

UNIVERSIDAD POLITÉCNICA DE CARTAGENA

Escuela Técnica Superior de Ingeniería de Telecomunicación

Aplicación de técnicas monopulso en sistemas RFID pasivos para localización angular

María Turpín Buendía



Resumen

La tecnología RFID UHF pasiva se trata de un sistema de identificación de productos y almacenamiento de información bastante maduro en la actualidad desde el punto de vista tecnológico. Es usado en multitud de escenarios con el fin de identificar y trazar objetos y personas empleando tags pasivos mediante la técnica de “back-scattering” o recepción de ecos. No obstante, la capacidad de estos sistemas de determinar la localización de dichos objetos o personas todavía tiene una resolución bastante limitada.

Algunas de las soluciones actuales a la localización e identificación por radiofrecuencia (RFID+L, “*Radio Frequency IDentification and Localization*”) implican el uso de Dispositivos de Radio Definido por Software (USRP, “*Universal Software Radio Peripheral*”), que resultan demasiado costosos y dedicados.

El objetivo principal de este proyecto es demostrar que, mediante la utilización de readers comerciales (COTS, “*Commercial off-the-shelf or commercially available off-the-shelf*”) y de antenas de tipo panel, es posible aplicar técnicas monopolso para realizar la estimación del ángulo en el que se encuentra el tag respecto del lector con bastante precisión angular. Para ello, se presenta el desarrollo de un software para la adquisición de datos de amplitud (Received Signal Strength Indicator), fase de dos puertos y canal de transmisión en la banda europea de frecuencias de un sistema RFID pasivo capaz de medir diagramas de radiación digitales de forma automatizada y de adquirir datos independientes. Así como el procesado de señal monopolso que permite la estimación de localización de los tags de forma precisa.

Índice de contenido

Conceptos necesarios y abreviaturas.....	13
Sistema monopolso	13
Received Signal Strength Indicator	13
Field Of View	13
Direction Of Arrival.....	14
Capítulo 1. Tecnología RFID.....	15
1.1. Funcionamiento de un sistema RFID.....	15
1.2. Clasificación de sistemas RFID	16
1.3. Aplicaciones actuales.....	16
1.4. Ventajas y desventajas de la tecnología RFID.....	17
Capítulo 2. Software de adquisición	19
2.1. Disposición	19
2.2. Protocolos de comunicación entre las herramientas	20
2.3. MATLAB.....	21
2.4. Eclipse (Java).....	23
Capítulo 3. Cámara Anecoica.....	26
3.1. Instrumentación empleada	26
3.1.1. Controlador y mesa giratoria.....	27
3.1.2. Analizador de redes	28
3.1.3. Antenas de panel	29
3.1.4. Antena bicónica.....	30
3.1.5. Reader Impinj R420	31
3.1.6. Tags	32
3.2. Obtención de los diagramas analógicos.....	32
3.2.1. Caracterización de las antenas por separado	32
3.2.2. Caracterización de las antenas en configuración monopolso.....	36
3.3. Obtención de los diagramas digitales	39
Capítulo 4. Aplicación de técnicas monopolso y procesado para localización angular	44
4.1. Función monopolso.....	44

4.2. Valor monopolso y error monopolso	46
4.3. Pruebas de localización angular	47
Capítulo 5. Experimento de localización angular de 3 tags	58
5.1. Localización angular del tag 1	59
5.1.1. Canal 1 (865,7 MHz).....	59
5.1.2. Canal 2 (866,3 MHz).....	62
5.1.3. Canal 3 (866,9 MHz).....	65
5.1.4. Canal 4 (867,5 MHz).....	68
5.2. Localización angular del tag 2	71
5.2.1. Canal 1 (865,7 MHz).....	71
5.2.2. Canal 2 (866,3 MHz).....	74
5.2.3. Canal 3 (866,9 MHz).....	77
5.2.4. Canal 4 (867,5 MHz).....	79
5.3. Localización angular del tag 3	82
5.3.1. Canal 1 (865,7 MHz).....	83
5.3.2. Canal 2 (866,3 MHz).....	85
5.3.3. Canal 3 (866,9 MHz).....	88
5.3.4. Canal 4 (867,5 MHz).....	91
5.4. Análisis general de los resultados	94
Capítulo 6. Conclusiones y líneas futuras	95
Bibliografía	96
Anexo I. Clases Java.....	98
I.I. MainServidor.....	98
I.II. Conexion	98
I.III. Servidor	99
I.IV. Config.....	100
I.V. Start.....	101
I.VI. HelloOctaneSdk	107
I.VI. TagReportListenerImplementation	108
Anexo II. MATLAB	111
II.I. Cliente.m.....	111

II.II. Conexion.m	112
II.III. gira_mesa.m	112
II.IV. mediaRSSI_dBm.m	112
Anexo III. Hojas de Características	115

