicó un `movmean` adicional. Es esta versión la que se emplea en la etapa de detección de bordes.

### 7.3.3 DETECCIÓN DE BORDES POR UMBRAL

La detección de los bordes de la banda se realiza buscando, sobre la derivada suavizada  $\tilde{d}[k]$ , las posiciones donde se producen sus picos positivo y negativo más relevantes. Para ello, primero se calculan dos umbrales a partir de los valores extremos de la propia derivada:

```
umbral_subida = porc_umbral * max(derivada_smooth);  
umbral_bajada = porc_umbral * min(derivada_smooth);
```

El umbral de subida es un porcentaje (`porc_umbral`) del valor máximo de la derivada suavizada, y el umbral de bajada es ese mismo porcentaje aplicado al valor mínimo (que es negativo). Con `porc_umbral = 1`, los umbrales coinciden exactamente con el máximo y el mínimo de la derivada; con un valor menor que 1, los umbrales se vuelven más permisivos, exigiendo solo una fracción del pico máximo para considerar que se ha encontrado un borde. Esta forma de definir el umbral como proporción del propio máximo (en lugar de un valor absoluto fijo) tiene la ventaja de adaptarse automáticamente a la amplitud de cada señal, sin necesidad de recalibrar el umbral para cada caso.

El algoritmo distingue dos modos de búsqueda.

### 7.3.3.1 Modo 1: subida primero, bajada después

```
idx_subida_temp = find(derivada_smooth > umbral_subida, 1, 'first');  
[~, idx_peak] = max(S_smooth);  
if isempty(idx_subida_temp)  
    [~, idx_subida_temp] = max(derivada_smooth);  
end  
idx_subida = idx_subida_temp;
```

Primero se busca el primer bin, recorriendo el espectro de izquierda a derecha, donde la derivada suavizada supera el umbral de subida. Ese bin se interpreta como el borde de subida  $\hat{f}_l$ . Si ningún bin llega a superar el umbral (algo que puede ocurrir si el umbral es muy exigente o la señal está muy degradada por el ruido), el algoritmo recurre a una alternativa de seguridad: toma directamente la posición del valor máximo de la derivada, sea cual sea su magnitud. Esta lógica de respaldo (if isempty(...)) es importante porque garantiza que el algoritmo siempre devuelva una estimación, en lugar de fallar o devolver un valor vacío cuando las condiciones de ruido son desfavorables.

Paralelamente, se localiza el pico del espectro suavizado original,  $\text{idx\_peak}$ , que se usará como punto de referencia para acotar la búsqueda del borde de bajada:

```
region_bajada = derivada_smooth(idx_peak:end);  
idx_bajada_temp = find(region_bajada < umbral_bajada, 1, 'first');  
if isempty(idx_bajada_temp)  
    [~, idx_min] = min(derivada_smooth(idx_peak:end));  
    idx_bajada_temp = idx_min;  
end  
idx_bajada = idx_peak + idx_bajada_temp - 1;
```

La búsqueda del borde de bajada se restringe deliberadamente a la región posterior al pico del espectro (`derivada_smooth(idx_peak:end)`), en lugar de buscar en todo el vector. Esto evita que el algoritmo confunda el propio borde de subida (que también produce un descenso brusco de la derivada justo después de su pico positivo) con el borde de bajada real, que debe estar más adelante, después de toda la banda. Dentro de esa región acotada, se busca el primer bin donde la derivada cae por debajo del umbral de bajada; si no se encuentra ninguno, de nuevo se recurre al mínimo absoluto de la derivada en esa región como alternativa de seguridad. Como la búsqueda se hizo sobre un subvector que empieza en `idx_peak`, el índice resultante debe corregirse sumando `idx_peak - 1` para referenciarlo al vector completo.

### 7.3.3.2 Modo 2: bajada primero, subida antes

```
idx_bajada_temp = find(derivada_smooth < umbral_bajada, 1, 'first');
if isempty(idx_bajada_temp)
    [~, idx_bajada_temp] = min(derivada_smooth);
end
idx_bajada = idx_bajada_temp;

region_subida = derivada_smooth(1:idx_bajada);
idx_subida_temp = find(region_subida > umbral_subida, 1, 'last');
if isempty(idx_subida_temp)
    [~, idx_max] = max(derivada_smooth(1:idx_bajada));
    idx_subida_temp = idx_max;
end
idx_subida = idx_subida_temp;
```

El Modo 2 invierte el orden de búsqueda: primero localiza, recorriendo todo el vector desde el principio, el primer bin donde la derivada cae por debajo del umbral de bajada, y lo toma como  $\hat{f}_h$ . A continuación, restringe la búsqueda del borde de subida a la región anterior a ese punto (`derivada_smooth(1:idx_bajada)`), buscando esta vez el último bin (no el primero) donde la derivada supera el umbral de subida, ya que se quiere el borde más cercano a la bajada ya encontrada, no uno anterior. Al igual que en el Modo 1, ambas búsquedas incluyen su correspondiente alternativa de seguridad basada en el valor extremo absoluto cuando no se encuentra ningún bin que supere el umbral.

### 7.3.3.3 Comprobación de consistencia final

```
if idx_subida > idx_bajada
    temp = idx_subida;
    idx_subida = idx_bajada;
    idx_bajada = temp;
end
```

Como medida final de seguridad, independientemente del modo empleado, se comprueba que el índice de subida sea efectivamente anterior al de bajada; si por alguna razón (por ejemplo, una combinación de ruido y umbrales muy exigentes) el algoritmo hubiera devuelto los índices en orden inverso, esta comprobación los intercambia automáticamente, garantizando que  $\hat{f}_l < \hat{f}_h$  siempre se cumpla en el resultado final.

### 7.3.4 RESULTADOS EXPERIMENTALES

Para evaluar el comportamiento del Algoritmo 1 se aplicó sobre la señal recibida por el ADALM-PLUTO correspondiente a OcuSync 2 (valores de referencia  $f_l^{ref} = 7,5$  MHz,  $f_h^{ref} = 25,5$  MHz, según la Tabla 3), añadiendo ruido AWGN a dos niveles de SNR distintos: -15 dB y -23 dB, y empleando el Modo 1 de detección con el conjunto de steps {1, 50, 100, 500, 1000, 2500, 5000, 7500, 10000, 12323}.

### 7.3.4.1 Caso $SNR = -15$ dB

| Step                   | $\hat{f}_l$ (MHz) | $\hat{f}_h$ (MHz) | Err. $f_l$ /ref (%) | Err. $f_h$ /ref (%) | Err. $f_l$ / $f_s$ (%) | Err. $f_h$ / $f_s$ (%) |
|------------------------|-------------------|-------------------|---------------------|---------------------|------------------------|------------------------|
| 1                      | 7,603             | 25,470            | 1,38                | 0,12                | 0,17                   | 0,05                   |
| 50                     | 7,602             | 25,467            | 1,36                | 0,13                | 0,17                   | 0,05                   |
| 100                    | 7,602             | 25,466            | 1,36                | 0,13                | 0,17                   | 0,05                   |
| 500                    | 7,577             | 10,981            | 1,03                | 56,94               | 0,13                   | 23,63                  |
| 1000                   | 7,553             | 25,454            | 0,70                | 0,18                | 0,09                   | 0,08                   |
| 2500                   | 7,306             | 25,358            | 2,58                | 0,56                | 0,32                   | 0,23                   |
| 5000                   | 7,299             | 25,195            | 2,67                | 1,20                | 0,33                   | 0,50                   |
| 7500                   | 7,181             | 25,161            | 4,25                | 1,33                | 0,52                   | 0,55                   |
| 10000                  | 7,172             | 25,024            | 4,37                | 1,87                | 0,53                   | 0,78                   |
| 12323                  | 7,180             | 24,884            | 4,26                | 2,41                | 0,52                   | 1,00                   |
| <b>Mediana (final)</b> | <b>7,430</b>      | <b>25,276</b>     | <b>0,94</b>         | <b>0,88</b>         | <b>0,11</b>            | <b>0,36</b>            |

Tabla 5: Resultados del Algoritmo 1 por step, para  $SNR = -15$  dB.

Para este nivel de ruido, la estimación final por mediana arroja un error relativo de apenas el 0,94% en  $f_l$  y el 0,88% en  $f_h$  respecto a los valores de referencia (equivalentes a un 0,11% y un 0,36% del ancho de banda completo disponible,  $f_s = 61,44$  MHz), lo que constituye un resultado muy satisfactorio. Se observa además que, salvo el resultado anómalo del step=500 en  $f_h$  (10,98 MHz, con un error del 56,94% respecto al valor de referencia y del 23,63% respecto a  $f_s$ , claramente desviado del resto), todos los demás steps convergen a valores razonablemente próximos a la referencia, con errores individuales que en ningún caso superan el 4,4% respecto al valor teórico ni el 0,55% respecto al ancho de banda completo. Esto demuestra que, con esta SNR, el algoritmo es robusto incluso sin necesidad del mecanismo de mediana: la mayoría de steps individuales ya proporcionan una buena estimación por sí solos.

### 7.3.4.2 Caso $SNR = -23$ dB

| Step                   | $\hat{f}_l$ (MHz) | $\hat{f}_h$ (MHz) | Err. $f_l$ /ref (%) | Err. $f_h$ /ref (%) | Err. $f_l$ / $f_s$ (%) | Err. $f_h$ / $f_s$ (%) |
|------------------------|-------------------|-------------------|---------------------|---------------------|------------------------|------------------------|
| 1                      | 7,715             | 21,678            | 2,87                | 14,99               | 0,35                   | 6,22                   |
| 50                     | 7,714             | 21,675            | 2,85                | 15,00               | 0,35                   | 6,23                   |
| 100                    | 7,713             | 21,672            | 2,84                | 15,01               | 0,35                   | 6,23                   |
| 500                    | -20,877           | 21,658            | 378,4               | 15,07               | 46,19                  | 6,26                   |
| 1000                   | -20,898           | 21,635            | 378,6               | 15,16               | 46,22                  | 6,29                   |
| 2500                   | -20,975           | 30,483            | 379,7               | 19,54               | 46,35                  | 8,11                   |
| 5000                   | 12,306            | 24,910            | 64,08               | 2,32                | 7,82                   | 0,96                   |
| 7500                   | 24,873            | 26,345            | 231,6               | 3,31                | 28,28                  | 1,38                   |
| 10000                  | 24,878            | 26,257            | 231,7               | 2,97                | 28,29                  | 1,23                   |
| 12323                  | 12,139            | 24,880            | 61,86               | 2,43                | 7,55                   | 1,01                   |
| <b>Mediana (final)</b> | <b>7,715</b>      | <b>23,279</b>     | <b>2,86</b>         | <b>8,71</b>         | <b>0,35</b>            | <b>3,62</b>            |

Tabla 6: Resultados del Algoritmo 1 por step, para  $SNR = -23$  dB.

A este nivel de ruido, considerablemente más severo, el comportamiento es notablemente distinto. La estimación de  $f_l$  por mediana sigue siendo buena (error relativo del 2,86% respecto al valor de referencia, equivalente a un 0,35% del ancho de banda completo disponible), pero la de  $f_h$  se degrada de forma apreciable (error relativo del 8,71% respecto al valor de referencia, un 3,62% respecto al ancho de banda completo).

Resulta especialmente ilustrativo observar el comportamiento individual de cada step: para  $f_l$ , únicamente los tres steps más pequeños (1, 50, 100) ofrecen una estimación coherente con la referencia (en torno a 7,71 MHz, con errores inferiores al 3% respecto al valor teórico y al 0,35% respecto al ancho de banda completo).

Los steps intermedios (500, 1000, 2500) producen estimaciones completamente erróneas, con valores negativos en torno a -20,9 MHz, cuyo error relativo al valor de referencia supera el 378% (un 46% del ancho de banda completo, una magnitud que por sí sola ya delata lo desproporcionado de la estimación), mientras que los steps más grandes (5000 a 12323) oscilan entre 12,1 y 24,9 MHz, tampoco cercanos al valor real, con errores de entre el 62% y el 232% respecto a la referencia (entre el 7,6% y el 28,3% del ancho de banda completo).

A pesar de que solo 3 de los 10 steps producen una estimación correcta, la mediana consigue identificar igualmente el valor adecuado, porque esos tres resultados correctos ocupan la posición central al ordenar las diez estimaciones: los tres valores muy negativos quedan por debajo, y los cuatro valores de steps grandes quedan por encima, dejando exactamente a los tres steps pequeños en el centro de la distribución. Este es un ejemplo claro y favorable del comportamiento robusto de la mediana descrito en el apartado 7.2.3: aunque la mayoría de las estimaciones individuales sean erróneas, el resultado final puede seguir siendo correcto si los valores acertados se concentran en torno al centro de la distribución ordenada.

Para  $f_h$ , en cambio, la situación es menos favorable: los steps pequeños (1 a 1000) convergen de forma consistente en torno a 21,6-21,7 MHz, un valor que subestima la referencia real de 25,5 MHz con un error de en torno al 15% respecto al valor teórico (un 6,2-6,3% del ancho de banda completo), mientras que los steps grandes (5000 a 12323, excluyendo el valor atípico de 30,48 MHz en step=2500) se sitúan en torno a 24,9-26,3 MHz, más próximos al valor correcto, con errores de entre el 2,3% y el 3,3% respecto a la referencia (entre el 0,96% y el 1,38% del ancho de banda completo).

Al tomar la mediana entre estos dos grupos sistemáticamente distintos, el resultado final (23,28 MHz) queda en un punto intermedio que no representa fielmente a ninguno de los dos grupos, lo que explica el error más elevado obtenido para  $f_h$  en este caso. Esto pone de manifiesto una limitación del mecanismo de mediana: funciona bien cuando los aciertos se concentran en el centro de la distribución de estimaciones, pero pierde eficacia cuando existen dos grupos de resultados sistemáticamente distintos (en lugar de un grupo correcto y valores atípicos dispersos), como ocurre aquí entre los steps pequeños y los steps grandes.

### 7.3.4.3 Recorrido visual del algoritmo ( $SNR = -15$ dB, $step = 12323$ )

Con fines didácticos, las Figuras Figura 23 a Figura 28 muestran las distintas etapas internas del Algoritmo 1 para el caso de  $SNR = -15$  dB, fijando el step en 12323 (el mayor del conjunto evaluado), lo que permite visualizar con claridad el funcionamiento del detector.

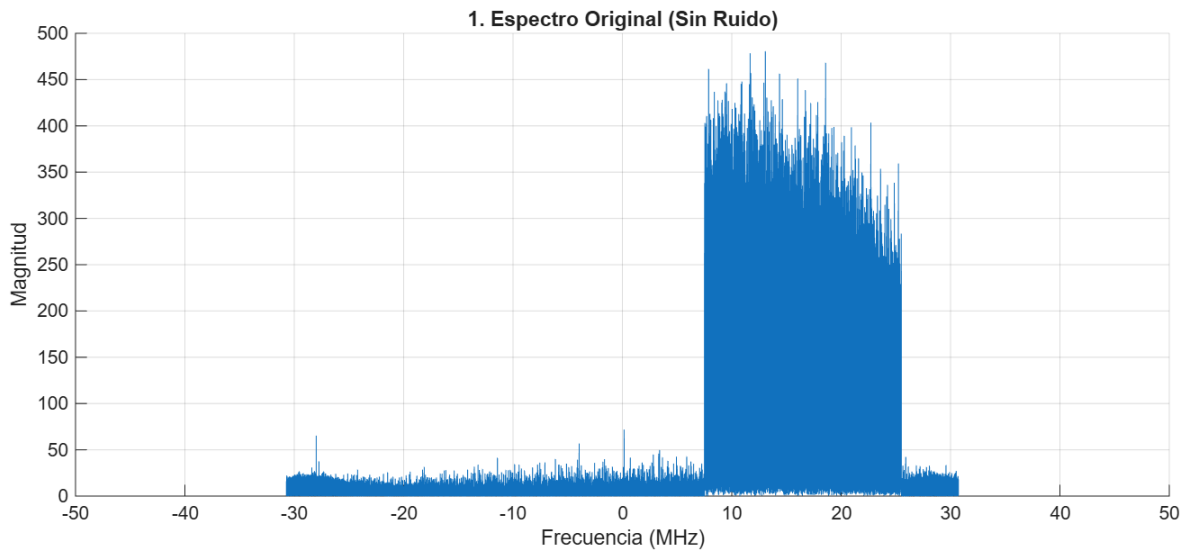


Figura 23: Espectro de magnitud original, sin ruido añadido.

La Figura 23 muestra el espectro de referencia, sin ruido, en el que la banda ocupada (aproximadamente entre 7,5 y 25,5 MHz) es perfectamente identificable a simple vista, con un nivel de magnitud que alcanza picos de hasta 480 dentro de la banda frente a un fondo de ruido residual inferior a 60.

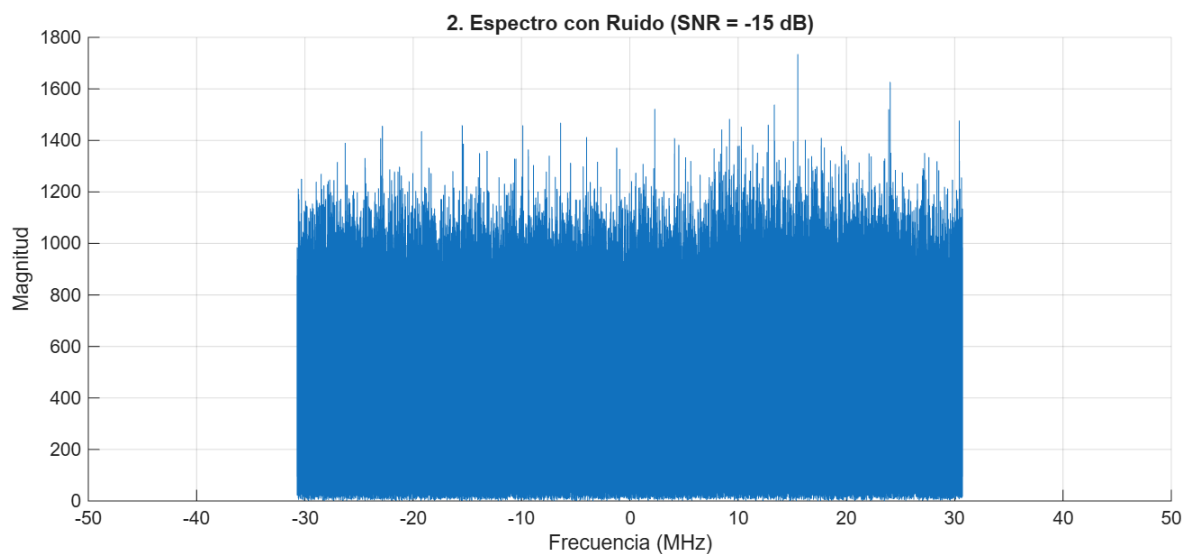


Figura 24: Espectro de magnitud tras añadir ruido AWGN con SNR = -15 dB.