Índice de Figuras

<i>Figura 1.</i> Field Of View	14
<i>Figura 1.1.</i> Partes de un sistema RFID	15
<i>Figura 2.1.</i> Disposición de los dispositivos	19
<i>Figura 2.2.</i> Estructura de "datos_medidaX"	21
<i>Figura 2.3.</i> RSSI media para la antena 1 en el canal 1.....	22
<i>Figura 2.4.</i> RSSI media para la antena 2 en el canal 1.....	22
<i>Figura 2.5.</i> Representación de la RSSI media obtenida.....	23
<i>Figura 2.6.</i> Diagrama de clases UML	24
<i>Figura 2.7.</i> Consola de Eclipse durante la adquisición	25
<i>Figura 3.1.</i> Cámara Anecoica.....	26
<i>Figura 3.2.</i> Compact Table CT 0800-P.....	27
<i>Figura 3.3.</i> Controlador MCU <i>Maturo</i>	27
<i>Figura 3.4.</i> Interfaz de <i>Keysight Connection Expert</i>	28
<i>Figura 3.5.</i> Analizador de Redes <i>Agilent Technologies</i>	28
<i>Figura 3.6.</i> Antena de panel Rain RFID de <i>Laird</i>	29
<i>Figura 3.7.</i> Antena bicónica SBA 9113	30
<i>Figura 3.8.</i> Reader <i>Impinj Speedway Revolution R420</i>	31
<i>Figura 3.9.</i> Tag pasivo Omni-ID.....	32
<i>Figura 3.10.</i> Disposición para medida analógica de una antena de panel	33
<i>Figura 3.11.</i> Diagramas de radiación analógicos de las 4 antenas de panel en el canal 1 ...	33
<i>Figura 3.12.</i> Diagramas de radiación analógicos de la antena 1	34
<i>Figura 3.13.</i> Diagramas de radiación analógicos de la antena 2.....	34
<i>Figura 3.14.</i> Diagramas de radiación analógicos de la antena 3.....	35
<i>Figura 3.15.</i> Diagramas de radiación analógicos de la antena 4.....	35
<i>Figura 3.16.</i> Antenas 1 y 4 en configuración monopulso	36

<i>Figura 3.17.</i> Disposición de los elementos para la obtención de los diagramas en configuración monopulso	37
<i>Figura 3.18.</i> Diagramas analógicos en configuración monopulso.....	38
<i>Figura 3.19.</i> Diagramas de radiación analógicos en forma polar	38
<i>Figura 3.20.</i> Disposición de los instrumentos para la toma de datos digitales	39
<i>Figura 3.21.</i> Interior de la cámara anecoica.....	40
<i>Figura 3. 22.</i> Diagramas de radiación digitales para un tilt lateral de 30°	40
<i>Figura 3.23.</i> Antenas en configuración monopulso para las medidas digitales.....	41
<i>Figura 3.24.</i> Diagramas de radiación digitales.....	42
<i>Figura 3.25.</i> Diagramas de radiación digitales normalizados	43
<i>Figura 3.26.</i> Field Of View obtenido	43
<i>Figura 4.1.</i> Obtención del coeficiente de corrección digital	45
<i>Figura 4.2.</i> Funciones monopulso.....	45
<i>Figura 4.3.</i> Error monopulso en el ángulo 0°	47
<i>Figura 4.4.</i> Valores monopulso en el ángulo 0°	48
<i>Figura 4.5.</i> Pseudo espectros angulares en el ángulo 0°	48
<i>Figura 4.6.</i> DoA estimadas en el ángulo 0°	49
<i>Figura 4.7.</i> Errores de estimación del DoA en el ángulo 0°.....	49
<i>Figura 4.8.</i> Error monopulso en el ángulo -15°.....	50
<i>Figura 4.9.</i> Valores monopulso en el ángulo -15°.....	50
<i>Figura 4.10.</i> Pseudoespectros angulares para el ángulo -15°	51
<i>Figura 4.11.</i> DoA estimadas para el ángulo -15°	51
<i>Figura 4.12.</i> Errores de estimación del DoA en el ángulo -15°	52
<i>Figura 4.13.</i> Error monopulso en el ángulo 27°.....	52
<i>Figura 4.14.</i> Valores monopulso en el ángulo 27°.....	53
<i>Figura 4.15.</i> Pseudoespectros angulares para el ángulo 27°	53
<i>Figura 4.16.</i> DoA estimadas para el ángulo 27°.....	54
<i>Figura 4.17.</i> Errores de estimación del DoA en el ángulo 27°.....	54

<i>Figura 4.18.</i> Error monopolso para el ángulo -50°	55
<i>Figura 4.19.</i> Valores monopolso para el ángulo -50°	55
<i>Figura 4.20.</i> Pseudoespectros angulares para el ángulo -50°	56
<i>Figura 4.21.</i> DoA estimadas para el ángulo -50°	56
<i>Figura 4.22.</i> Error de estimación del DoA para el ángulo -50°	57
<i>Figura 5.1.</i> Posición de los tags en el experimento de localización.....	58
<i>Figura 5.2.</i> Evolución de las RSSI del tag 1 en el canal 1.....	59
<i>Figura 5.3.</i> Errores monopolso del tag 1 en el canal 1.....	60
<i>Figura 5.4.</i> Valores monopolso del tag 1 en el canal 1.....	60
<i>Figura 5.5.</i> Valores monopolso exactos del tag 1 en el canal 1.....	61
<i>Figura 5.6.</i> Pseudoespectros angulares para el tag 1 en el canal 1.....	61
<i>Figura 5.7.</i> DoA estimadas para el tag 1 en el canal 1.....	61
<i>Figura 5.8.</i> Errores cometidos para el tag 1 en el canal 1.....	62
<i>Figura 5.9.</i> Evolución de la RSSI del tag 1 en el canal 2.....	62
<i>Figura 5.10.</i> Errores monopolso para el tag 1 en el canal 2.....	63
<i>Figura 5.11.</i> Valores monopolso para el tag 1 en el canal 2.....	63
<i>Figura 5.12.</i> Valores monopolso exactos para el tag 1 en el canal 2.....	64
<i>Figura 5.13.</i> Pseudoespectros angulares para el tag 1 en el canal 2.....	64
<i>Figura 5.14.</i> DoA estimadas para el tag 1 en el canal 2.....	64
<i>Figura 5.15.</i> Errores de estimación para el tag 1 en el canal 2.....	65
<i>Figura 5.16.</i> Evolución de las RSSI para el tag 1 en el canal 3.....	65
<i>Figura 5.17.</i> Error monopolso para el tag 1 en el canal 3.....	66
<i>Figura 5.18.</i> Valores monopolso para el tag 1 en el canal 3.....	66
<i>Figura 5.19.</i> Valores monopolso exactos para el tag 1 en el canal 3.....	67
<i>Figura 5.20.</i> Pseudoespectros angulares para el tag 1 en el canal 3.....	67
<i>Figura 5.21.</i> DoA estimadas para el tag 1 en el canal 3.....	67
<i>Figura 5.22.</i> Errores de estimación para el tag 1 en el canal 3.....	68
<i>Figura 5.23.</i> Evolución de la RSSI para el tag 1 en el canal 4.....	68

<i>Figura 5.24.</i> Errores monopolso para el tag 1 en el canal 4.....	69
<i>Figura 5.25.</i> Valores monopolso para el tag 1 en el canal 4	69
<i>Figura 5.26.</i> Valores monopolso exactos para el tag 1 en el canal 4	69
<i>Figura 5.27.</i> Pseudoespectros angulares para el tag 1 en el canal 4.....	70
<i>Figura 5.28.</i> DoA estimadas para el tag 1 en el canal 4.....	70
<i>Figura 5.29.</i> Errores de estimación para el tag 1 en el canal 4	71
<i>Figura 5.30.</i> Evolución de la RSSI para el tag 2 en el canal 1.....	71
<i>Figura 5.31.</i> Funciones de error para el tag 2 en el canal 1	72
<i>Figura 5.32.</i> Valores monopolso para el tag 2 en el canal 1	72
<i>Figura 5.33.</i> Valores monopolso exactos para el tag 2 en el canal 1	73
<i>Figura 5.34.</i> Pseudoespectros angulares para el tag 2 en el canal 1.....	73
<i>Figura 5.35.</i> DoA estimadas para el tag 2 en el canal 1.....	73
<i>Figura 5.36.</i> Errores de estimación para el tag 2 en el canal 1	74
<i>Figura 5.37.</i> Evolución de la RSSI para el tag 2 en el canal 2.....	74
<i>Figura 5.38.</i> Errores monopolso para el tag 2 en el canal 2.....	75
<i>Figura 5.39.</i> Valores monopolso para el tag 2 en el canal 2	75
<i>Figura 5.40.</i> Valores monopolso exactos para el tag 2 en el canal 2	75
<i>Figura 5.41.</i> Pseudoespectros angulares para el tag 2 en el canal 2.....	76
<i>Figura 5. 42.</i> DoA estimadas para el tag 2 en el canal 2	76
<i>Figura 5.43.</i> Errores de estimación para el tag 2 en el canal 2	76
<i>Figura 5.44.</i> Evolución de la RSSI para el tag 2 en el canal 3.....	77
<i>Figura 5.45.</i> Errores monopolso para el tag 2 en el canal 3.....	77
<i>Figura 5.46.</i> Valores monopolso para el tag 2 en el canal 3	78
<i>Figura 5.47.</i> Valores monopolso exactos para el tag 2 en el canal 3	78
<i>Figura 5.48.</i> Pseudoespectros angulares para el tag 2 en el canal 3.....	78
<i>Figura 5.49.</i> DoA estimadas para el tag 2 en el canal 3.....	79
<i>Figura 5.50.</i> Errores de estimación para el tag 2 en el canal 3	79
<i>Figura 5.51.</i> Evolución de las RSSI para el tag 2 en el canal 4	80

<i>Figura 5.52.</i> Funciones de error para el tag 2 en el canal 4	80
<i>Figura 5.53.</i> Valores monopolso para el tag 2 en el canal 4	81
<i>Figura 5.54.</i> Valores monopolso exactos para el tag 2 en el canal 4	81
<i>Figura 5.55.</i> Pseudoespectros angulares para el tag 2 en el canal 4.....	81
<i>Figura 5.56.</i> DoA estimadas para el tag 2 en el canal 4.....	82
<i>Figura 5.57.</i> Errores de estimación para el tag 2 en el canal 4	82
<i>Figura 5.58.</i> Evolución de las RSSI para el tag 3 en el canal 1	83
<i>Figura 5.59.</i> Errores monopolso para el tag 3 en el canal 1	83
<i>Figura 5.60.</i> Valores monopolso para el tag 3 en el canal 1	84
<i>Figura 5.61.</i> Valores monopolso exactos para el tag 3 en el canal 1	84
<i>Figura 5.62.</i> Pseudoespectros angulares para el tag 3 en el canal 1.....	84
<i>Figura 5.63.</i> DoA estimadas para el tag 3 en el canal 1	85
<i>Figura 5.64.</i> Errores cometidos para el tag 3 en el canal 1	85
<i>Figura 5.65.</i> Evolución de las RSSI para el tag 3 en el canal 2	86
<i>Figura 5.66.</i> Errores monopolso para el tag 3 en el canal 2.....	86
<i>Figura 5.67.</i> Valores monopolso para el tag 3 en el canal 2	87
<i>Figura 5.68.</i> Valores monopolso exactos para el tag 3 en el canal 2	87
<i>Figura 5.69.</i> Pseudoespectros angulares para el tag 3 en el canal 2.....	87
<i>Figura 5.70.</i> DoA estimadas para el tag 3 en el canal 2.....	88
<i>Figura 5.71.</i> Errores de estimación para el tag 3 en el canal 2	88
<i>Figura 5.72.</i> Evolución de las RSSI para el tag 3 en el canal 3	89
<i>Figura 5.73.</i> Errores monopolso para el tag 3 en el canal 3.....	89
<i>Figura 5.74.</i> Valores monopolso para el tag 3 en el canal 3	90
<i>Figura 5.75.</i> Valores monopolso exactos para el tag 3 en el canal 3	90
<i>Figura 5.76.</i> Pseudoespectros angulares para el tag 3 en el canal 3.....	90
<i>Figura 5.77.</i> DoA estimadas para el tag 3 en el canal 3.....	91
<i>Figura 5.78.</i> Errores de estimación para el tag 3 en el canal 3	91
<i>Figura 5.79.</i> Evolución de las RSSI para el tag 3 en el canal 4	92