Tras añadir ruido a -15 dB (Figura 24), la situación cambia drásticamente: el nivel de ruido (en torno a 1.000-1.400) supera ampliamente la magnitud que tenía la señal original sin ruido, y la banda ocupada deja de ser distinguible a simple vista en el espectro de magnitud. Este es precisamente el escenario que justifica la necesidad de un algoritmo de detección: a este nivel de degradación, ya no es posible identificar los bordes de la banda mediante simple inspección visual del espectro de magnitud.

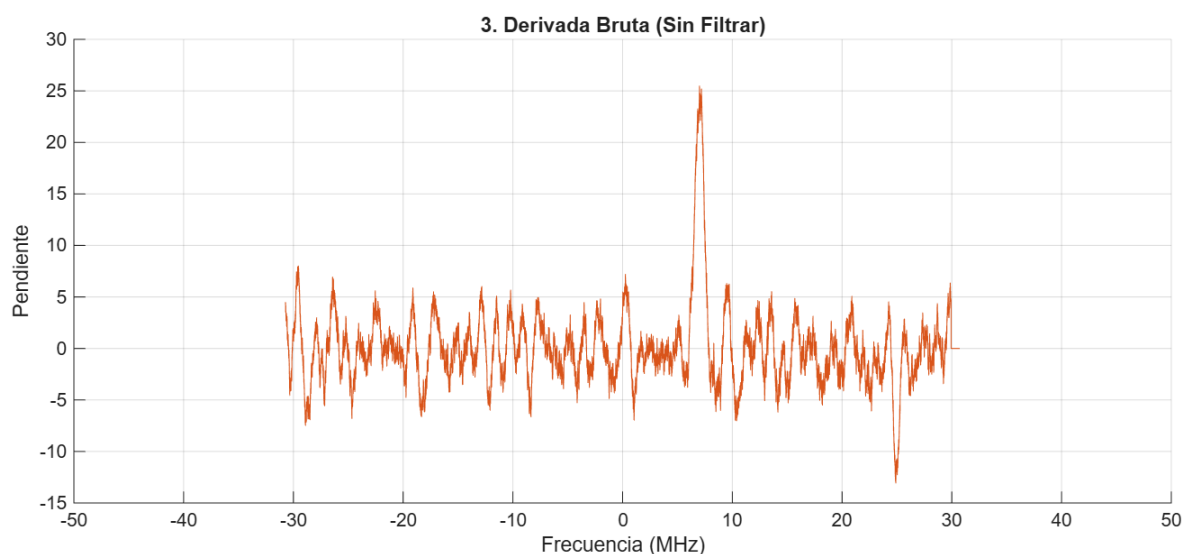


Figura 25: Derivada bruta del espectro suavizado, sin filtrar.

A pesar de que el espectro de magnitud (Figura 24) no permite distinguir la banda, al calcular la derivada del espectro ya suavizado (Figura 25) reaparece información reconocible: se observa un pico positivo claramente dominante en torno a 7 MHz (alcanzando un valor de pendiente de aproximadamente 25, frente a un fondo de ruido que apenas supera  $\pm 8$ ) y un valle negativo pronunciado en torno a 25 MHz (con un valor de aproximadamente -13). Esto demuestra el efecto fundamental del suavizado y la derivada: aunque el ruido eleva el nivel absoluto del espectro hasta enmascarar la banda, las transiciones de energía en sus bordes siguen produciendo una respuesta característica en la derivada, suficientemente distinguible del ruido de fondo de la propia derivada.

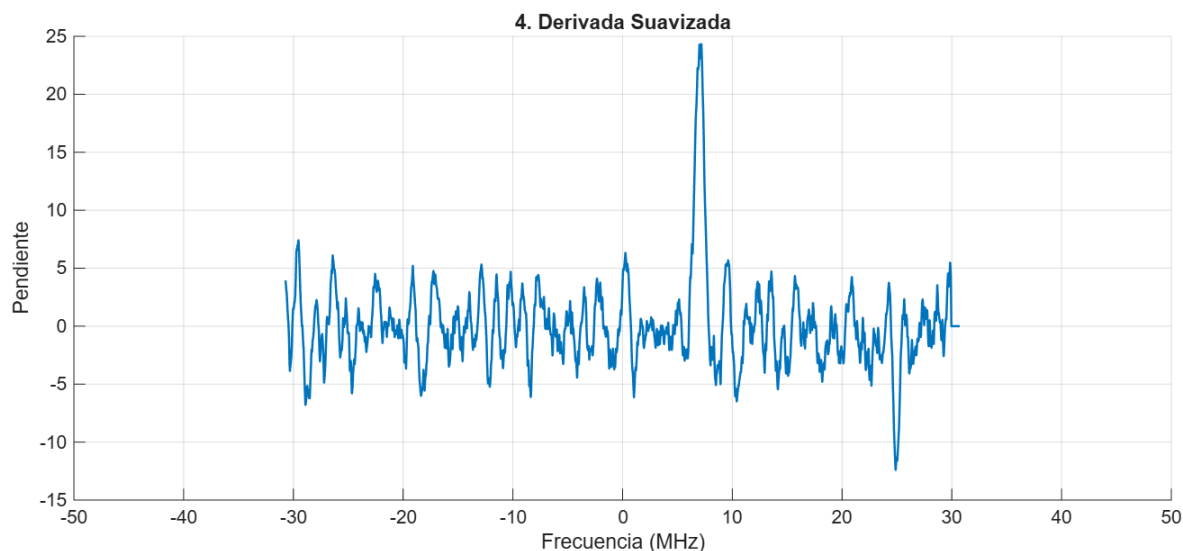


*Figura 26: Derivada suavizada mediante movmean.*

El segundo suavizado aplicado sobre la derivada (Figura 26) atenúa ligeramente las fluctuaciones más finas sin alterar significativamente los dos picos principales, que siguen siendo claramente identificables.

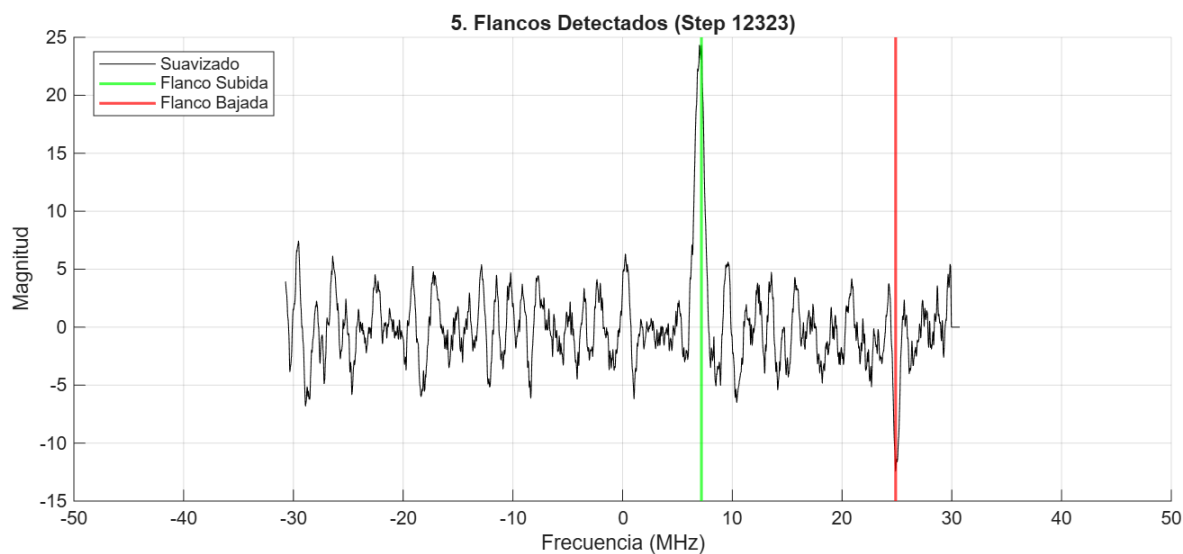


Figura 27: Bordes detectados para  $step = 12323$ .

La Figura 27 muestra el resultado de aplicar el criterio de umbral descrito en el apartado 7.3.3 sobre la derivada suavizada: el algoritmo identifica correctamente el flanco de subida (línea verde, en torno a 7,2 MHz) y el flanco de bajada (línea roja, en torno a 24,9 MHz), coincidiendo en ambos casos con los dos picos visibles en la derivada.

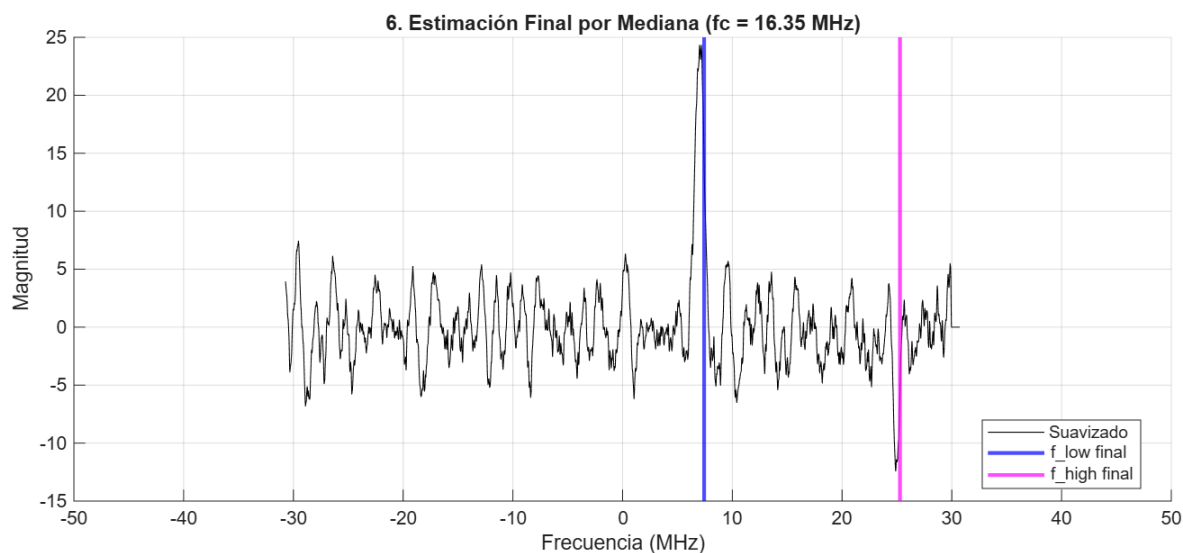


Figura 28: Estimación final por mediana ( $f_c = 16,35$  MHz).

Por último, la Figura 28 muestra la estimación final tras aplicar la mediana sobre los diez steps evaluados (línea azul para  $\hat{f}_l$ , línea magenta para  $\hat{f}_h$ ), con una frecuencia central resultante de 16,35 MHz, coherente con los valores de la Tabla 5.

Este recorrido visual ilustra de forma clara el principio de funcionamiento del algoritmo: aunque la señal original quede completamente enmascarada por el ruido en el espectro de magnitud, las transiciones de energía en los bordes de la banda dejan una huella suficientemente distinguible en la derivada del espectro suavizado, lo que permite al algoritmo recuperar una estimación razonablemente precisa de los límites de la banda incluso en condiciones de ruido considerablemente desfavorables.

## ***7.4 ALGORITMO 2: SUAVIZADO POR BLOQUES Y DETECCIÓN POR DIFERENCIAS***

El segundo algoritmo desarrollado aborda el mismo problema de estimación de los bordes de la banda que el Algoritmo 1, pero introduce dos diferencias de planteamiento relevantes. En primer lugar, en lugar de suavizar el espectro mediante una media móvil deslizante, este algoritmo lo divide en bloques contiguos no solapados, sustituyendo cada uno por su valor representativo. En segundo lugar, en lugar de calcular la derivada del espectro completo y aplicar un umbral configurable sobre ella, este algoritmo trabaja directamente sobre las diferencias entre bloques y localiza sus valores extremos (máximo y mínimo), sin necesidad de fijar un umbral de decisión. Como se verá en los apartados siguientes, ambos algoritmos comparten el mismo principio general (suavizar, calcular diferencias con step variable, detectar los extremos correspondientes a los bordes, y combinar los resultados de varios steps mediante mediana), pero difieren en los detalles concretos de implementación de cada una de estas etapas.

### 7.4.1 SUAVIZADO POR BLOQUES

La primera etapa del Algoritmo 2 también tiene como objetivo reducir la variabilidad bin a bin del espectro de magnitud antes de aplicar cualquier criterio de detección, pero lo hace mediante un procedimiento distinto al de la media móvil empleada en el Algoritmo 1. En lugar de calcular una media deslizante centrada en cada bin, el espectro se divide en bloques contiguos y no solapados de tamaño  $W$ , y todos los bins dentro de un mismo bloque se sustituyen por el valor medio de dicho bloque:

```
function [S_smooth] = suavizado_bloques(S, N, ventana)
    S_smooth = zeros(N, 1);
    for i = 1:ventana:N
        fin = min(i + ventana - 1, N);
        bloque = S(i:fin);
        media = mean(bloque);
        S_smooth(i:fin) = media;
    end
end
```

El bucle recorre el espectro en saltos de tamaño  $W$  (ventana), tomando en cada iteración un bloque de bins consecutivos  $S(i:fin)$ . El límite  $fin$  se calcula como  $\min(i + ventana - 1, N)$  para que, en caso de que  $N$  no sea múltiplo exacto de  $W$ , el último bloque se trunque correctamente en lugar de intentar acceder a posiciones fuera del vector. Para cada bloque se calcula su media aritmética, y ese único valor se asigna a todos los bins que componen el bloque ( $S\_smooth(i:fin) = media$ ), de modo que el resultado es constante dentro de cada bloque. Matemáticamente, para el bloque  $b$ -ésimo, que abarca los bins  $[bW, (b + 1)W - 1]$ :

$$\tilde{S}[k] = \frac{1}{W} \sum_{m=0}^{W-1} S[bW + m], k \in [bW, (b + 1)W - 1]$$

Esta forma de suavizado produce un resultado cualitativamente distinto al de la media móvil del Algoritmo 1. Mientras que movmean genera una curva continua que varía suavemente bin a bin (porque cada ventana se solapa parcialmente con la anterior y la siguiente), el suavizado por bloques genera una función escalonada: un valor constante dentro de cada bloque, con saltos discretos entre un bloque y el siguiente. Esta diferencia tiene implicaciones directas sobre la etapa de detección posterior, descrita en el apartado 7.4.2, ya que en lugar de trabajar sobre una derivada continua, el Algoritmo 2 trabaja sobre diferencias entre valores de bloques discretos.

El tamaño de bloque  $W$  se determina mediante el mismo mecanismo de doble configuración ya descrito en el apartado 7.3.1: bien como un valor fijo en número de muestras, bien como un porcentaje del total  $N$ . En las pruebas realizadas en este TFG se empleó la opción de valor fijo, con  $W = 12.323$  muestras.

## 7.4.2 DETECCIÓN POR DIFERENCIAS ENTRE BLOQUES

Una vez obtenido el espectro suavizado por bloques, el Algoritmo 2 no calcula una derivada continua como hacía el Algoritmo 1, sino que trabaja directamente sobre un valor representativo de cada bloque y construye diferencias entre bloques separados por una distancia configurable (step, esta vez expresado en número de bloques, no en bins).

### 7.4.2.1 Extracción del valor representativo de cada bloque

```
N = length(S_smooth);
num_bloques = ceil(N / ventana);

valores_bloques = zeros(num_bloques, 1);
frecuencias_bloques = zeros(num_bloques, 1);

for i = 1:num_bloques
    idx_centro = min(i * ventana, N);
    valores_bloques(i) = S_smooth(idx_centro);
    frecuencias_bloques(i) = f(idx_centro);
end
```

El número de bloques se calcula como  $M = \lceil N/W \rceil$ . Para cada bloque  $i$ , el código toma el índice  $\text{idx\_centro} = \min(i \cdot \text{ventana}, N)$ . Conviene señalar que, pese a su nombre, este índice no corresponde al centro del bloque, sino a su extremo derecho (el último bin del bloque, o  $N$  si el bloque final queda truncado). Esto no afecta al resultado en la práctica, ya que el espectro  $S_{\text{smooth}}$  es constante dentro de cada bloque (todos sus bins comparten el mismo valor medio, como se vio en el apartado 7.4.1), por lo que leer el valor en cualquier posición del bloque (incluido su extremo) da el mismo resultado que leerlo en su centro real. Sin embargo, la frecuencia asociada a ese bloque,  $\text{frecuencias\_bloques}(i)$ , sí queda ligeramente desplazada hacia el extremo derecho del bloque en lugar de su centro real, lo que introduce un pequeño sesgo de localización proporcional al tamaño de bloque  $W$ , normalmente despreciable frente al ancho de banda de la señal cuando  $W \ll N$ .