<i>Figura 5.80.</i> Valores monopolso para el tag 3 en el canal 4	92
<i>Figura 5.81.</i> Valores monopolso exactos para el tag 3 en el canal 4	93
<i>Figura 5.82.</i> Errores monopolso para el tag 3 en el canal 4.....	93
<i>Figura 5.83.</i> Pseudoespectros angulares para el tag 3 en el canal 4.....	93
<i>Figura 5.84.</i> DoA estimadas para el tag 3 en el canal 4.....	94
<i>Figura 5.85.</i> Errores de estimación para el tag 3 en el canal 4	94

Índice de Tablas

<i>Tabla 3.1.</i> Especificaciones Antena de panel	29
<i>Tabla 3.2.</i> Especificaciones técnicas de la Antena bicónica	30
<i>Tabla 3.3.</i> Ficha técnica del Reader <i>Impinj R420</i>	31

Conceptos necesarios y abreviaturas

Sistema monopolso

Un sistema monopolso es aquel capaz de estimar la posición angular, en este caso de un tag, empleando un único pulso. Este tipo de sistemas son idóneos para determinar el ángulo de llegada mediante la comparación de varias señales que se reciben en un instante de tiempo desde distintos haces. El sistema monopolso empleado en este proyecto se basa en la comparación de la amplitud de la señal recibida empleando para ello dos antenas.

Received Signal Strength Indicator

El Received Signal Strength Indicator (RSSI) es un indicador de la potencia de la señal recibida en un sistema típicamente inalámbrico. Nos servirá para estimar el nivel de potencia recibida por cada antena procedente del tag. A lo largo de este trabajo, nos referiremos a este concepto por su abreviatura RSSI.

Field Of View

Se define como Field Of View el rango angular en el que se puede estimar la dirección de llegada de forma inequívoca. Este rango angular, si nos centramos en un caso en el que los diagramas de radiación de dos antenas en configuración monopolso son digitales, queda determinado por la zona en la que tenemos información sobre las RSSI tanto de la antena 1 como de la antena 2. De forma generalizada, en una configuración monopolso de dos antenas donde los diagramas de radiación se cortan a -3 dB en el ángulo 0, este Field Of View o campo de visión comprenderá el rango angular entre los máximos de los diagramas de radiación de las antenas, esto es, equivaldrá al ancho a -3 dB de una antena.

La abreviatura empleada para Field Of View será FoV.

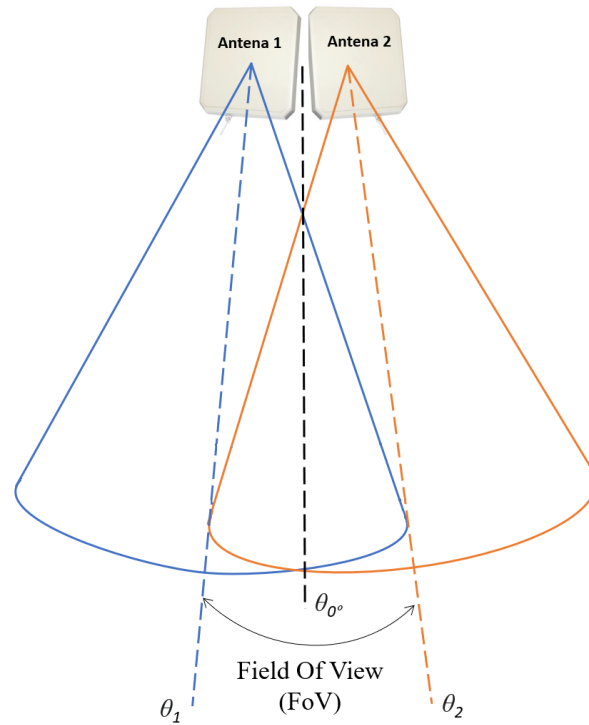


Figura 1. Field Of View

Direction Of Arrival

Cuando hablemos de Direction Of Arrival nos referiremos a la estimación del ángulo en el que se encuentra el tag pasivo RFID con respecto a la perpendicular, que será el punto de unión de los diagramas de radiación de las dos antenas en configuración monopulso.

Capítulo 1. Tecnología RFID

La tecnología RFID, o identificación por radiofrecuencia (en inglés, *Radio Frequency Identification*), se trata de un sistema de almacenamiento e identificación de productos, así como de recuperación de datos que emplea dispositivos llamados tags o etiquetas RFID, que se adhieren al producto u objeto a identificar y que contienen información sobre el mismo. Esta identificación se realiza a través de antenas que vienen incorporadas en dichos tags y que permiten que estos se comuniquen recibiendo y respondiendo a peticiones que se realizan desde un emisor o receptor RFID mediante radiofrecuencia. Típicamente, estos receptores y emisores son antenas.

Esta tecnología, en esencia, es similar a la de un código de barras con lectura por infrarrojos, los cuales se pueden considerar antecedentes de esta tecnología. No obstante, actualmente se ha impuesto la tecnología RFID sobre la de códigos de barra debido al gran número de ventajas que presenta sobre ésta última: no requiere de visión directa entre el código y el lector y son capaces de ser reprogramados, de almacenar una cantidad de datos mucho mayor, presentan una alta dificultad a ser duplicados, lo cual hace que la seguridad del sistema aumente drásticamente, y la lectura de los tags se hace de forma casi instantánea debido a las altas frecuencias en las que se puede implementar esta tecnología.