#### 7.4.2.2 Cálculo de las diferencias entre bloques

```
num_diferencias = num_bloques - step;
diferencias = zeros(num_diferencias, 1);
frecuencias_dif = zeros(num_diferencias, 1);

for i = 1:num_diferencias
    diferencias(i) = valores_bloques(i + step) - valores_bloques(i);

    if mod(step, 2) == 1
        idx_central = i + (step - 1)/2;
        frecuencias_dif(i) = frecuencias_bloques(idx_central);
    else
        idx_central1 = i + step/2 - 1;
        idx_central2 = i + step/2;
        frecuencias_dif(i) = (frecuencias_bloques(idx_central1) +
frecuencias_bloques(idx_central2)) / 2;
    end
end
```

Para cada posición  $i$ , se calcula la diferencia entre el valor del bloque  $i + \text{step}$  y el valor del bloque  $i$ , de forma análoga a la derivada del Algoritmo 1 pero contando en unidades de bloques en lugar de bins individuales:

$$\Delta_i = \bar{S}_{i+\text{step}} - \bar{S}_i, i = 1, \dots, M - \text{step}$$

La diferencia clave respecto al Algoritmo 1 está en cómo se asigna una frecuencia a cada diferencia calculada. En lugar de asociar la diferencia a la frecuencia del primer punto de la comparación (como hacía la derivada hacia adelante del Algoritmo 1, lo que introducía un desplazamiento creciente con el step), aquí se asigna la frecuencia del punto medio entre los dos bloques comparados. Si step es impar, existe un bloque exactamente a mitad de camino entre  $i$  e  $i + \text{step}$ , y se toma directamente su frecuencia. Si step es par, no existe un bloque central exacto, y se promedian las frecuencias de los dos bloques más próximos al centro:

$$f_{\Delta_i} = \begin{cases} \bar{f}_{i+(\text{step}-1)/2}, & \text{step impar} \\ \frac{\bar{f}_{i+\text{step}/2-1} + \bar{f}_{i+\text{step}/2}}{2}, & \text{step par} \end{cases}$$

Esta forma de centrar la frecuencia asignada es preferible, en términos de precisión de localización, a la asignación hacia adelante empleada en el Algoritmo 1: al situar la diferencia en el punto medio del intervalo de comparación, en lugar de en su extremo inicial, el sesgo de localización introducido por el uso de un step grande queda parcialmente compensado.

#### 7.4.2.3 Detección de los bordes

```
if modo_algoritmo == 1
    [~, idx_subida] = max(diferencias);
    [~, idx_min_rel] = min(diferencias(idx_subida:end));
    idx_bajada = idx_subida + idx_min_rel - 1;
else
    [~, idx_bajada] = min(diferencias);
    [~, idx_max_rel] = max(diferencias(1:idx_bajada));
    idx_subida = idx_max_rel;
end

f_low = frecuencias_dif(idx_subida);
f_high = frecuencias_dif(idx_bajada);
```

A diferencia del Algoritmo 1, que buscaba el primer cruce de un umbral configurable sobre la derivada, el Algoritmo 2 no emplea ningún umbral: busca directamente la posición del valor extremo (máximo o mínimo) del vector de diferencias. En el Modo 1, se localiza primero el máximo global de las diferencias (el borde de subida,  $\hat{f}_l$ ), y a continuación se busca el mínimo dentro de la región posterior a ese máximo (el borde de bajada,  $\hat{f}_h$ ), evitando así confundir el descenso inmediatamente posterior al propio pico de subida con la verdadera caída al final de la banda. En el Modo 2, el orden se invierte: se localiza primero el mínimo global (borde de bajada) y después el máximo dentro de la región anterior a él (borde de subida). Este planteamiento, al buscar directamente extremos globales acotados a una región en lugar de cruces de umbral, resulta más simple de implementar que el del Algoritmo 1, pero también lo hace, en principio, más sensible a la presencia de un único pico de ruido especialmente intenso que pudiera superar en magnitud a la transición real de la banda, ya que no existe ningún umbral mínimo de exigencia para considerar válida una detección.

Este procedimiento se repite, igual que en el Algoritmo 1, para cada valor de step del conjunto configurado, y la estimación final se obtiene como la mediana de los resultados, siguiendo el mecanismo común descrito en el apartado 7.2.3.

### 7.4.3 RESULTADOS EXPERIMENTALES

Para evaluar el Algoritmo 2 se empleó la misma señal de referencia que en el Algoritmo 1 (OcuSync 2, recibida por el ADALM-PLUTO, con valores de referencia  $f_l^{ref} = 7,5$  MHz y  $f_h^{ref} = 25,5$  MHz), con un tamaño de bloque fijo  $W = 12.323$  muestras y el conjunto de steps  $\{1,2,3,4,5\}$ , considerablemente más reducido que el empleado en el Algoritmo 1, ya que aquí el step se expresa en número de bloques y no en bins individuales: con  $N = 961.196$  muestras y  $W = 12.323$ , el espectro queda dividido en aproximadamente 78 bloques, cada uno de los cuales cubre un intervalo de frecuencia de en torno a 0,79 MHz, de modo que un step de solo 5 unidades ya representa una separación de casi 4 MHz entre los puntos comparados.

#### 7.4.3.1 Caso $SNR = -25$ dB

| Step                   | $\hat{f}_l$ (MHz) | $\hat{f}_h$ (MHz) | Err. $f_l$ /ref (%) | Err. $f_h$ /ref (%) | Err. $f_l$ / $f_s$ (%) | Err. $f_h$ / $f_s$ (%) |
|------------------------|-------------------|-------------------|---------------------|---------------------|------------------------|------------------------|
| 1                      | 16,542            | 18,905            | 120,55              | 25,86               | 14,72                  | 10,74                  |
| 2                      | 7,483             | 25,600            | 0,23                | 0,39                | 0,03                   | 0,16                   |
| 3                      | 17,329            | 25,994            | 131,06              | 1,94                | 16,00                  | 0,80                   |
| 4                      | 8,271             | 14,572            | 10,28               | 42,85               | 1,25                   | 17,79                  |
| 5                      | 8,665             | 26,782            | 15,53               | 5,03                | 1,90                   | 2,09                   |
| <b>Mediana (final)</b> | <b>8,665</b>      | <b>25,600</b>     | <b>15,53</b>        | <b>0,39</b>         | <b>1,90</b>            | <b>0,16</b>            |

Tabla 7: Resultados del Algoritmo 2 por step, para  $SNR = -25$  dB (Modo 1).

A este nivel de ruido tan severo, el comportamiento del algoritmo es desigual entre las dos frecuencias de corte. La estimación de  $f_h$  por mediana resulta sorprendentemente precisa, con un error relativo de apenas el 0,39% respecto a la referencia (un 0,16% del ancho de banda completo disponible), a pesar de que los resultados individuales por step son muy dispersos (entre 14,57 y 26,78 MHz). La estimación de  $f_l$ , en cambio, presenta un error relativo considerablemente mayor, del 15,53% respecto al valor de referencia (un 1,90% del ancho de banda completo), reflejo de que solo dos de los cinco steps (step=2, con un error del 0,23%, y step=4, con un error del 10,28%) se aproximan razonablemente al valor real, mientras que los otros tres (step=1 y step=3, con errores del 120,55% y el 131,06% respectivamente) sobrestiman notablemente la frecuencia de corte inferior. Con un conjunto de solo cinco steps, el margen de la mediana para "filtrar" valores erróneos es mucho menor que en el Algoritmo 1 (que disponía de diez), lo que explica esta mayor sensibilidad del resultado final ante un conjunto reducido de estimaciones discordantes.

#### 7.4.3.2 Caso SNR = -15 dB

| Step                   | $\hat{f}_l$ (MHz) | $\hat{f}_h$ (MHz) | Err. $f_l$ /ref (%) | Err. $f_h$ /ref (%) | Err. $f_l$ / $f_s$ (%) | Err. $f_h$ / $f_s$ (%) |
|------------------------|-------------------|-------------------|---------------------|---------------------|------------------------|------------------------|
| 1                      | 7,877             | 11,815            | 5,03                | 53,67               | 0,61                   | 22,27                  |
| 2                      | 7,483             | 25,600            | 0,23                | 0,39                | 0,03                   | 0,16                   |
| 3                      | 7,877             | 25,206            | 5,03                | 1,15                | 0,61                   | 0,48                   |
| 4                      | 8,271             | 24,812            | 10,28               | 2,70                | 1,25                   | 1,12                   |
| 5                      | 7,089             | 25,206            | 5,48                | 1,15                | 0,67                   | 0,48                   |
| <b>Mediana (final)</b> | <b>7,877</b>      | <b>25,206</b>     | <b>5,03</b>         | <b>1,15</b>         | <b>0,61</b>            | <b>0,48</b>            |

Tabla 8: Resultados del Algoritmo 2 por step, para SNR = -15 dB (Modo 1).

Con un nivel de ruido más moderado, el algoritmo mejora claramente su comportamiento: el error relativo de la estimación final es del 5,03% en  $f_l$  y del 1,15% en  $f_h$ . Se observa que cuatro de los cinco steps coinciden en un valor de  $f_h$  próximo a la referencia (entre 24,81 y 25,60 MHz), con la única excepción del step=1 (11,82 MHz), que la mediana descarta sin dificultad al quedar claramente aislado del resto. Para  $f_l$ , los cinco valores se mantienen relativamente agrupados entre 7,09 y 8,27 MHz, lo que explica el error moderado mantenido en la estimación final.

#### 7.4.3.3 Recorrido visual del algoritmo (SNR = -15 dB, step = 5, Modo 1)

Las Figuras Figura 29 a Figura 33 muestran las etapas internas del Algoritmo 2 para el caso de SNR = -15 dB y step = 5.

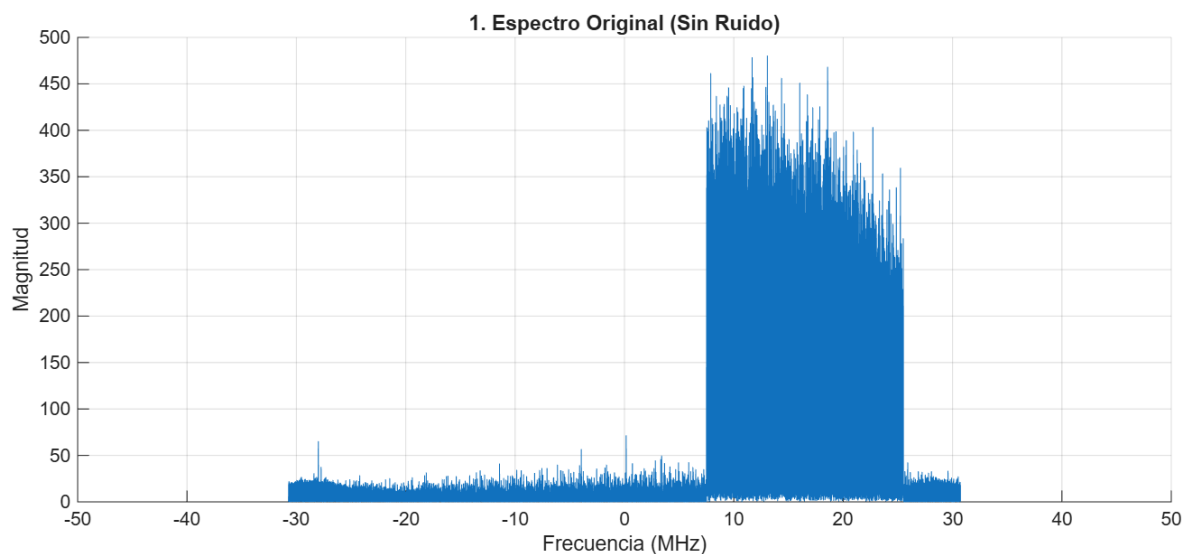


Figura 29: Espectro de magnitud original, sin ruido añadido.

Al igual que en el recorrido del Algoritmo 1, el espectro sin ruido (Figura 29) permite identificar a simple vista la banda ocupada, con un nivel de magnitud de hasta 480 dentro de la banda frente a un fondo residual inferior a 60.

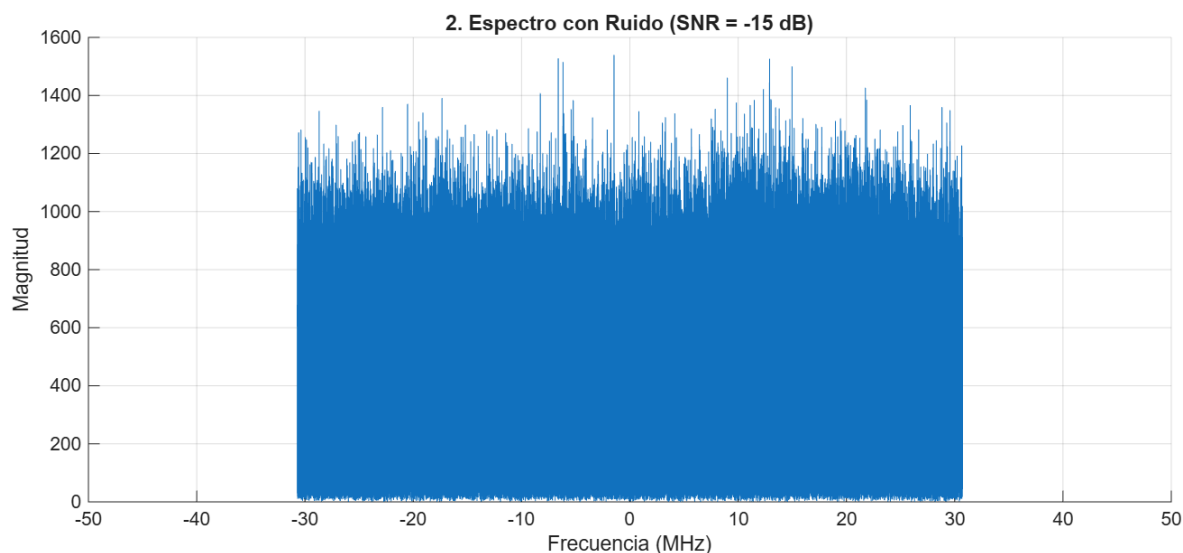


Figura 30: Espectro de magnitud tras añadir ruido AWGN con SNR = -15 dB.

Tras añadir ruido (Figura 30), el espectro queda dominado por valores que alcanzan picos de hasta 1.600, ocultando por completo la banda original a simple vista, exactamente igual que en el caso analizado para el Algoritmo 1.

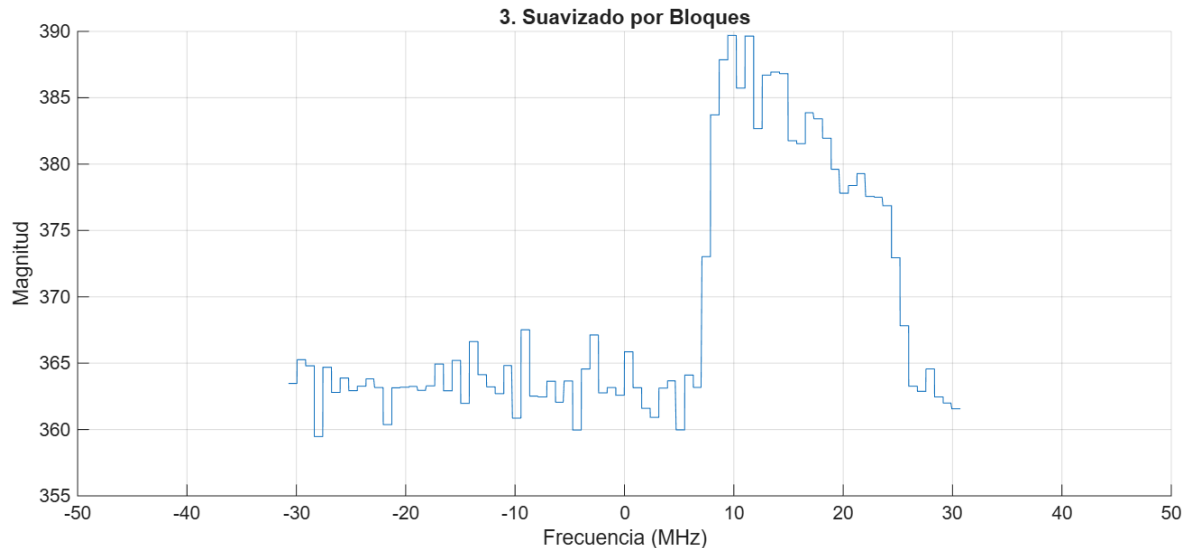


*Figura 31: Espectro suavizado mediante promediado por bloques.*

El efecto del suavizado por bloques (Figura 31) es notable: aunque el espectro ruidoso alcanzaba picos puntuales de hasta 1.600, el promedio dentro de cada bloque de 12.323 muestras reduce considerablemente esa magnitud, ya que dichos picos son excursiones puntuales del ruido y no representativas del nivel medio del bloque al que pertenecen. El resultado es una función escalonada que oscila entre aproximadamente 360 y 367 fuera de la banda, y entre 373 y 390 dentro de ella, recuperando así, en forma de un contraste de magnitud mucho más modesto pero claramente apreciable, la diferencia entre la región ocupada por la señal y el resto del espectro. Es precisamente esta diferencia, de en torno a 10-25 unidades, la que el algoritmo explota en la siguiente etapa para localizar los bordes de la banda, a pesar de que en el espectro ruidoso original esa diferencia resultaba completamente invisible.