1.1. Funcionamiento de un sistema RFID

Un sistema RFID se compone principalmente de una antena RFID, un lector o reader RFID y una etiqueta o tag RFID.



Figura 1.1. Partes de un sistema RFID

Como se puede apreciar en la figura 1. 1., para que un sistema RFID funcione correctamente debemos disponer de un dispositivo fundamental, que es el lector RFID o reader RFID. Éste es el encargado de poner en funcionamiento la antena RFID y de recibir la información que le llega de vuelta del tag, que será la información de interés que contendrá el identificador del producto en cuestión.

Las antenas RFID enviarán ondas de radiofrecuencia de una determinada longitud de onda que llegarán a la etiqueta o tag. Cuando estas ondas de radio interactúan con éste, serán enviadas de vuelta a la antena con una cierta modulación, que será única para cada identificador, ya que cada tag incluye un chip integrado, grabable y que se utiliza como parte de un sistema de gestión o seguimiento de los productos que vienen etiquetados.

Por supuesto, este sistema debe estar gestionado por un ordenador junto con una base de datos que contenga la información pertinente para procesar los datos que provienen del lector.

1.2. Clasificación de sistemas RFID

Los sistemas RFID se pueden clasificar en función del rango de frecuencias de trabajo de los mismos. En general, existen cuatro tipos de sistemas: los de baja frecuencia (LF), que emplean el rango de entre los 125 o 134,2 kHz, los de alta frecuencia (HF) que oscilan entre los 13,56 MHz, la de frecuencia ultra elevada (UHF), con un rango de frecuencias entre 868 hasta 956 MHz y, finalmente, el sistema de microondas, que opera en los 2,45 GHz.

A lo largo de este trabajo, se empleará la conocida como banda europea de frecuencias, compatible con la frecuencia de operación de las antenas de panel involucradas, y que se puede clasificar como sistema de alta frecuencia, incluyendo cuatro canales de frecuencia: 856,7 MHz, 866,3 MHz, 866,9 MHz y 867,5 MHz.

1.3. Aplicaciones actuales

Hoy en día, son muchas las empresas que tienen implantados sistemas RFID. Incluso en tareas cotidianas podríamos estar usando tecnología RFID sin ni siquiera percatarnos, como al realizar pagos o al acceder a un edificio. En esta sección se mencionarán algunas de las aplicaciones más implantadas actualmente de este tipo de sistemas.

- En logística y control de inventarios, cada vez son más las empresas que se suman al uso de esta tecnología debido a la ventaja que proporciona sobre el seguimiento de los productos que hay en los almacenes y la documentación acerca de su movilidad dentro del mismo. Gracias a estos sistemas la reducción de errores ha sido drástica con respecto a otros sistemas anteriormente implantados.

- En la industria de la alimentación, la tecnología RFID también juega un papel fundamental, debido a que en este sector se puede controlar fácilmente la caducidad y envasado de los productos, así como la cantidad de lotes y la distribución y venta de los mismos. Dado que un sistema RFID proporciona información inmediata y actualizada de los elementos etiquetados.
- Una de las aplicaciones más cercanas a la vida cotidiana es el de las tarjetas de crédito. En las tarjetas se puede implantar RFID de tal forma que se pueda tener un control sobre el número de extracciones y localizaciones de dónde han sucedido, lo cual puede resultar bastante útil en caso de robo.
- Al igual que ocurría en logística de almacenes, esta aplicación se puede extrapolar a sectores como hospitales, para el control de los productos sanitarios y análisis e incluso en sectores como la ganadería, para tener un control de la sanidad de los animales al igual que se hacía con la industria alimentaria y las fechas de caducidad.
- Los sistemas RFID se pueden emplear para controlar el aforo y los accesos de cualquier evento o instalación, pudiendo discernir si la persona entra o sale, y si tiene permisos para hacerlo.
- Por último, cabe mencionar que el RFID se ha implantado en gran número de tiendas, que fueron pioneras en utilizar etiquetas RFID. El objetivo era identificar los productos que se encontraban tanto en el interior de esta, o que se habían vendido o distribuido.

1.4. Ventajas y desventajas de la tecnología RFID

Una de las principales ventajas de emplear tecnología RFID frente a otros sistemas, como se ha mencionado anteriormente, es la de que no es necesario un contacto directo entre el lector y el producto, ya que se emplea radiofrecuencia en lugar de tecnología de infrarrojos.

Por otro lado, se ha mencionado que la velocidad a la que se pueden adquirir los datos deseados es muy elevada, tanto que podría considerarse que operan en tiempo real y el usuario no es capaz de percibir el lapso de tiempo en el que se adquiere la información, ya que la velocidad de respuesta en algunos casos llega a ser incluso menor de 100 ms.

A pesar de todas las ventajas que proporciona este tipo de sistemas, el uso más implantado de los mismos se centra en la identificación de objetos. Es por esto que no existen muchas soluciones para la localización de los mismos (RFID+L, “Radio Frecuency IDentification and Localization”) empleando estos sistemas y uno de los objetivos de este trabajo es el de demostrar que se pueden aplicar técnicas monopulso para estimar la localización angular mediante dos antenas de panel, que se emplean para recibir información de los tags, y readers

de uso comercial. Además, esta técnica monopolso podría extenderse al uso de 4 antenas para tener una localización total de la posición de las etiquetas identificadoras, pudiendo obtener así información tanto del producto como de su posición bajo unas determinadas condiciones.

Capítulo 2. Software de adquisición

Conviene clarificar el funcionamiento del software requerido para la adquisición de los datos de amplitud (RSSI, *Received Signal Strength Indicator*), canal de transmisión y puerto usando un tag RFID pasivo. El uso de diferentes herramientas de desarrollo software ha sido indispensable para la correcta programación del *Reader Impinj R420* y el posterior procesamiento de los datos obtenidos, así como la compatibilidad de versiones entre las mismas.

2.1. Disposición

Para la comprobación experimental del funcionamiento del software, se han empleado dos antenas direccionales de panel de la empresa *Laird* que operan en la banda de 865-868 MHz. En su hoja de características (*Anexo III*) se refleja que poseen una ganancia de 8,5 dBic y un ancho de haz de 70 grados. También es imprescindible el uso de un Reader digital *Impinj R420* que configuraremos a través del software como se explicará en el capítulo 2.4. La configuración y adquisición de los datos se realizará gracias a un ordenador portátil donde está implementado el software en Eclipse y MATLAB. Por último, es necesario emplear un tag RFID pasivo, de la marca Omni-ID.

La disposición de todos estos elementos viene detallada en la Figura 2.1., así como sus conexiones.

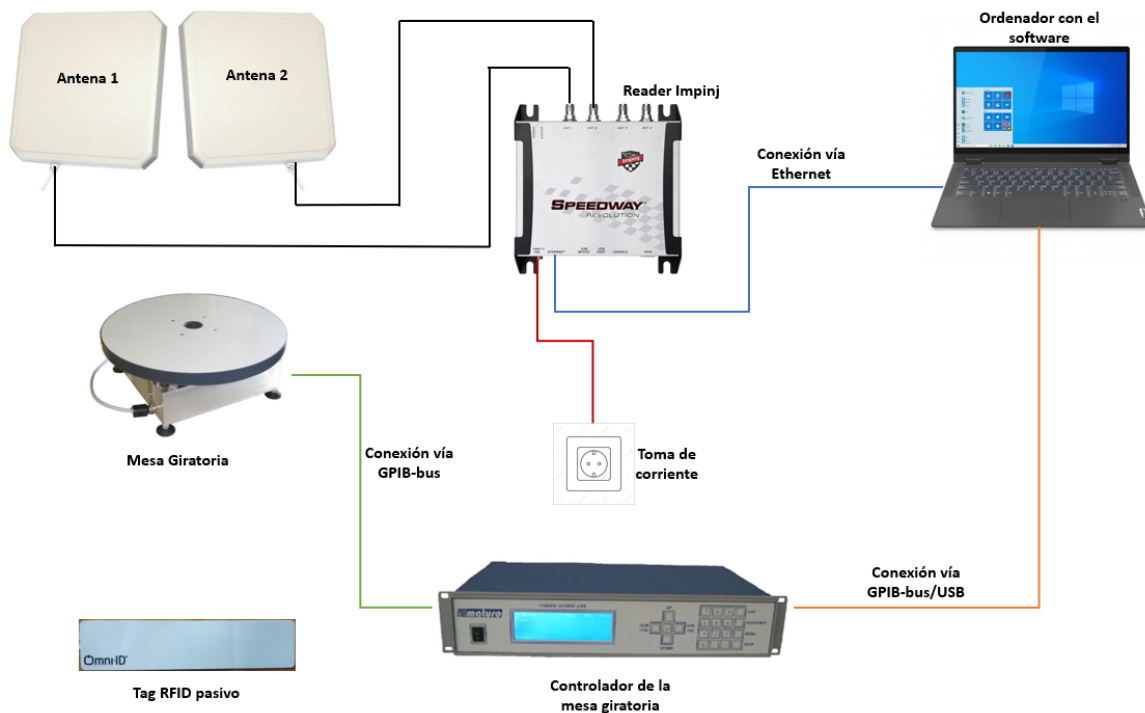


Figura 2.1. Disposición de los dispositivos

En las entradas pertinentes del Reader irán conectadas las antenas de panel 1 y 2, que se situarán en configuración monopulso frente al tag pasivo, que deberá distanciarse de ellas varios metros.

A su vez, el Reader estará conectado a una toma de corriente y mediante su puerto Ethernet a un ordenador portátil que contiene el software necesario para la correcta realización experimental de este trabajo.

El control de la mesa giratoria se realizará mediante un software en MATLAB con conexión remota de tipo TCP/IP, por tanto, el controlador de la empresa *Maturo* de la mesa giratoria irá conectado mediante un bus GPIB con adaptador USB al ordenador portátil. De igual forma, el controlador se conectará con la mesa mediante un bus GPIB que enviará los correspondientes comandos para su funcionamiento de giro en el ángulo de interés.

A continuación, se detallará en profundidad el comportamiento y funcionamiento del software.

2.2. Protocolos de comunicación entre las herramientas

La comunicación entre las distintas herramientas de desarrollo se fundamenta en el protocolo TCP IP Cliente/Servidor. En este caso, al emplear todas las plataformas en un mismo ordenador, solamente será necesario implementar la parte correspondiente al Servidor. De esta forma ha sido posible el traspaso de información desde Matlab tanto a Eclipse, que será el entorno en el que se realiza la toma de datos del Reader, como al controlador de la mesa giratoria, que hará posible que las antenas giren de -95 a 95 grados de tal forma que se pueda obtener el diagrama de radiación digital de las antenas en configuración monopulso.

El uso de Sockets de Java ha sido necesario para la apertura de la comunicación entre Eclipse y Matlab, gracias a los cuales podemos indicar desde Matlab cuál es el ángulo que se está midiendo. Para que la comunicación sea bidireccional se ha habilitado el uso de un *engine* de Matlab, que encontraremos en el directorio de instalación de dicha aplicación, importándolo a nuestro proyecto Java. Gracias a esto, podremos ejecutar comandos de Matlab desde el software de Java, así como incluir variables en el Workspace creadas desde Eclipse.