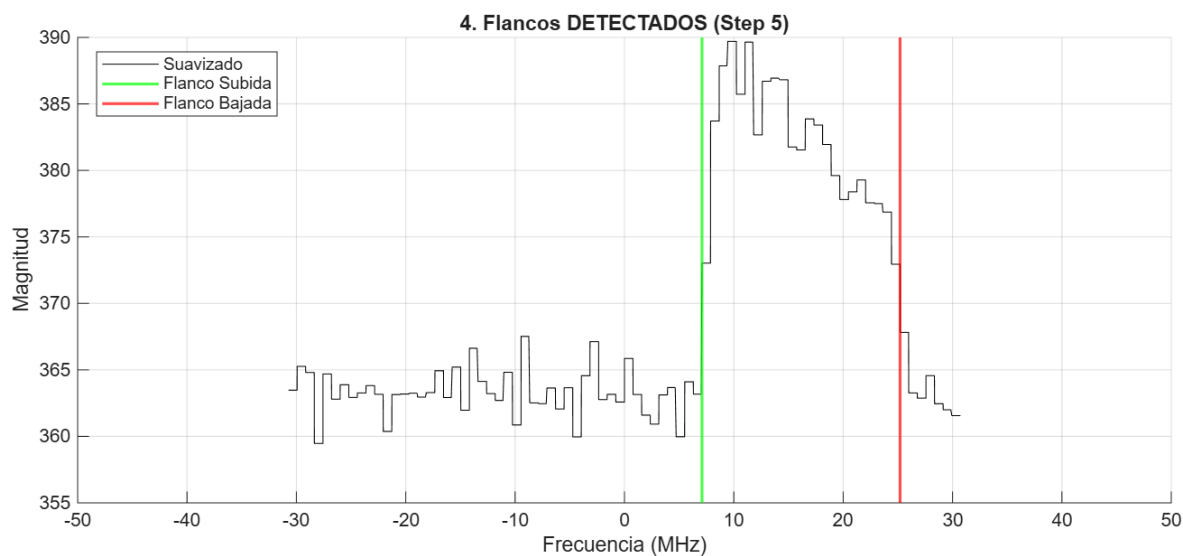


*Figura 32: Bordes detectados para step = 5.*

La Figura 32 muestra el resultado de buscar el máximo y el mínimo de las diferencias entre bloques separados por 5 posiciones: el flanco de subida (línea verde) se sitúa en torno a 7,1 MHz y el flanco de bajada (línea roja) en torno a 25,2 MHz, ambos coincidiendo con los valores recogidos para step=5 en la Tabla 8.

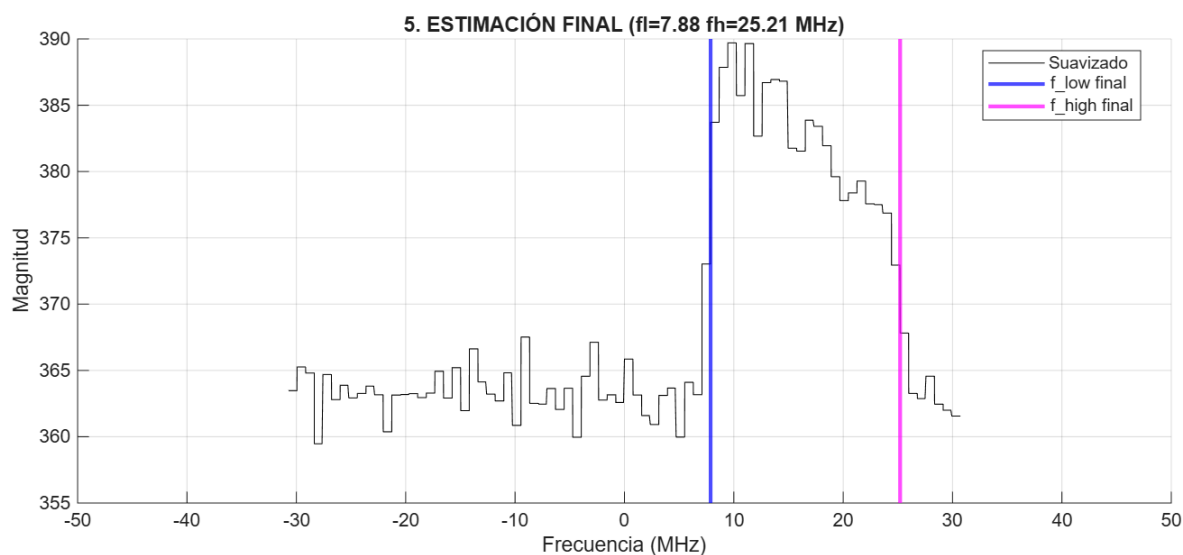


*Figura 33: Estimación final por mediana ( $\hat{f}_l = 7,88$  MHz,  $\hat{f}_h = 25,21$  MHz).*

Por último, la Figura 33 muestra la estimación final tras aplicar la mediana sobre los cinco steps evaluados, con valores de 7,88 MHz y 25,21 MHz, coherentes con la fila de mediana de la Tabla 8.

Este recorrido confirma que, al igual que ocurría con el Algoritmo 1, la clave del funcionamiento del Algoritmo 2 reside en que el suavizado (en este caso, por bloques) es capaz de revelar una diferencia de magnitud entre la banda ocupada y el resto del espectro que, en presencia de ruido, resulta imperceptible en el espectro original sin procesar.

## 7.5 COMPARACIÓN ENTRE ALGORITMOS

Antes de comparar los dos algoritmos, es importante señalar una limitación de las pruebas presentadas en los apartados 7.3.4 y 7.4.3: se trata de ejecuciones individuales a niveles de SNR concretos, y no de una caracterización estadística sistemática (por ejemplo, repitiendo cada prueba sobre múltiples realizaciones de ruido independientes, o barriendo de forma continua un rango de SNR y calculando el porcentaje de estimaciones válidas según el criterio de tolerancia  $\epsilon$  definido en el apartado 7.1). Por ello, las conclusiones de este apartado deben entenderse como observaciones cualitativas extraídas de los casos analizados, y no como una comparación estadísticamente exhaustiva entre ambos algoritmos.

| <i>Algoritmo</i>                    | $\hat{f}_l(\text{MHz})$ | <i>Err.</i><br>(%) | $f_l/\text{ref}$ | <i>Err.</i><br>(%) | $f_l/f_s$ | $\hat{f}_h(\text{MHz})$ | <i>Err.</i><br>(%) | $f_h/\text{ref}$ | <i>Err.</i><br>(%) | $f_h/f_s$ |
|-------------------------------------|-------------------------|--------------------|------------------|--------------------|-----------|-------------------------|--------------------|------------------|--------------------|-----------|
| Algoritmo 1 (movmean + derivada)    | 7,430                   | 0,94               |                  | 0,11               |           | 25,276                  | 0,88               |                  | 0,36               |           |
| Algoritmo 2 (bloques + diferencias) | 7,877                   | 5,03               |                  | 0,61               |           | 25,206                  | 1,15               |                  | 0,48               |           |

Tabla 9: Comparación de resultados a SNR = -15 dB, único punto evaluado en común entre ambos algoritmos.

En el único punto de SNR evaluado en condiciones directamente comparables (-15 dB), el Algoritmo 1 ofrece un resultado claramente más preciso que el Algoritmo 2, especialmente en la estimación de  $f_l$  (error del 0,94% frente al 5,03% respecto al valor de referencia; 0,11% frente al 0,61% respecto al ancho de banda completo disponible). Esta diferencia puede atribuirse, al menos en parte, a tres diferencias de diseño entre ambos algoritmos:

En primer lugar, el número de steps evaluados: el Algoritmo 1 emplea diez valores de step, mientras que el Algoritmo 2 emplea solo cinco. Como se discutió en los apartados 7.3.4 y 7.4.3, el mecanismo de mediana depende de disponer de suficientes estimaciones individuales para que los valores correctos puedan "imponerse" sobre los erróneos al ordenar el conjunto; con solo cinco valores, cualquier estimación desviada tiene un peso proporcionalmente mayor sobre el resultado final que con diez.

En segundo lugar, la granularidad del step: en el Algoritmo 1, el step se mide en bins individuales, lo que permite evaluar separaciones muy finas (desde 1 bin) hasta separaciones muy amplias (12.323 bins) dentro de un mismo barrido. En el Algoritmo 2, el step se mide en bloques de 12.323 muestras cada uno, de modo que incluso el step más pequeño (1) ya representa una separación de casi 0,79 MHz; el algoritmo carece, por tanto, de la capacidad de evaluar separaciones muy finas que sí están disponibles para el Algoritmo 1.

En tercer lugar, el criterio de detección: el Algoritmo 1 emplea un umbral configurable sobre la derivada suavizada, lo que permite ajustar la exigencia del criterio de detección y ofrece un mecanismo de respaldo cuando ningún punto supera dicho umbral. El Algoritmo 2 busca directamente el extremo global de las diferencias entre bloques, sin ningún umbral de por medio, lo que simplifica la implementación pero elimina ese margen de ajuste.

Frente a estas diferencias, el Algoritmo 2 presenta una ventaja relevante en su etapa de suavizado: al promediar bloques completos de muestras en lugar de aplicar una ventana deslizante, consigue una reducción de la magnitud del ruido considerablemente más drástica (como se observó en la Figura 31, donde picos de hasta 1.600 en el espectro ruidoso se reducían a un rango de apenas 360-390 tras el suavizado por bloques), lo que en principio podría hacerlo más adecuado en escenarios de ruido extremo si se combinara con un mayor número de steps o con bloques de menor tamaño que permitieran mayor granularidad sin perder esta capacidad de atenuación.

En definitiva, los resultados disponibles sugieren que el Algoritmo 1 ofrece, en las condiciones evaluadas, una estimación más precisa, gracias fundamentalmente a la mayor cantidad de información combinada por el mecanismo de mediana y a su mayor granularidad de step. No obstante, una comparación más concluyente requeriría extender las pruebas a un barrido sistemático de SNR común a ambos algoritmos, evaluando el porcentaje de estimaciones válidas según el criterio del apartado 7.1 sobre un número suficientemente elevado de realizaciones de ruido independientes, lo que se propone como línea natural de trabajo futuro derivada directamente de este capítulo.

## **Capítulo 8. RESULTADOS EXPERIMENTALES Y**

### **ANÁLISIS DE RESULTADOS**

Este capítulo recoge los resultados de la campaña experimental completa diseñada para evaluar el comportamiento de los dos algoritmos de detección desarrollados en el capítulo 7 (Capítulo 7. ), aplicados sobre el conjunto íntegro de señales del proyecto: las tres capturas reales y las tres capturas reales acotadas de cada uno de los cinco drones, junto con su correspondiente señal simulada, todas ellas recibidas a través del ADALM-PLUTO según la metodología descrita en el capítulo 6 (Capítulo 6. ).

Para cada combinación de dron, tipo de señal, algoritmo y modo de detección, se evaluaron diez niveles de relación señal-ruido,  $SNR \in \{10, 5, 0, -5, -10, -15, -20, -25, -30, -35\}$  dB, repitiendo cada prueba 100 veces de forma independiente (cada repetición con una realización distinta del ruido AWGN añadido) para obtener una caracterización estadísticamente representativa del comportamiento de cada algoritmo, en lugar de depender de ejecuciones puntuales como las presentadas con fines ilustrativos en el capítulo anterior. Teniendo en cuenta que el Algoritmo 1 evalúa once valores de step y el Algoritmo 2 evalúa seis, el número total de estimaciones individuales generadas a lo largo de esta campaña asciende a 170.000, repartidas entre las 110.000 correspondientes al Algoritmo 1 y las 60.000 correspondientes al Algoritmo 2.

Para determinar si cada una de estas estimaciones se considera válida, se emplea el criterio general definido en el apartado 7.1, concretado aquí con un margen de tolerancia  $\epsilon$  del 1% sobre el ancho de banda total disponible ( $f_s = 61,44$  MHz), de forma que una estimación  $\hat{f}$  se considera válida si  $|\hat{f} - f^{ref}| \leq 0,01 \cdot f_s = 0,6144$  MHz. Este margen, al ser fijo y común a las cinco señales y a ambas frecuencias de corte, permite comparar de forma homogénea el comportamiento del sistema entre drones cuyos valores de referencia difieren notablemente entre sí, evitando el efecto de escala que se observó en el capítulo 7 (Capítulo 7.) al expresar el error como porcentaje del propio valor teórico.

El volumen de resultados generado por esta campaña hace inviable, evidentemente, presentar de forma exhaustiva los 170.000 resultados individuales en la memoria. Resulta, por tanto, especialmente importante seleccionar con criterio qué resultados se presentan y cómo se agregan, de forma que la elección sea representativa del comportamiento real del sistema y no transmita una impresión sesgada de su rendimiento, ya sea de forma deliberada o, lo que resulta más delicado de evitar, de forma inconsciente: por ejemplo, eligiendo casos favorables sin un criterio objetivo previo, o presentando únicamente las configuraciones donde el sistema funciona mejor. Para mitigar este riesgo, los criterios de selección de los resultados presentados en los apartados siguientes (qué dron, qué tipo de señal, qué nivel de SNR) se fijan de antemano y se explicitan junto a cada resultado, de manera que el lector pueda valorar tanto los casos favorables como los desfavorables, y entender el motivo concreto por el que cada uno se ha incluido en la memoria.

## 8.1 CRITERIO DE SELECCIÓN DE LOS RESULTADOS PRESENTADOS

Aunque el número total de estimaciones individuales generadas por la campaña experimental asciende a 170.000, el análisis y la selección de resultados no se realizan sobre estimaciones individuales, sino sobre el porcentaje de acierto agregado: para cada combinación de dron (o dron y muestra concreta, en el caso de las señales reales y reales acotadas), algoritmo, modo, step (o mediana) y nivel de SNR, se calcula el porcentaje de las 100 iteraciones que resultan válidas al compararse con los valores de referencia de la Tabla 3. Esta agregación reduce el conjunto a 1.700 puntos (porcentaje de acierto por nivel de SNR), agrupados en 170 curvas distintas, cada una formada por los 10 puntos de SNR evaluados para una combinación concreta de dron/muestra, algoritmo, modo y step.

Buscar directamente el mejor y el peor resultado entre el conjunto completo de estas 170 curvas resultaría, sin embargo, poco informativo: por la propia naturaleza de los datos, el mejor resultado global recaería casi con total seguridad sobre alguna señal simulada (al carecer de interferencias e imperfecciones propias de una captura real), y el peor sobre alguna señal real sin acotar. Para extraer información más rica del conjunto de datos, la búsqueda de mejor y peor resultado se realiza, por tanto, de forma independiente dentro de cada uno de los tres tipos de señal del proyecto (simulada, real acotada y real), y dentro de cada tipo de señal, de forma independiente para cada una de las cuatro combinaciones de algoritmo y modo (Algoritmo 1-Modo 1, Algoritmo 1-Modo 2, Algoritmo 2-Modo 1, Algoritmo 2-Modo 2). Dentro de cada una de estas cuatro categorías, se seleccionan las dos curvas con mayor puntuación (para las tablas de mejor resultado) o las dos con menor puntuación (para las tablas de peor resultado), según la puntuación ponderada:

$$\text{Puntuación} = 0,25 \cdot P(-35) + 0,25 \cdot P(-30) + 0,30 \cdot P(-25) + 0,15 \cdot P(-20) + 0,10 \cdot P(-15)$$

Esto da lugar a seis tablas y figuras (mejor y peor resultado para cada uno de los tres tipos de señal), cada una compuesta por ocho curvas (cuatro combinaciones algoritmo-modo, con dos instancias cada una).

## 8.2 RESULTADOS MÁS FAVORABLES

### 8.2.1 SEÑALES SIMULADAS

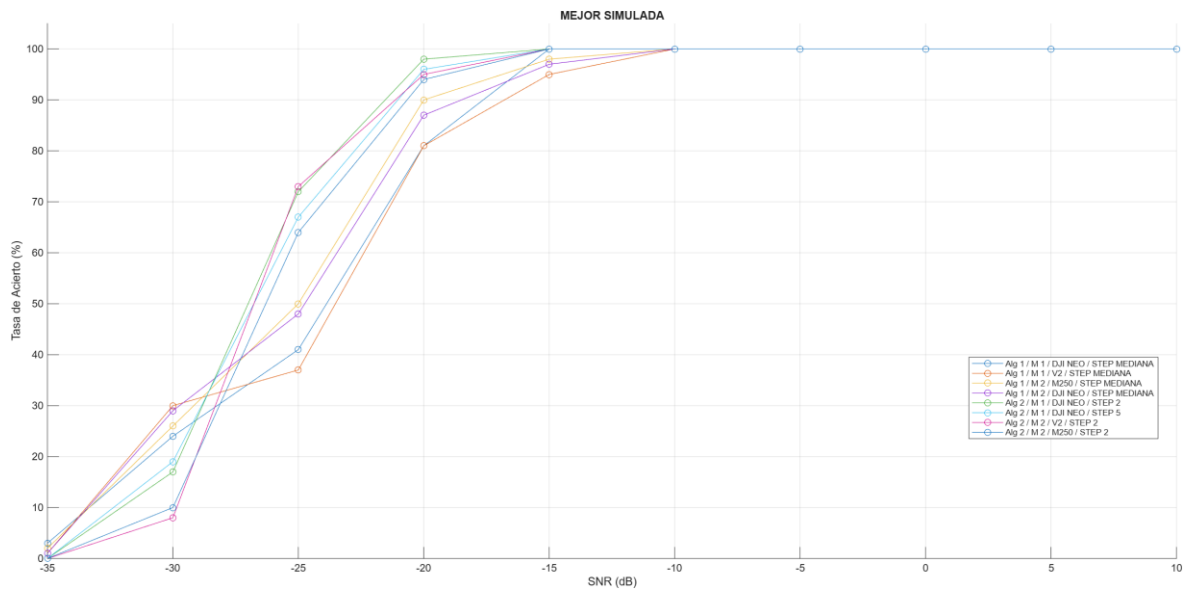


Figura 34: Tasa de detecciones válidas frente a SNR para las configuraciones con mejor rendimiento sobre señales simuladas."

| Algoritmo | Modo | Dron    | Step    | 10  | 5   | 0   | -5  | -10 | -15 | -20 | -25 | -30 | -35 |
|-----------|------|---------|---------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| 1         | 1    | DJI Neo | Mediana | 100 | 100 | 100 | 100 | 100 | 100 | 81  | 41  | 24  | 3   |
| 1         | 1    | V2      | Mediana | 100 | 100 | 100 | 100 | 100 | 95  | 81  | 37  | 30  | 1   |
| 1         | 2    | M250    | Mediana | 100 | 100 | 100 | 100 | 100 | 98  | 90  | 50  | 26  | 2   |
| 1         | 2    | DJI Neo | Mediana | 100 | 100 | 100 | 100 | 100 | 97  | 87  | 48  | 29  | 1   |
| 2         | 1    | DJI Neo | 2       | 100 | 100 | 100 | 100 | 100 | 100 | 98  | 72  | 17  | 0   |
| 2         | 1    | DJI Neo | 5       | 100 | 100 | 100 | 100 | 100 | 100 | 96  | 67  | 19  | 0   |
| 2         | 2    | V2      | 2       | 100 | 100 | 100 | 100 | 100 | 100 | 95  | 73  | 8   | 0   |
| 2         | 2    | M250    | 2       | 100 | 100 | 100 | 100 | 100 | 100 | 94  | 64  | 10  | 0   |

Tabla 10: Mejor rendimiento sobre señales simuladas, por algoritmo, modo y dron.

Las ocho configuraciones analizadas presentan un comportamiento muy robusto frente al ruido. De hecho, todas ellas mantienen una tasa de detección del 100 % hasta una SNR de -10 dB, un resultado especialmente relevante si se considera que, en este régimen, la potencia de la señal ya es inferior a la del ruido.

Las diferencias entre algoritmos comienzan a manifestarse a partir de -15 dB. Mientras que el Algoritmo 1 experimenta una ligera degradación, el Algoritmo 2 mantiene todavía una detección perfecta. Esta ventaja se acentúa al reducir aún más la SNR y resulta especialmente significativa a -25 dB, donde el Algoritmo 2 conserva tasas de acierto superiores al 60 %, frente a valores inferiores al 50 % en el Algoritmo 1. En conjunto, estos resultados indican una mayor robustez del Algoritmo 2 en condiciones de ruido severo.

Este comportamiento es coherente con el análisis realizado en el apartado 7.5. El suavizado por bloques empleado por el Algoritmo 2 produce una atenuación del ruido considerablemente más intensa que la media móvil utilizada por el Algoritmo 1. Como se mostró en la Figura 31, picos de ruido de hasta 1.600 unidades se reducen a valores del orden de 30 unidades, proporcionando un mayor margen para la identificación de los bordes espectrales cuando la SNR es muy baja.

Por el contrario, el modo de detección empleado tiene una influencia limitada en el rendimiento global. Aunque el Modo 2 ofrece una ligera mejora respecto al Modo 1 en algunas configuraciones del Algoritmo 1, las diferencias son reducidas y, en el caso del Algoritmo 2, ambos modos presentan un comportamiento prácticamente idéntico. Esto sugiere que, para las señales simuladas analizadas, la elección del algoritmo tiene un impacto mucho mayor que la selección del modo de detección.

Otro aspecto destacable es la presencia recurrente de la señal DJI Neo, que aparece en seis de las ocho mejores configuraciones obtenidas. Este resultado puede relacionarse con sus características espectrales, ya que presenta un ancho de banda relativamente elevado y una distribución simétrica respecto a la frecuencia central. Aunque ninguno de los algoritmos asume explícitamente esta simetría, los patrones espectrales más regulares parecen favorecer una localización más consistente de los bordes de la banda ocupada.

En conjunto, estos resultados representan el escenario más favorable de toda la campaña experimental. Incluso para valores de SNR inferiores a -20 dB, donde la banda ocupada resulta prácticamente indistinguible a simple vista en el espectro de magnitud, el sistema mantiene una tasa de detección mayoritaria, lo que pone de manifiesto la elevada robustez de la metodología propuesta.

## 8.2.2 SEÑALES REALES ACOTADAS

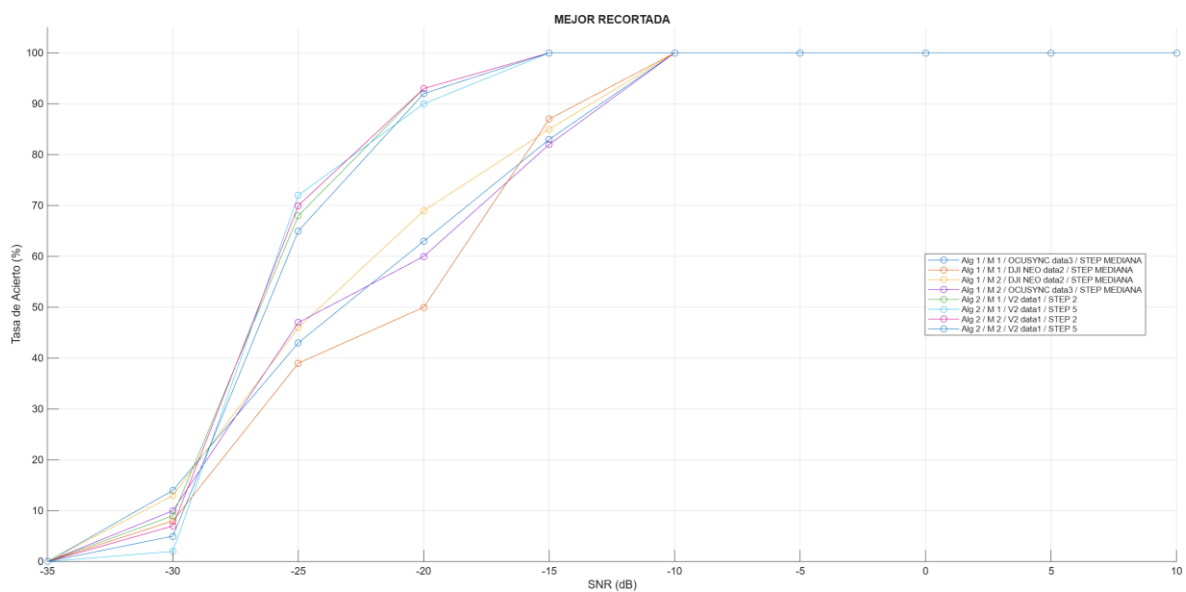


Figura 35: Tasa de detecciones válidas frente a SNR para las configuraciones con mejor rendimiento sobre señales reales acotadas.

| Algoritmo | Modo | Dron      | Step | 10  | 5   | 0   | -5 | -10 | -15 | -20 | -25 | -30 | -35 |
|-----------|------|-----------|------|-----|-----|-----|----|-----|-----|-----|-----|-----|-----|
| 1         | 1    | OcuSync 2 | 500  | 100 | 100 | 93  | 47 | 32  | 19  | 1   | 0   | 0   | 0   |
| 1         | 1    | OcuSync 2 | 7500 | 100 | 100 | 90  | 54 | 29  | 14  | 0   | 0   | 0   | 0   |
| 1         | 2    | OcuSync 2 | 2500 | 100 | 100 | 90  | 32 | 20  | 10  | 0   | 0   | 0   | 0   |
| 1         | 2    | OcuSync 2 | 500  | 100 | 100 | 83  | 20 | 25  | 9   | 0   | 0   | 0   | 0   |
| 2         | 1    | V2        | 1    | 100 | 100 | 100 | 63 | 41  | 19  | 0   | 0   | 0   | 0   |
| 2         | 1    | V2        | 3    | 100 | 100 | 100 | 70 | 36  | 13  | 1   | 0   | 0   | 0   |
| 2         | 2    | V2        | 1    | 100 | 100 | 100 | 73 | 44  | 9   | 0   | 0   | 0   | 0   |
| 2         | 2    | V2        | 3    | 100 | 100 | 100 | 62 | 22  | 5   | 0   | 0   | 0   | 0   |

Tabla 11: Mejor rendimiento sobre señales reales acotadas, por algoritmo, modo, dron y muestra.

Al igual que en el caso de las señales simuladas, las ocho configuraciones mantienen una tasa de acierto del 100 % hasta una SNR de -10 dB. A partir de este punto, el patrón de degradación reproduce de forma muy similar el comportamiento previamente observado. El Algoritmo 1 comienza a perder detecciones a -15 dB, mientras que el Algoritmo 2 mantiene todavía una detección perfecta. Esta diferencia se hace aún más evidente a -20 dB, donde el Algoritmo 2 alcanza tasas de acierto próximas al 90 %, frente a valores claramente inferiores obtenidos por el Algoritmo 1.

La elevada concordancia entre los resultados obtenidos con señales simuladas y con señales reales acotadas constituye una de las conclusiones más relevantes de este capítulo. En particular, pone de manifiesto que el procedimiento de acotación espectral descrito en el apartado 5.5 consigue aproximar de forma efectiva el comportamiento de una captura real al de una señal sintética perfectamente caracterizada, al menos para las configuraciones de mejor rendimiento de cada algoritmo.

No obstante, este comportamiento cambia en el escenario de ruido más severo. A una SNR de -30 dB, ambos algoritmos presentan tasas de detección muy reducidas y, en algunos casos, el Algoritmo 1 llega incluso a superar ligeramente al Algoritmo 2. Esta inversión puntual sugiere que la ventaja observada para el Algoritmo 2 en el intervalo de -15 a -25 dB no se mantiene indefinidamente cuando el nivel de ruido es extremadamente elevado, tendiendo ambos algoritmos hacia un comportamiento igualmente degradado.

En cuanto al tipo de señal empleada, se observa una diferencia significativa respecto al caso simulado. Las mejores configuraciones del Algoritmo 1 se reparten entre las señales OcuSync 2 y DJI Neo, mientras que las del Algoritmo 2 corresponden exclusivamente a la señal V2. Este resultado indica que, en las señales reales acotadas, el rendimiento de cada algoritmo depende en mayor medida de las particularidades de cada captura concreta y no únicamente de las características teóricas de la señal. Por último, el modo de detección vuelve a desempeñar un papel secundario, ya que ambos modos presentan resultados muy similares y ninguno de ellos ofrece una mejora sistemática sobre el otro.

### 8.2.3 SEÑALES REALES

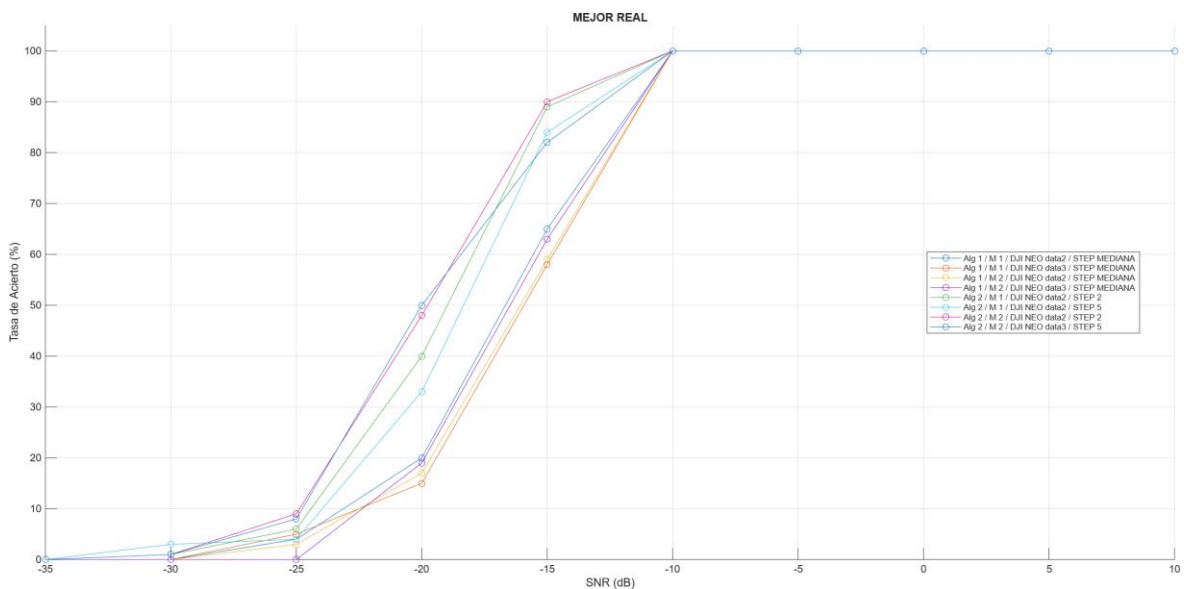


Figura 36: Tasa de detecciones válidas frente a SNR para las configuraciones con mejor rendimiento sobre señales reales sin acotar.

| Algoritmo | Modo | Dron (muestra) | Step | 10 | 5  | 0  | -5 | -10 | -15 | -20 | -25 | -30 | -35 |
|-----------|------|----------------|------|----|----|----|----|-----|-----|-----|-----|-----|-----|
| 1         | 1    | Mini 2 (data3) | 500  | 59 | 30 | 28 | 14 | 8   | 0   | 0   | 0   | 0   | 0   |
| 1         | 1    | Mini 2 (data3) | 100  | 62 | 35 | 37 | 10 | 5   | 0   | 0   | 0   | 0   | 0   |
| 1         | 2    | Mini 2 (data3) | 7500 | 45 | 51 | 25 | 5  | 0   | 0   | 0   | 0   | 0   | 0   |
| 1         | 2    | Mini 2 (data3) | 1000 | 60 | 55 | 27 | 7  | 0   | 0   | 0   | 0   | 0   | 0   |
| 2         | 1    | Mini 2 (data3) | 2    | 63 | 53 | 42 | 15 | 3   | 0   | 0   | 0   | 0   | 0   |
| 2         | 1    | Mini 2 (data3) | 3    | 53 | 40 | 35 | 10 | 0   | 0   | 0   | 0   | 0   | 0   |
| 2         | 2    | Mini 2 (data3) | 2    | 57 | 39 | 32 | 9  | 0   | 0   | 0   | 0   | 0   | 0   |
| 2         | 2    | Mini 2 (data3) | 3    | 64 | 51 | 39 | 14 | 0   | 0   | 0   | 0   | 0   | 0   |

Tabla 12: Mejor rendimiento sobre señales reales sin acotar, por algoritmo, modo, dron y muestra.

La Tabla 12 presenta la diferencia de rendimiento más acusada entre el Algoritmo 1 y el Algoritmo 2 de toda la campaña experimental. La superioridad del Algoritmo 2 se hace especialmente evidente a partir de -15 dB, donde supera al Algoritmo 1 en aproximadamente 20-30 puntos porcentuales, y se mantiene también a -20 dB, llegando incluso a duplicar su tasa de acierto en algunas configuraciones. Aunque esta diferencia se reduce para niveles de ruido más severos, el Algoritmo 2 continúa mostrando un comportamiento más robusto en la mayor parte del rango de SNR analizado.

Estos resultados constituyen la evidencia más sólida de las conclusiones extraídas en el apartado 7.5. Cuando la señal de entrada conserva no solo el ruido AWGN añadido, sino también las imperfecciones propias de una captura real sin procesar, como picos espurios, tonos de referencia o componentes residuales fuera de banda, el suavizado por bloques empleado por el Algoritmo 2 proporciona un beneficio significativamente mayor que en las señales simuladas o espectralmente acotadas. Su mecanismo de promediado resulta especialmente eficaz para atenuar estas irregularidades adicionales, permitiendo una localización más robusta de los bordes espectrales en condiciones adversas.

Por otra parte, las ocho mejores configuraciones corresponden, sin excepción, a las capturas de DJI Neo (muestras data2 y data3). Este resultado refuerza la conclusión ya apuntada en el apartado 8.3.1. Gracias a su ancho de banda relativamente elevado y a la simetría de su espectro respecto a la frecuencia central, DJI Neo constituye la señal más favorable para la detección en las tres variantes analizadas del proyecto, simulada, real acotada y real sin procesar, representando el caso de mayor robustez entre las cinco señales estudiadas.

## 8.3 RESULTADOS MÁS DESFAVORABLES

### 8.3.1 SEÑALES SIMULADAS

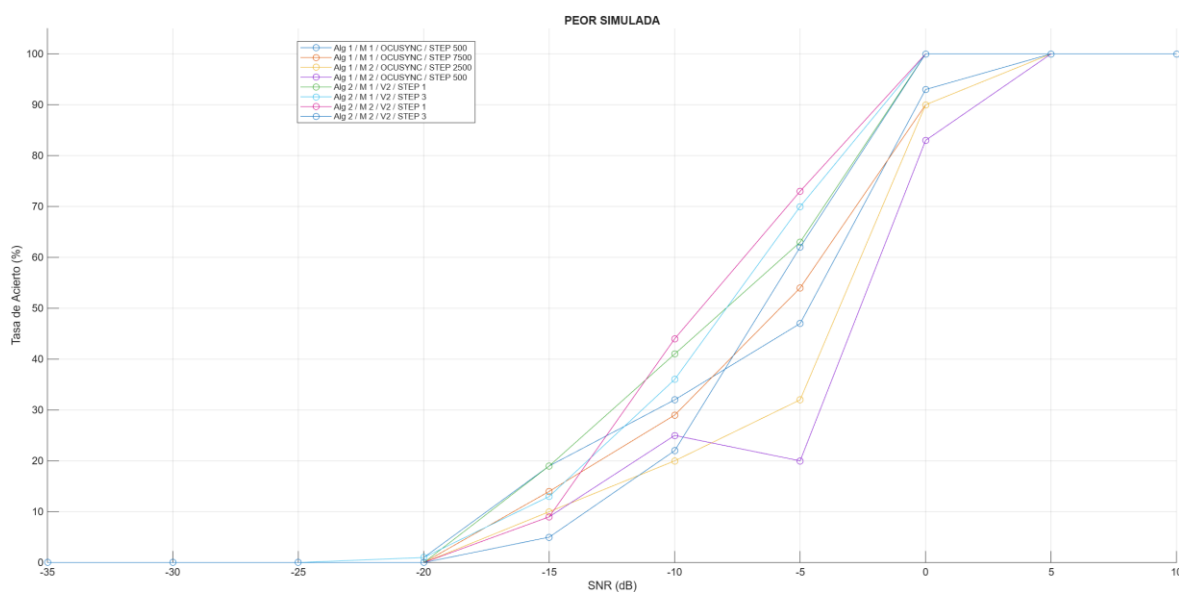


Figura 37: Tasa de detecciones válidas frente a SNR para las configuraciones con peor rendimiento sobre señales simuladas.

| Algoritmo | Modo | Dron      | Step | 10  | 5   | 0   | -5 | -10 | -15 | -20 | -25 | -30 | -35 |
|-----------|------|-----------|------|-----|-----|-----|----|-----|-----|-----|-----|-----|-----|
| 1         | 1    | OcuSync 2 | 500  | 100 | 100 | 93  | 47 | 32  | 19  | 1   | 0   | 0   | 0   |
| 1         | 1    | OcuSync 2 | 7500 | 100 | 100 | 90  | 54 | 29  | 14  | 0   | 0   | 0   | 0   |
| 1         | 2    | OcuSync 2 | 2500 | 100 | 100 | 90  | 32 | 20  | 10  | 0   | 0   | 0   | 0   |
| 1         | 2    | OcuSync 2 | 500  | 100 | 100 | 83  | 20 | 25  | 9   | 0   | 0   | 0   | 0   |
| 2         | 1    | V2        | 1    | 100 | 100 | 100 | 63 | 41  | 19  | 0   | 0   | 0   | 0   |
| 2         | 1    | V2        | 3    | 100 | 100 | 100 | 70 | 36  | 13  | 1   | 0   | 0   | 0   |
| 2         | 2    | V2        | 1    | 100 | 100 | 100 | 73 | 44  | 9   | 0   | 0   | 0   | 0   |
| 2         | 2    | V2        | 3    | 100 | 100 | 100 | 62 | 22  | 5   | 0   | 0   | 0   | 0   |

Tabla 13: Peor rendimiento sobre señales simuladas, por algoritmo, modo, dron y step.