En cuanto a la mesa giratoria, simplemente ha sido necesario abrir una comunicación con el controlador de la mesa giratoria desde Matlab tras haber instalado los drivers correspondientes y actualizados de la empresa *Keysight* para el controlador de *Maturo* mediante comandos GPIB-bus.

Este software de adquisición difícilmente puede considerarse como partes claramente diferenciadas debido a la mencionada comunicación constante entre los distintos entornos. Sin embargo, es conveniente para su comprensión discernir entre la funcionalidad del código de Matlab y la de Java, expuesto en los próximos epígrafes.

2.3. MATLAB

La manipulación de los datos obtenidos se realizará mediante la herramienta MATLAB, en su versión *R2019b*. Tras haber abierto la comunicación TCP IP desde Eclipse, el software realizado en esta plataforma será el primero que se ejecutará, dado que es el que nos permite controlar en qué ángulo estamos midiendo y nos proporciona información gráfica, a través de un plot, acerca de la RSSI media de los datos medidos por canal y antena.

El script principal de MATLAB, "*cliente.m*" (*Anexo II*), es el encargado de establecer las comunicaciones tanto con Eclipse como con la mesa giratoria a través del constructor "*tcpip*" que crea un objeto de tipo cliente o servidor y queda a la espera de instrucciones procedentes de Eclipse mediante el comando "*matlab.engine.shareEngine*".

A continuación, en este mismo script se procede a la creación de unas matrices de ceros correspondientes a las RSSIs medias de cada antena y cada canal que nos serán útiles para su posterior uso una vez adquiridos los datos. Así como la configuración de la posición inicial de la mesa giratoria.

A su vez, contiene un bucle de 1 a 191 (que será el número de ángulos que moveremos la mesa giratoria en la cámara anecoica, es decir, de -95° a 95° en pasos de 1°) dentro del cual se ejecuta una función "*conexion.m*" que introduce como parámetro el ángulo y el objeto de tipo cliente creado para la comunicación con Eclipse. Cada vez que se llama a esta función, en Eclipse se realiza la toma de los datos y la ordenación de estos en matrices que se pasarán al Workspace para cada ángulo. Estas matrices se llaman "*datos_medidaX*", donde X es el ángulo medido, y su estructura es la representada en la Figura 2.2.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	1	865.7000	-69.5000	0	1	866.3000	-69	0	1	866.9000	-69	0	1	867.5000	-70
2	1	865.7000	-69.5000	0	1	866.3000	-69	0	1	866.9000	-69.5000	0	1	867.5000	-69.5000
3	1	865.7000	-69.5000	0	1	866.3000	-69.5000	0	1	866.9000	-69	0	1	867.5000	-70.5000
4	1	865.7000	-69.5000	0	1	866.3000	-69.5000	0	1	866.9000	-69	0	1	867.5000	-70
5	1	865.7000	-70	0	1	866.3000	-68.5000	0	1	866.9000	-69.5000	0	1	867.5000	-70
6	1	865.7000	-70	0	1	866.3000	-69.5000	0	1	866.9000	-69.5000	0	1	867.5000	-70
7	1	865.7000	-70.5000	0	1	866.3000	-69.5000	0	1	866.9000	-69.5000	0	1	867.5000	-70
8	1	865.7000	-69.5000	0	1	866.3000	-69.5000	0	1	866.9000	-69.5000	0	1	867.5000	-70
9	2	865.7000	-60	0	2	866.3000	-59.5000	0	2	866.9000	-59.5000	0	2	867.5000	-59.5000
10	2	865.7000	-60	0	2	866.3000	-59.5000	0	2	866.9000	-59.5000	0	2	867.5000	-60
11	2	865.7000	-59.5000	0	2	866.3000	-59.5000	0	2	866.9000	-60	0	2	867.5000	-60
12	2	865.7000	-60	0	2	866.3000	-59.5000	0	2	866.9000	-60	0	2	867.5000	-59.5000
13	2	865.7000	-60	0	2	866.3000	-59.5000	0	2	866.9000	-60	0	2	867.5000	-60
14	2	865.7000	-60	0	2	866.3000	-59.5000	0	2	866.9000	-59.5000	0	2	867.5000	-59.5000
15	2	865.7000	-60	0	2	866.3000	-59.5000	0	2	866.9000	-60	0	2	867.5000	-60
16	2	865.7000	-60	0	2	866.3000	-59.5000	0	2	866.9000	-59.5000	0	2	867.5000	-60

Figura 2.2. Estructura de "datos_medidaX"

De forma independiente, se han creado dos funciones para realización de la media de los datos obtenidos, ya que obtendremos varias RSSI por canal y por puerto que habrá que promediar para la correcta representación, y otra para el control de la mesa giratoria, “*mediaRSSI_dBm.m*” y “*gira_mesa.m*”, respectivamente. Estas funciones serán llamadas desde eclipse gracias al comando “*feval*” proporcionado por el paquete “*com.mathworks.engine.MatlabEngine*”.

Gracias a la información recogida en estas matrices ya podríamos proceder al procesado para calcular las medias de las RSSI pertenecientes a un mismo canal, ángulo y antena gracias a una sencilla función. Esta función, llamada “*mediaRSSI_dBm*”, se invocará a continuación desde Eclipse y generará una serie de vectores de tamaño 1x191 que contienen la media de las RSSI recibidas para cada canal y antena, donde los 191 valores corresponden a un ángulo, como podremos observar en las figuras 2.3. y 2.4., correspondientes a la antena 1 y antena 2 para la frecuencia de 865,7 MHz.