El primer aspecto destacable de esta tabla es que, a pesar de representar el peor resultado dentro de las señales simuladas, las ocho configuraciones mantienen una tasa de acierto del 100 % hasta 5 dB. Además, el Algoritmo 2 conserva este comportamiento incluso a 0 dB, mientras que el Algoritmo 1 comienza a mostrar una ligera degradación en este punto.

La superioridad del Algoritmo 2 se mantiene a -5 dB, donde alcanza tasas de detección claramente superiores a las del Algoritmo 1. Sin embargo, esta ventaja se reduce progresivamente al disminuir la SNR y prácticamente desaparece a -15 dB, donde ambos algoritmos presentan porcentajes de acierto muy similares. A partir de -20 dB, el rendimiento de ambos métodos se degrada hasta niveles que pueden considerarse no funcionales.

Otro resultado de interés es que el dron asociado al peor caso difiere entre algoritmos. Mientras que las configuraciones más desfavorables del Algoritmo 1 corresponden a la señal OcuSync 2, en el Algoritmo 2 el peor comportamiento se obtiene con la señal V2. Este resultado contrasta con el patrón observado en las mejores configuraciones, dominadas de forma consistente por DJI Neo, y sugiere que la dificultad de una señal no es una propiedad intrínseca de la propia señal, sino que depende de la interacción entre sus características espectrales y el mecanismo de detección empleado.

Una posible explicación, aunque no ha sido verificada de forma exhaustiva en este trabajo, es que la asignación de frecuencia hacia adelante empleada por el Algoritmo 1 penalice en mayor medida las bandas desplazadas, como la de OcuSync 2, mientras que la granularidad más gruesa asociada a los steps del Algoritmo 2 resulte menos adecuada para señales de ancho de banda reducido, como la señal V2.

En cualquier caso, la conclusión más relevante de esta tabla es que incluso el peor escenario simulado mantiene un funcionamiento satisfactorio en un amplio margen de SNR, sin experimentar una degradación abrupta hasta valores claramente negativos de la relación señal a ruido.

### 8.3.2 SEÑALES REALES ACOTADAS

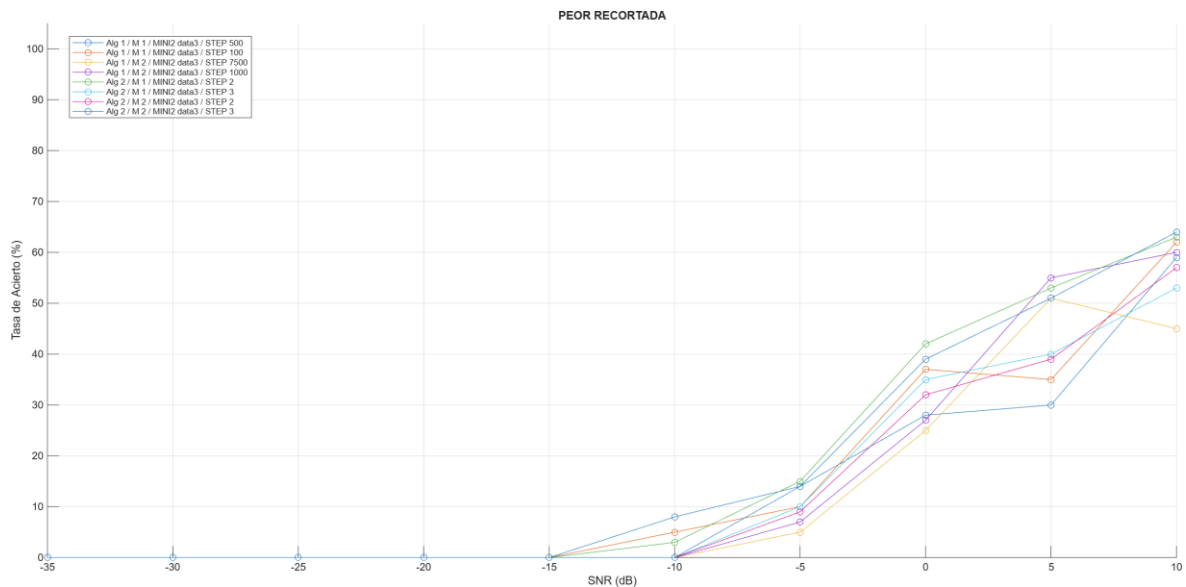


Figura 38: Tasa de detecciones válidas frente a SNR para las configuraciones con peor rendimiento sobre señales reales acotadas.

| Algoritmo | Modo | Dron (muestra) | Step | 10 | 5  | 0  | -5 | -10 | -15 | -20 | -25 | -30 | -35 |
|-----------|------|----------------|------|----|----|----|----|-----|-----|-----|-----|-----|-----|
| 1         | 1    | Mini 2 (data3) | 500  | 59 | 30 | 28 | 14 | 8   | 0   | 0   | 0   | 0   | 0   |
| 1         | 1    | Mini 2 (data3) | 100  | 62 | 35 | 37 | 10 | 5   | 0   | 0   | 0   | 0   | 0   |
| 1         | 2    | Mini 2 (data3) | 7500 | 45 | 51 | 25 | 5  | 0   | 0   | 0   | 0   | 0   | 0   |
| 1         | 2    | Mini 2 (data3) | 1000 | 60 | 55 | 27 | 7  | 0   | 0   | 0   | 0   | 0   | 0   |
| 2         | 1    | Mini 2 (data3) | 2    | 63 | 53 | 42 | 15 | 3   | 0   | 0   | 0   | 0   | 0   |
| 2         | 1    | Mini 2 (data3) | 3    | 53 | 40 | 35 | 10 | 0   | 0   | 0   | 0   | 0   | 0   |
| 2         | 2    | Mini 2 (data3) | 2    | 57 | 39 | 32 | 9  | 0   | 0   | 0   | 0   | 0   | 0   |
| 2         | 2    | Mini 2 (data3) | 3    | 64 | 51 | 39 | 14 | 0   | 0   | 0   | 0   | 0   | 0   |

Tabla 14: Peor rendimiento sobre señales reales acotadas, por algoritmo, modo, dron, muestra y step.

Esta es la primera tabla del capítulo en la que resulta necesario moderar el tono de éxito generalizado observado hasta el momento. Las ocho configuraciones corresponden, sin excepción, a la misma captura, Mini 2 (muestra data3), y ninguna de ellas supera el 64 % de acierto ni siquiera a una SNR de 10 dB, un nivel de ruido que, en principio, puede considerarse favorable.

Este comportamiento puede relacionarse con las características de la propia señal. Como se indicó en el apartado 5.2.2, Mini 2 presenta el menor ancho de banda de las cinco señales analizadas, aproximadamente 9,43 MHz, frente a los 18-36 MHz del resto. En consecuencia, el margen disponible para identificar correctamente los dos bordes espectrales es significativamente menor, especialmente cuando la señal incorpora además las imperfecciones presentes en la captura data3.

A diferencia de lo observado en las tablas anteriores, la superioridad del Algoritmo 2 sobre el Algoritmo 1 deja de ser evidente. Ambos algoritmos presentan rendimientos muy similares en niveles moderados de ruido y, en el escenario más severo, el Algoritmo 1 llega incluso a obtener resultados ligeramente superiores. Aunque ambos métodos operan ya muy cerca de su límite de funcionamiento, este comportamiento resulta coherente con la hipótesis planteada en el apartado anterior. Ante una banda especialmente estrecha, la mayor granularidad de step disponible en el Algoritmo 1, que permite evaluar separaciones más finas, podría proporcionar una ligera ventaja frente a los steps más gruesos del Algoritmo 2, con mayor riesgo de no localizar correctamente los bordes reales de la señal.

Este resultado no debe interpretarse como un fallo del sistema, sino como la identificación de uno de sus límites de aplicabilidad práctica. Las señales con anchos de banda significativamente más estrechos que los analizados de forma predominante en este trabajo constituyen un reto adicional que merecería un estudio específico en futuras investigaciones, posiblemente mediante una mayor resolución espectral o la utilización de un step mínimo más reducido que el empleado en este TFG.

### 8.3.3 SEÑALES REALES

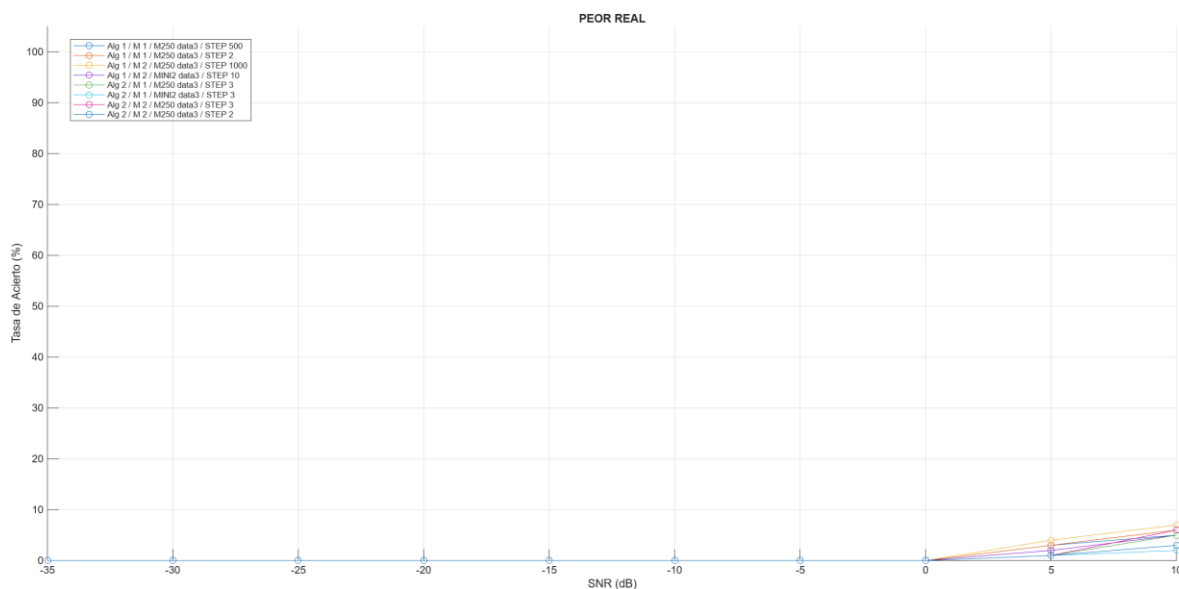


*Figura 39: Tasa de detecciones válidas frente a SNR para las configuraciones con peor rendimiento sobre señales reales sin acotar.*

| Algoritmo | Modo | Dron (muestra) | Step | 10 | 5 | 0 | -5 | -10 | -15 | -20 | -25 | -30 | -35 |
|-----------|------|----------------|------|----|---|---|----|-----|-----|-----|-----|-----|-----|
| 1         | 1    | M250 (data3)   | 500  | 5  | 3 | 0 | 0  | 0   | 0   | 0   | 0   | 0   | 0   |
| 1         | 1    | M250 (data3)   | 2    | 6  | 3 | 0 | 0  | 0   | 0   | 0   | 0   | 0   | 0   |
| 1         | 2    | M250 (data3)   | 1000 | 7  | 4 | 0 | 0  | 0   | 0   | 0   | 0   | 0   | 0   |
| 1         | 2    | Mini 2 (data3) | 10   | 5  | 2 | 0 | 0  | 0   | 0   | 0   | 0   | 0   | 0   |
| 2         | 1    | M250 (data3)   | 3    | 5  | 1 | 0 | 0  | 0   | 0   | 0   | 0   | 0   | 0   |
| 2         | 1    | Mini 2 (data3) | 3    | 2  | 1 | 0 | 0  | 0   | 0   | 0   | 0   | 0   | 0   |
| 2         | 2    | M250 (data3)   | 3    | 6  | 1 | 0 | 0  | 0   | 0   | 0   | 0   | 0   | 0   |
| 2         | 2    | M250 (data3)   | 2    | 3  | 1 | 0 | 0  | 0   | 0   | 0   | 0   | 0   | 0   |

*Tabla 15: Peor rendimiento sobre señales reales sin acotar, por algoritmo, modo, dron, muestra y step.*

Esta tabla (Tabla 15) representa el escenario de peor rendimiento de toda la campaña experimental. Ninguna de las ocho configuraciones supera el 7 % de validez ni siquiera en el caso más favorable analizado, correspondiente a una SNR de 10 dB, y el sistema deja de proporcionar detecciones válidas a partir de 0 dB. Además, no se observa una ventaja sistemática de un algoritmo sobre el otro, ya que ambos presentan porcentajes de acierto muy reducidos y similares. Este comportamiento constituye, en sí mismo, un resultado relevante, ya que sugiere que la limitación observada no está asociada al diseño de un algoritmo concreto, sino a las características de la propia señal de entrada.

Esta interpretación es coherente con lo descrito en el apartado 5.2.2. La captura real de M250 presenta, ya en su forma original, un nivel de magnitud y una relación señal-ruido considerablemente inferiores a los del resto de señales analizadas. La transmisión de esta señal mediante el ADALM-PLUTO, incorporando las pérdidas y el ruido inherentes a la cadena de RF descrita en el capítulo 6 (Capítulo 6. ), y la posterior adición de ruido AWGN en esta campaña experimental, dan lugar a una señal con una relación señal-ruido efectiva demasiado reducida para que cualquiera de los dos algoritmos pueda localizar de forma fiable los bordes de la banda ocupada.

Por tanto, este resultado pone de manifiesto uno de los límites prácticos de aplicabilidad del sistema desarrollado. Su rendimiento depende de forma directa de la calidad de la captura de partida y, cuando la señal original presenta una relación señal-ruido muy degradada, la capacidad de detección se ve severamente comprometida con independencia del algoritmo empleado. No obstante, este resultado también pone de relieve la importancia de las etapas de preprocesado de la señal. Como se observó en el apartado 8.2.2, la aplicación del procedimiento de acotación espectral sobre las capturas reales produce mejoras sustanciales en el rendimiento del sistema, lo que sugiere que una parte importante de la dificultad asociada a este peor caso puede mitigarse mediante un preprocesado adecuado antes de la etapa de detección.

## **8.4 SÍNTESIS DEL CAPÍTULO**

Los resultados presentados en este capítulo proceden de una campaña de más de 170.000 estimaciones individuales, en la que se han evaluado las tres categorías de señal del proyecto, los dos algoritmos desarrollados y diez niveles de SNR. En conjunto, los resultados muestran una robustez del sistema superior a la inicialmente prevista. Tanto para las señales simuladas como para las señales reales acotadas, las mejores configuraciones mantienen un 100 % de detecciones válidas hasta una SNR de -10 dB y conservan tasas de acierto mayoritarias incluso en torno a -20 dB. Este resultado es especialmente relevante si se considera que, como se mostró en el capítulo 7 (Capítulo 7. ), a estos niveles de degradación la banda ocupada por la señal resulta prácticamente indistinguible a simple vista en el espectro de magnitud.

Además, estos resultados no corresponden a casos seleccionados de antemano, sino al mejor comportamiento obtenido dentro de un conjunto de 170 configuraciones evaluadas de forma sistemática y bajo un único criterio de valoración. Esto refuerza la validez de las conclusiones obtenidas sobre el funcionamiento del sistema.

Incluso al analizar las configuraciones más desfavorables de cada categoría de señal, el sistema muestra, en general, un comportamiento más robusto de lo esperado. Las peores configuraciones simuladas y reales acotadas siguen siendo funcionales en un amplio rango de SNR y se degradan de forma gradual, en lugar de fallar de manera abrupta. La principal excepción es la captura real sin acotar de M250, cuyo rendimiento es reducido incluso para valores de SNR favorables. Sin embargo, los resultados indican que esta limitación está asociada principalmente a la baja relación señal-ruido de la captura original y no a una deficiencia de los algoritmos de detección. De hecho, la aplicación del procedimiento de acotación espectral del capítulo 5 (Capítulo 5. ) mejora de forma significativa el rendimiento de este tipo de señales.

En conjunto, la campaña experimental confirma que el sistema desarrollado en este TFG es capaz de estimar con precisión los límites de banda de las señales de dron analizadas. Esta capacidad se ha verificado a lo largo de toda la cadena de procesamiento propuesta, desde la generación y caracterización de las señales hasta la transmisión y recepción mediante el ADALM-PLUTO y la aplicación de los algoritmos de detección espectral. Además, la precisión alcanzada se mantiene en condiciones de ruido más severas de las inicialmente previstas. Las implicaciones de estos resultados y las posibles líneas de mejora derivadas de las limitaciones identificadas, como la sensibilidad a anchos de banda estrechos observada en Mini 2 y la dependencia de la calidad de la captura de partida observada en M250, se abordan con mayor detalle en el capítulo 9 (Capítulo 9. ).

## Capítulo 9. CONCLUSIONES Y TRABAJOS FUTUROS

Este capítulo cierra el cuerpo principal de la memoria. Recoge en qué medida se han cumplido los objetivos planteados al inicio del proyecto, sintetiza las conclusiones técnicas más relevantes obtenidas a lo largo de los distintos capítulos, expone con honestidad las limitaciones identificadas durante el desarrollo y propone líneas concretas de trabajo futuro derivadas de ellas.

### 9.1 CUMPLIMIENTO DE LOS OBJETIVOS DEL PROYECTO

Al inicio de este TFG se plantearon cinco objetivos concretos. A continuación se revisa cada uno de ellos, indicando en qué capítulo de la memoria queda demostrado su cumplimiento.

El primer objetivo era analizar señales reales de distintos drones. Este objetivo se ha cumplido en el capítulo 5 (Capítulo 5. ), donde se han caracterizado espectralmente las cinco señales de referencia del proyecto (M250, V2, OcuSync 2, Mini 2 y DJI Neo), a partir de las capturas reales facilitadas por el director del trabajo. De este análisis se obtuvieron los parámetros de banda recogidos en la Tabla 3 ( $f_l$ ,  $f_h$ ,  $f_c$ , ancho de banda y amplitud característica), que han servido como referencia para todo el trabajo posterior.

El segundo objetivo era generar y estudiar señales OFDM simuladas. Este objetivo también se ha cumplido en el capítulo 5 (Capítulo 5. ), donde se ha desarrollado un procedimiento de síntesis basado en ruido gaussiano acotado en frecuencia, justificado teóricamente por el comportamiento estadístico de las señales OFDM con un número elevado de subportadoras. Este mismo procedimiento se ha reutilizado de forma consistente a lo largo del proyecto, tanto para generar las cinco señales simuladas del conjunto de datos como para generar la señal de prueba empleada en la puesta a punto de la plataforma SDR en el capítulo 6 (Capítulo 6. ).

El tercer objetivo era desarrollar algoritmos de detección espectral y estimación de ancho de banda. Este es, probablemente, el objetivo que ha recibido mayor atención en la memoria. En el capítulo 7 (Capítulo 7. ) se han diseñado, justificado e implementado dos algoritmos completos, con fundamentos distintos (suavizado por media móvil y derivada en el primero, suavizado por bloques y diferencias en el segundo), y se ha evaluado su funcionamiento de forma detallada. La campaña experimental del capítulo 8 (Capítulo 8. ) confirma que ambos algoritmos son capaces de estimar con precisión los límites de banda de las señales evaluadas, incluso en condiciones de ruido considerablemente severas.

El cuarto objetivo era comparar señales simuladas y reales. Esta comparación se ha abordado desde dos ángulos complementarios. En el capítulo 5 (Capítulo 5. ) se ha comparado el espectro de las señales simuladas con el de las señales reales y reales acotadas, confirmando la coherencia de la unificación de las tres categorías en una estructura común. En el capítulo 8 (Capítulo 8. ), esta comparación se ha llevado al terreno experimental, evaluando el rendimiento de los algoritmos de detección sobre cada una de las tres categorías de señal, lo que ha permitido cuantificar, entre otras cosas, el efecto positivo del procedimiento de acotación espectral sobre la detectabilidad de una señal real.

El quinto y último objetivo era validar experimentalmente los resultados mediante transmisiones y recepciones con el ADALM-PLUTO. Este objetivo se ha cumplido en el capítulo 6 (Capítulo 6. ), donde se ha diseñado e implementado la plataforma de transmisión y recepción, documentando además el proceso de prueba y error seguido hasta alcanzar una configuración física funcional. Todos los resultados del capítulo 8 (Capítulo 8. ) proceden de señales recibidas a través de esta plataforma, lo que confirma que la validación experimental no se ha quedado en un plano puramente simulado, sino que se ha verificado con hardware real.

En conjunto, puede afirmarse que los cinco objetivos planteados al inicio del proyecto se han cumplido de forma satisfactoria.

## 9.2 EVOLUCIÓN DEL DESARROLLO: DE LA VERSIÓN PRELIMINAR A LOS RESULTADOS FINALES

Resulta ilustrativo comparar los resultados finales del capítulo 8 (Capítulo 8. ) con una versión preliminar del sistema de detección, evaluada durante una presentación de seguimiento intermedia del proyecto. En aquella fase, el algoritmo de detección correspondía a una versión todavía poco desarrollada de lo que acabaría siendo el Algoritmo 2 (suavizado por bloques y diferencias), evaluada únicamente para un par de valores de step (step=2 y step=3) y sin el conjunto completo de elementos descritos en el capítulo 7 (Capítulo 7. ). Además, en aquella evaluación preliminar, el ruido AWGN se añadía directamente sobre la señal simulada en banda base, sin que la señal pasara por ningún proceso de transmisión y recepción real.

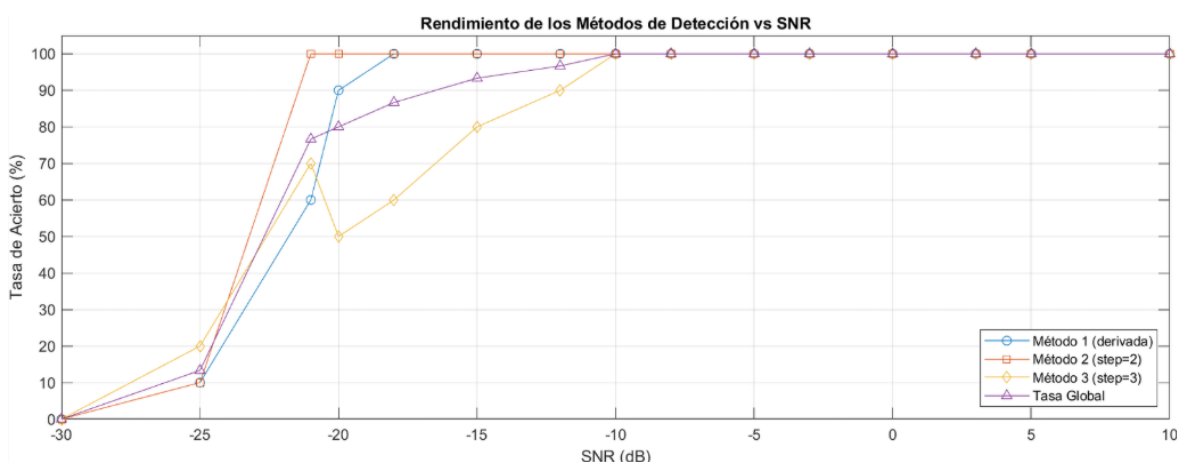


*Figura 40: Resultados de la versión preliminar del algoritmo de detección, evaluados durante el seguimiento intermedio del proyecto, con ruido añadido directamente sobre la señal simulada en banda base.*

Aquella versión preliminar mostraba un comportamiento razonable hasta aproximadamente -18 o -20 dB, pero se degradaba con rapidez por debajo de ese punto, acercándose a tasas de acierto casi nulas en torno a -25 y -30 dB. Los resultados finales del Algoritmo 2 presentados en el Capítulo 8 mejoran este comportamiento de forma clara: las mejores configuraciones mantienen tasas de acierto elevadas hasta -20 dB y conservan todavía entre un 64% y un 73% de validez a -25 dB, un nivel de SNR en el que la versión preliminar apenas ofrecía resultados utilizables.

Esta mejora es, además, doble. Por un lado, refleja el propio avance del algoritmo a lo largo del desarrollo del TFG, con la incorporación de elementos clave que no estaban presentes en la versión preliminar: el conjunto completo de steps evaluados, el mecanismo de robustez por mediana descrito en el apartado 7.2.3, y los dos modos de detección del apartado 7.2.4. Por otro lado, y de forma especialmente relevante, los resultados finales se han obtenido sobre señales que han pasado por la cadena de transmisión y recepción real del ADALM-PLUTO, con todas las pérdidas, el ruido de hardware y las imperfecciones propias de un sistema de radiofrecuencia físico, descritas en el capítulo 6 (Capítulo 6. ). Es decir, la versión final del sistema no solo mejora en términos algorítmicos, sino que lo hace evaluándose sobre un escenario de prueba considerablemente más exigente y realista que el empleado en la fase preliminar. Esta doble mejora, en el algoritmo y en el rigor del escenario de validación, respalda que el avance observado entre ambas fases del proyecto es genuino y no atribuible a una simplificación de las condiciones de prueba.

### **9.3 CONCLUSIONES TÉCNICAS PRINCIPALES**

Más allá de los números del capítulo 8 (Capítulo 8. ), el proyecto deja varias conclusiones técnicas que conviene resumir.

Sobre la generación de señales, el método empleado (ruido gaussiano acotado en frecuencia) ha resultado suficiente. No hace falta justificarlo solo con la teoría del apartado 5.3. Los propios resultados lo confirman: la señal simulada y la señal real acotada se comportan de forma muy parecida ante los algoritmos de detección. Esto demuestra que el método capta bien lo esencial de una señal real, sin necesidad de simular una cadena OFDM completa.

Sobre la plataforma SDR, el capítulo 6 (Capítulo 6. ) deja una lección clara. La configuración más sencilla (antenas conectadas directamente a un único Pluto) fue la única que funcionó bien. Las alternativas más ambiciosas, como usar dos Pluto o cables de extensión, fallaron por motivos distintos: problemas de sincronización de reloj en un caso, y un mal contacto del conector en el otro. Esto muestra que, en un proyecto con hardware real, las limitaciones prácticas pesan tanto como las decisiones de diseño. También se vio que la mejor frecuencia portadora para las pruebas no fue la banda de 2,4 GHz de los drones, sino 868 MHz, por tener menos pérdidas y menos interferencias de otros dispositivos como el Wi-Fi.

Sobre los algoritmos de detección, ambos funcionan bien, pero con puntos fuertes distintos. El Algoritmo 1 es más preciso con ruido moderado, gracias a su mayor número de steps y a su umbral configurable. El Algoritmo 2 destaca con ruido severo y con señales reales sin acotar, gracias a su suavizado por bloques, más agresivo. El mecanismo de la mediana, común a ambos, queda validado: tanto en los ejemplos del capítulo 7 (Capítulo 7. ) como en el análisis del capítulo 8 (Capítulo 8. ), la mediana suele dar de los mejores resultados posibles.

Por último, un patrón que se repite en los capítulos 5 (Capítulo 5. ) y 8 (Capítulo 8. ) es que las características de cada señal influyen tanto como el algoritmo usado. DJI Neo, con una banda ancha y simétrica, fue la señal más fácil de detectar en sus tres versiones. Mini 2, con la banda más estrecha, fue la más difícil. De cara al futuro, esto indica que conviene tener en cuenta no solo el nivel de ruido, sino también la forma del espectro de cada señal.

## **9.4 LIMITACIONES DEL TRABAJO**

El proyecto también tiene limitaciones, que conviene reconocer con claridad.

La primera tiene que ver con la calidad de algunas capturas reales. La señal de M250 partía ya de una relación señal-ruido baja, antes incluso de añadir ruido para las pruebas. Esto hizo que, en su forma sin acotar, el sistema apenas pudiera detectarla, como se vio en el apartado 8.3.3. No es un fallo de los algoritmos, sino una limitación de los datos de partida.

La segunda tiene que ver con el ancho de banda. La señal de Mini 2, al ser la más estrecha del conjunto, resultó más difícil de detectar que el resto. Esto sugiere que el sistema, tal y como está ahora, funciona mejor con señales de banda ancha que con señales más estrechas.

La tercera limitación es de tipo experimental. No fue posible probar el sistema con las antenas separadas a mayor distancia. El intento con dos Pluto falló por problemas de sincronización de reloj, y el intento con cables de extensión falló por un mal contacto del conector. Por tanto, todos los resultados del Capítulo 8 corresponden a una única distancia fija entre antenas, y no se sabe con certeza cómo se comportaría el sistema a distancias mayores.

La cuarta limitación afecta a la acotación espectral del apartado 5.5. Los límites de amplitud usados para limpiar cada señal real se fijaron a mano, mirando cada espectro. No se siguió un criterio automático o cuantitativo. Esto funciona bien para este proyecto, pero no sería directamente escalable si se quisiera aplicar a muchas más señales.

Por último, el sistema desarrollado estima los bordes de una banda ya conocida, pero no decide si esa banda corresponde realmente a un dron, ni de qué tipo. Es decir, el trabajo aborda el problema de la estimación de ancho de banda, pero no el de la detección de presencia ni el de la clasificación del dron, que quedarían como un paso posterior y diferenciado.

## **9.5 LÍNEAS DE TRABAJO FUTURO**

De las limitaciones anteriores, y de algunas ideas que han surgido durante el desarrollo del proyecto, se derivan varias líneas de trabajo futuro.

En relación con las señales reales, el reto no es solo el ancho de banda estrecho de señales como Mini 2. También influyen los armónicos espurios que aparecen en muchas capturas reales, ajenos a la propia banda de la señal. Estos armónicos son, en parte, los responsables de que los algoritmos resulten menos eficientes sobre señales reales que sobre señales simuladas. Sería interesante, por tanto, trabajar en hacer los algoritmos más robustos frente a este tipo de componentes espurios, no solo frente al ruido AWGN añadido de forma controlada.

En relación con la distancia entre antenas, sería interesante repetir las pruebas con cables SMA adecuados, certificados para el conector del ADALM-PLUTO, y comprobar con resultados válidos cómo afecta realmente la distancia al rendimiento del sistema. Esto es algo que no se pudo completar en este TFG, al fallar los cables empleados.

En esta misma línea, el uso de dos unidades de SDR independientes seguía siendo, en el fondo, un enfoque muy interesante, ya que es el que más se aproxima a una situación real de despliegue, con el transmisor y el receptor en ubicaciones distintas. El motivo de su fallo en este proyecto fue la falta de sincronización de reloj entre los dos ADALM-PLUTO. Otros modelos de SDR, menos limitados por este tipo de restricciones de software o con soporte para una referencia de reloj externa, podrían hacer viable este enfoque, que sería el más realista de todos los probados en este trabajo.

En relación con los algoritmos, el Algoritmo 2 cuenta con menos steps que el Algoritmo 1, lo que hace que su mediana sea un mecanismo de robustez menos eficaz, como se vio en el Capítulo 7. En vez de simplemente añadir más steps, sería más interesante explorar otra forma de combinar los resultados de los distintos steps, distinta de la mediana, que aproveche mejor la información disponible incluso con un número reducido de steps.

Por último, la línea de trabajo futuro más ambiciosa sería ir más allá de la estimación de ancho de banda y construir un verdadero sistema predictivo. Mediante técnicas de aprendizaje automático, se podría entrenar un modelo capaz de identificar, con un porcentaje de confianza, a qué dron concreto corresponde una señal detectada. Junto con esto, sería muy útil desarrollar una interfaz donde el usuario pudiera generar una señal con distintas características, elegir qué algoritmo de detección usar, con cuántos steps, qué nivel de ruido añadir, y configurar el sistema completo de principio a fin. Esa misma interfaz podría mostrar, además de la estimación de ancho de banda, el resultado del modelo de aprendizaje automático sobre a qué dron pertenece la señal, integrando en una sola herramienta todo lo desarrollado en este TFG junto con esta nueva capacidad de clasificación.

## **9.6 CONCLUSIÓN FINAL**

Este TFG ha recorrido el camino completo de un sistema de detección de señales de drones mediante radio definida por software: desde la caracterización de señales reales y la generación de señales simuladas, pasando por el diseño de una plataforma de transmisión y recepción con el ADALM-PLUTO, hasta el desarrollo y la validación experimental de dos algoritmos de detección espectral. Cada etapa ha aportado su propio aprendizaje, no solo en los resultados obtenidos, sino también en las decisiones tomadas ante los problemas que han ido surgiendo por el camino, desde fallos de hardware hasta limitaciones propias de las señales disponibles.

Los resultados finales, recogidos en el capítulo 8 (Capítulo 8. ), muestran que el sistema desarrollado es capaz de estimar con precisión los límites de banda de las señales evaluadas, incluso en condiciones de ruido considerablemente exigentes. Al mismo tiempo, las limitaciones identificadas y las líneas de trabajo futuro propuestas en este capítulo dejan claro que este TFG es un punto de partida, no un punto final, para seguir avanzando en la detección y caracterización de señales de drones mediante SDR.

## Capítulo 10. BIBLIOGRAFÍA

- [1] Analog Devices, "ADALM-PLUTO Active Learning Module — Product Overview," Analog Devices Inc., Norwood, MA, USA, 2017.
- [2] D. P. Wright and E. A. Ball, "Highly portable, low-cost SDR instrument for RF propagation studies," *IEEE Trans. Instrum. Meas.*, vol. 68, no. 8, pp. 3006–3013, 2019.
- [3] R. Giuliani, G. Baldini, and D. Symeonidis, "Reconfigurable radio system test bed for security research," Joint Research Centre (JRC), European Commission, 2009.
- [4] A. Aboltins and N. Tihomorskis, "Software-defined radio implementation and performance evaluation of frequency-modulated antipodal chaos shift keying communication system," *Electronics*, vol. 12, no. 5, p. 1240, Mar. 2023, doi: 10.3390/electronics12051240.
- [5] M. Lichtman, *PySDR: A Guide to SDR and DSP Using Python*, 2022. [Online]. Available: <https://pysdr.org/content/pluto.html>.
- [6] G. Léger and C. Wegener, "Guest editorial: Analog, mixed-signal and RF testing," *J. Electron. Test., Theory Appl. (JETTA)*, vol. 32, pp. 405–406, Jul. 2016, doi: 10.1007/s10836-016-5608-y.
- [7] C. Bender, "DJI drone IDs are not encrypted," preprint, The University of Tulsa, 2023.
- [8] H. Shin, K. Choi, Y. Park, J. Choi, and Y. Kim, "Security analysis of FHSS-type drone controller," in *Proc. Int. Conf. Inf. Secur. Appl. (WISA)*, Springer, 2015.
- [9] D. Aouladhadj, E. Kpre, V. Deniau, A. Kharchouf, C. Gransart, and C. Gaquière, "Drone detection and tracking using RF identification signals," *Sensors*, vol. 23, no. 17, p. 7650, Sep. 2023, doi: 10.3390/s23177650.
- [10] Y. Kwon, M. Park, and J. Paek, "A measurement study of ADALM-PLUTO software defined radio with IEEE 802.15.4," in *Proc. 13th Int. Conf. Inf. Commun. Technol. Converg. (ICTC)*, IEEE, 2022, pp. 865–867, doi: 10.1109/ICTC55196.2022.9952931.
- [11] A. Bello, "RF-based drone detection and classification toolbox," M.S. thesis, Dept. Elect. Comput. Eng., Carleton Univ., Ottawa, ON, Canada, 2019.
- [12] B. Taha and A. Shoufan, "Machine learning-based drone detection and classification: State-of-the-art in research," *IEEE Access*, vol. 7, pp. 138669–138682, 2019, doi: 10.1109/ACCESS.2019.2942944.

- [13] J. G. Proakis and M. Salehi, *Digital Communications*, 5th ed. New York, NY, USA: McGraw-Hill Education, 2008.
- [14] A. V. Oppenheim and R. W. Schaffer, *Discrete-Time Signal Processing*, 3rd ed. Harlow, U.K.: Pearson Education Limited, 2014.
- [15] H. Ochiai and H. Imai, "On the distribution of the peak-to-average power ratio in OFDM signals," *IEEE Trans. Commun.*, vol. 49, no. 2, pp. 282–289, Feb. 2001.
- [16] Analog Devices, "PLUTO HDL project — HDL reference design documentation." [Online]. Available: <https://analogdevicesinc.github.io/hdl/projects/pluto/index.html>
- [17] The MathWorks, Inc., "resample — Resample uniform or nonuniform data to new fixed rate," *MATLAB Signal Processing Toolbox Documentation*. [Online]. Available: <https://www.mathworks.com/help/signal/ref/resample.html>
- [18] The MathWorks, Inc., "comm.SDRTxPluto and comm.SDRRxPluto," *MATLAB Communications Toolbox Support Package for Analog Devices ADALM-PLUTO Radio*, Official Documentation. [Online]. Available: <https://www.mathworks.com/help/comm/plutoradio/ref/comm.sdrtxpluto-system-object.html>
- [19] T. S. Rappaport, *Wireless Communications: Principles and Practice*, 2nd ed. Upper Saddle River, NJ, USA: Prentice Hall, 2002.
- [20] L. Ophir, Y. Bitran, and I. Sherman, "Wi-Fi (IEEE 802.11) and Bluetooth coexistence: Issues and solutions," in *Proc. IEEE 60th Veh. Technol. Conf. (VTC 2004-Fall)*, IEEE, 2004, pp. 847–852.
- [21] R. Natarajan, P. Zand, and M. Nabi, "Analysis of coexistence between IEEE 802.15.4, BLE and IEEE 802.11 in the 2.4 GHz ISM band," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, IEEE, 2016, pp. 6025–6032.
- [22] J. M. Merlo, S. Wagner, J. B. Lancaster, and J. A. Nanzer, "Real-time high-accuracy digital wireless time, frequency, and phase calibration for coherent distributed antenna arrays," *IEEE Trans. Microw. Theory Techn.*, early access, 2025.
- [23] Analog Devices EngineerZone, "Multiple ADALM PLUTO SDR's synchronization," ADI University Program Support Forum. [Online]. Available: <https://ez.analog.com/adiacademy/university-program/f/q-a/603307/multiple-adalm-pluto-sdr-s-synchronization>
- [24] The MathWorks, Inc., "movmean — Moving mean," *MATLAB Documentation*. [Online]. Available: <https://www.mathworks.com/help/matlab/ref/movmean.html>

- [25] Wikipedia contributors, "Gatwick Airport drone incident," *Wikipedia, The Free Encyclopedia*. [Online]. Available: [https://en.wikipedia.org/wiki/Gatwick\\_Airport\\_drone\\_incident](https://en.wikipedia.org/wiki/Gatwick_Airport_drone_incident).
- [26] Analog Devices, "AD9363 — Highly integrated RF agile transceiver, data sheet," Analog Devices Inc., 2017. [Online]. Available: <https://www.analog.com/en/products/ad9363.html>
- [27] Xilinx Inc., "Zynq-7000 SoC technical reference manual," AMD-Xilinx, 2018. [Online]. Available: <https://docs.amd.com/v/u/en-US/ug585-Zynq-7000-TRM>
- [28] L. J. Cimini, "Analysis and simulation of a digital mobile channel using orthogonal frequency division multiplexing," *IEEE Trans. Commun.*, vol. COM-33, no. 7, pp. 665–675, Jul. 1985.
- [29] T. Yucek and H. Arslan, "A survey of spectrum sensing algorithms for cognitive radio applications," *IEEE Commun. Surveys Tuts.*, vol. 11, no. 1, pp. 116–130, 2009.
- [30] Z. Tian and G. B. Giannakis, "A wavelet approach to wideband spectrum sensing for cognitive radios," in *Proc. Int. Conf. Cogn. Radio Oriented Wireless Netw. Commun. (CrownCom)*, Mykonos, Greece, Jun. 2006, pp. 1–5.
- [31] H. Urkowitz, "Energy detection of unknown deterministic signals," *Proc. IEEE*, vol. 55, no. 4, pp. 523–531, Apr. 1967.

## Capítulo 11. ANEXO I: ALINEACIÓN DEL PROYECTO CON LOS ODS

Este proyecto contribuye al **ODS 9 (Industria, Innovación e Infraestructura)** mediante el desarrollo de un sistema de detección y caracterización de señales RF de drones basado en Radio Definida por Software, combinando hardware de bajo coste con algoritmos propios de estimación espectral. El uso del ADALM-PLUTO como plataforma experimental demuestra que es posible abordar problemas de detección RF con recursos accesibles, reduciendo la barrera de entrada a este tipo de investigación.

Asimismo, apoya el **ODS 11 (Ciudades y Comunidades Sostenibles)** al contribuir al desarrollo de sistemas de vigilancia del espacio aéreo en entornos urbanos, donde la proliferación de drones comerciales plantea riesgos crecientes para la seguridad de personas e infraestructuras.

De forma complementaria, se relaciona con el **ODS 4 (Educación de Calidad)**, ya que el sistema desarrollado, al estar íntegramente implementado en MATLAB sobre hardware SDR de bajo coste, constituye una herramienta directamente aplicable a la docencia en comunicaciones inalámbricas y procesamiento digital de señal.

## Capítulo 12. ANEXO II

Este anexo recoge los fragmentos de código MATLAB más relevantes desarrollados a lo largo del proyecto, organizados por capítulos. Se omiten las funciones de visualización y generación de figuras, que no forman parte del núcleo algorítmico del sistema.

### 12.1 GENERACIÓN Y CARACTERIZACIÓN DE SEÑALES (CAPÍTULO 5.)

#### 12.1.1 GENERACIÓN DE SEÑAL OFDM SIMULADA

La función `gen_ofdm_like` genera una señal OFDM simulada de banda limitada a partir de los parámetros espectrales de cada dron. Se basa en la asignación de coeficientes gaussianos complejos a los bins de frecuencia comprendidos entre  $f_l$  y  $f_h$ , seguida de una IFFT para obtener la señal en el dominio temporal.

```
function sig = gen_ofdm_like(N, fs, fl, fh)
    f = linspace(-fs/2, fs/2, N)';
    X = zeros(N, 1);
    idx = (f >= fl) & (f <= fh);
    X(idx) = randn(sum(idx), 1) + 1j*randn(sum(idx), 1);
    sig = ifft(ifftshift(X));
    sig = sig / max(abs(sig));
    sig = single(sig);
end
```

#### 12.1.2 CARACTERIZACIÓN ESPECTRAL DE LAS SEÑALES DE REFERENCIA

El script `step03_drone_band_characterization.m` define los parámetros espectrales (frecuencia inferior, superior, central y ancho de banda) de las cinco señales de referencia empleadas en el proyecto.

```
names = {'M250','V2','OCCUSYNC2','MINI2','DJI_NEO'};

fl = [-2.45 -0.47 0.75 -2.351 -1.8] * 1e7;
fh = [ 1.151 0.473 2.55 -0.55 1.8] * 1e7;

fc = (fl + fh)/2;
BW = fh - fl;

T = table(names', fl'/1e6, fh'/1e6, fc'/1e6, BW'/1e6, ...
    'VariableNames', {'Drone','fl_MHz','fh_MHz','fc_MHz','BW_MHz'});
disp(T)
```

## 12.2 PLATAFORMA SDR (CAPÍTULO 6.)

### 12.2.1 REMUESTREO DE SEÑALES PARA EL ADALM-PLUTO

La función `resample_for_pluto` adapta la frecuencia de muestreo de las señales (originalmente a 92,16 MHz) al límite operativo del ADALM-PLUTO (61,44 MHz), mediante una interpolación por 2 y un diezmado por 3.

```
function [txSig, fs] = resample_for_pluto(txSig, fs_original)
    txSig = resample(txSig, 2, 3);
    fs = fs_original * 2 / 3;
    txSig = txSig(:);
    if mod(length(txSig), 2)
        txSig = txSig(1:end-1);
    end
end
```

### 12.2.2 TRANSMISIÓN Y RECEPCIÓN CON EL ADALM-PLUTO

El siguiente fragmento muestra la configuración de los objetos `comm.SDRTxPluto` y `comm.SDRRxPluto` empleados para transmitir y recibir señales con el dispositivo, incluyendo el modo de dos Plutos independientes.

```
tx = comm.SDRTxPluto( ...  
    'RadioID',          txID, ...  
    'CenterFrequency',  fc, ...  
    'BasebandSampleRate', fs, ...  
    'Gain',             txGain);  
  
rx = comm.SDRRxPluto( ...  
    'RadioID',          rxID, ...  
    'CenterFrequency',  fc, ...  
    'BasebandSampleRate', fs, ...  
    'GainSource',       'Manual', ...  
    'Gain',             rxGain, ...  
    'SamplesPerFrame',  N, ...  
    'OutputDataType',   'double');  
  
transmitRepeat(tx, txSig);  
pause(1);  
rxSig = rx();  
  
release(tx);  
release(rx);
```

## ***12.3 ALGORITMOS DE DETECCIÓN ESPECTRAL (CAPÍTULO 7.)***

### **12.3.1 ALGORITMO 1: MEDIA MÓVIL Y DERIVADA**

```
function ventana = calcular_ventana(N, modo_porcentaje, porcentaje, valor_fijo)  
    if modo_porcentaje == 1  
        ventana = round(N * porcentaje / 100);  
        ventana = max(ventana, 10);  
    else  
        ventana = valor_fijo;  
    end  
end  
  
function S_smooth = suavizado_movmean(S, ventana)  
    S_smooth = movmean(S, ventana);  
end  
  
function [derivada, derivada_smooth] = calcular_derivada_multi_suavizada( ...  
    S_smooth, step, ventana, porc_suavizado_deriv)  
    derivada = zeros(length(S_smooth), 1);  
    for i = 1:length(S_smooth) - step  
        derivada(i) = S_smooth(i + step) - S_smooth(i);  
    end  
    ventana_deriv = max(round(ventana / porc_suavizado_deriv), 3);  
    derivada_smooth = movmean(derivada, ventana_deriv);  
end
```

```
function [idx_subida, idx_bajada] = detectar_bordes( ...
    S_smooth, derivada_smooth, porc_umbral, modo)
    umbral_subida = porc_umbral * max(derivada_smooth);
    umbral_bajada = porc_umbral * min(derivada_smooth);

    if modo == 1
        idx_subida = find(derivada_smooth > umbral_subida, 1, 'first');
        if isempty(idx_subida)
            [~, idx_subida] = max(derivada_smooth);
        end
        [~, idx_peak] = max(S_smooth);
        region_bajada = derivada_smooth(idx_peak:end);
        idx_bajada_rel = find(region_bajada < umbral_bajada, 1, 'first');
        if isempty(idx_bajada_rel)
            [~, idx_bajada_rel] = min(derivada_smooth(idx_peak:end));
        end
        idx_bajada = idx_peak + idx_bajada_rel - 1;

    elseif modo == 2
        idx_bajada = find(derivada_smooth < umbral_bajada, 1, 'first');
        if isempty(idx_bajada)
            [~, idx_bajada] = min(derivada_smooth);
        end
        region_subida = derivada_smooth(1:idx_bajada);
        idx_subida = find(region_subida > umbral_subida, 1, 'last');
        if isempty(idx_subida)
            [~, idx_subida] = max(derivada_smooth(1:idx_bajada));
        end
    end

    if idx_subida > idx_bajada
        [idx_subida, idx_bajada] = deal(idx_bajada, idx_subida);
    end
end
```

### 12.3.2 ALGORITMO 2: SUAVIZADO POR BLOQUES Y DETECCIÓN POR STEP

```
function S_smooth = suavizado_bloques(S, N, ventana)
    S_smooth = zeros(N, 1);
    for i = 1:ventana:N
        fin = min(i + ventana - 1, N);
        S_smooth(i:fin) = mean(S(i:fin));
    end
end
```

```
function [f_low, f_high] = deteccion_bloques_step( ...
    S_smooth, f, ventana, step, modo_algoritmo)
N = length(S_smooth);
num_bloques = ceil(N / ventana);

valores_bloques = zeros(num_bloques, 1);
frecuencias_bloques = zeros(num_bloques, 1);
for i = 1:num_bloques
    idx_centro = min(i * ventana, N);
    valores_bloques(i) = S_smooth(idx_centro);
    frecuencias_bloques(i) = f(idx_centro);
end

num_diferencias = num_bloques - step;
diferencias = zeros(num_diferencias, 1);
frecuencias_dif = zeros(num_diferencias, 1);
for i = 1:num_diferencias
    diferencias(i) = valores_bloques(i+step) - valores_bloques(i);
    if mod(step,2) == 1
        frecuencias_dif(i) = frecuencias_bloques(i + (step-1)/2);
    else
        frecuencias_dif(i) = (frecuencias_bloques(i+step/2-1) + ...
            frecuencias_bloques(i+step/2)) / 2;
    end
end

if modo_algoritmo == 1
    [~, idx_subida] = max(diferencias);
    [~, idx_min_rel] = min(diferencias(idx_subida:end));
    idx_bajada = idx_subida + idx_min_rel - 1;
else
    [~, idx_bajada] = min(diferencias);
    [~, idx_subida] = max(diferencias(1:idx_bajada));
end

f_low = frecuencias_dif(idx_subida);
f_high = frecuencias_dif(idx_bajada);
end

function [f_low, f_high, fc, BW, res_fl, res_fh] = deteccion_por_consistencia(
    ...
    S_smooth, f, ventana, steps, modo_algoritmo)
res_fl = zeros(length(steps), 1);
res_fh = zeros(length(steps), 1);
for s = 1:length(steps)
    [res_fl(s), res_fh(s)] = deteccion_bloques_step( ...
        S_smooth, f, ventana, steps(s), modo_algoritmo);
end
f_low = median(res_fl);
f_high = median(res_fh);
BW = abs(f_high - f_low);
fc = (f_high + f_low) / 2;
end
```

## ***12.4 VALIDACIÓN EXPERIMENTAL (CAPÍTULO 8.)***

### **12.4.1 TRANSMISIÓN Y RECEPCIÓN DE TODAS LAS SEÑALES DEL PROYECTO**

El script `step03_transmit_all_signals.m` itera sobre los tres tipos de señal (simulada, real sin filtrar y real filtrada) y los cinco drones de referencia, transmite cada señal mediante el ADALM-PLUTO y almacena las capturas recibidas para su análisis posterior.

```

fc      = 868e6;
txGain  = 0;
rxGain  = 20;
iterations = 10;
A_tx    = 0.5;

types = {'simulated', 'real', 'real_filtered'};

for t = 1:length(types)
    type = types{t};
    drones = fieldnames(all_data_resampled.(type));

    for d = 1:length(drones)
        drone = drones{d};
        s_res = all_data_resampled.(type).(drone).signal;
        fs_pluto = all_data_resampled.(type).(drone).fs;

        txSig = A_tx * s_res / max(abs(s_res));
        txSig = txSig(:);
        if mod(length(txSig), 2), txSig = txSig(1:end-1); end
        N = length(txSig);

        tx = comm.SDRTxPluto( ...
            'CenterFrequency', fc, ...
            'BasebandSampleRate', fs_pluto, ...
            'Gain', txGain);

        rx = comm.SDRRxPluto( ...
            'CenterFrequency', fc, ...
            'BasebandSampleRate', fs_pluto, ...
            'GainSource', 'Manual', ...
            'Gain', rxGain, ...
            'SamplesPerFrame', N, ...
            'OutputDataType', 'double');

        transmitRepeat(tx, txSig);
        pause(0.5);

        rxSig = zeros(N, iterations);
        for k = 1:iterations
            rxSig(:,k) = rx();
        end

        release(tx); release(rx);

        filename = sprintf('rx_%s_%s.mat', type, drone);
        save(fullfile(outputFolder, filename), ...
            'rxSig', 'txSig', 'fs_pluto', 'fc', 'txGain', 'rxGain',
            'iterations');
    end
end
end

```