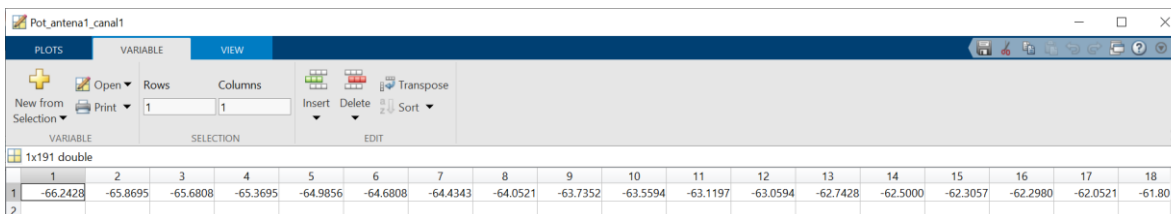


Figura 2.3. RSSI media para la antena 1 en el canal 1

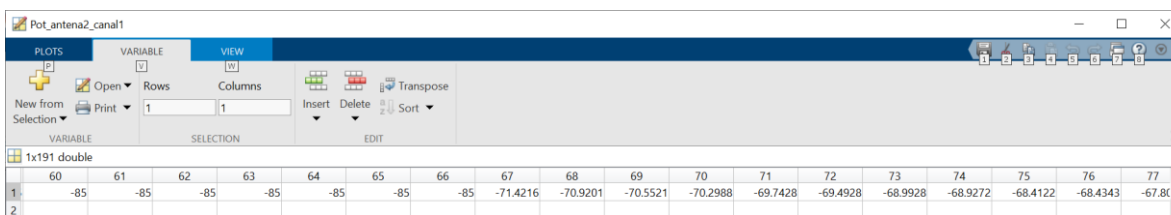


Figura 2.4. RSSI media para la antena 2 en el canal 1

Estas matrices se irán representando conforme vayamos haciendo la adquisición de datos. De esta forma podremos controlar si se está realizando correctamente o, por el contrario, ha habido algún dato incongruente. La representación obtenida puede verse en la figura 2.5. gracias a la cual se puede intuir la correcta ejecución del proceso.

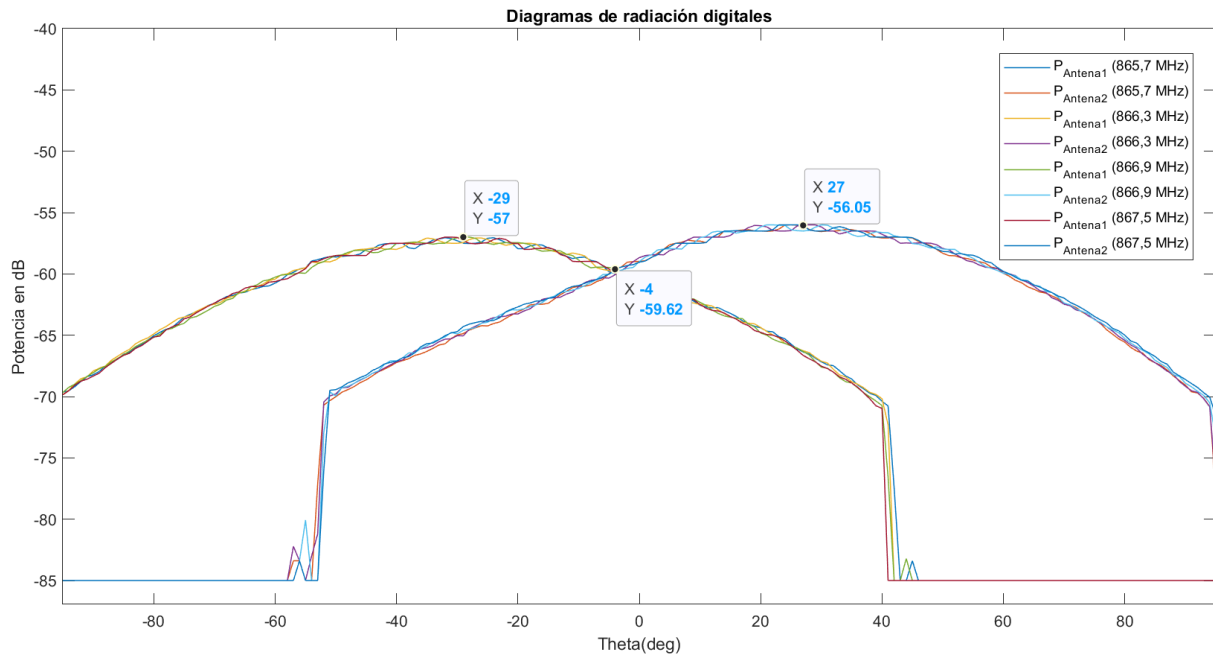


Figura 2.5. Representación de la RSSI media obtenida

2.4. Eclipse (Java)

El lenguaje de programación empleado para la configuración del *Reader Impinj R420* ha sido Java. Es por ello por lo que Eclipse ha sido el entorno de trabajo elegido, concretamente, la versión *Oxygen*.

Para una correcta ejecución del software es imprescindible importar en el entorno de Eclipse las librerías proporcionadas para el lenguaje Java por la empresa *Impinj*, llamadas *Octane SDK*, así como seguir los pasos que se indican en su página web¹.

La comunicación entre Eclipse y Matlab se realizará empleando un protocolo Cliente/Servidor mediante los sockets de Java. Este protocolo nos permite intercambiar información de forma simple entre ambas herramientas de desarrollo

El software está compuesto por 7 clases y una interfaz: *MainServidor*, *Servidor*, *Conexion*, *Config*, *Start*, *HelloOctaneSdk*, *TagReportListenerImplementation*, y *TagReportListener*, cuyas dependencias, métodos y atributos quedan expresadas en diagrama de clases UML de la figura 2.6.

¹ <https://support.impinj.com/hc/en-us/articles/219015497-Setting-up-Octane-SDK-Java-with-Eclipse>

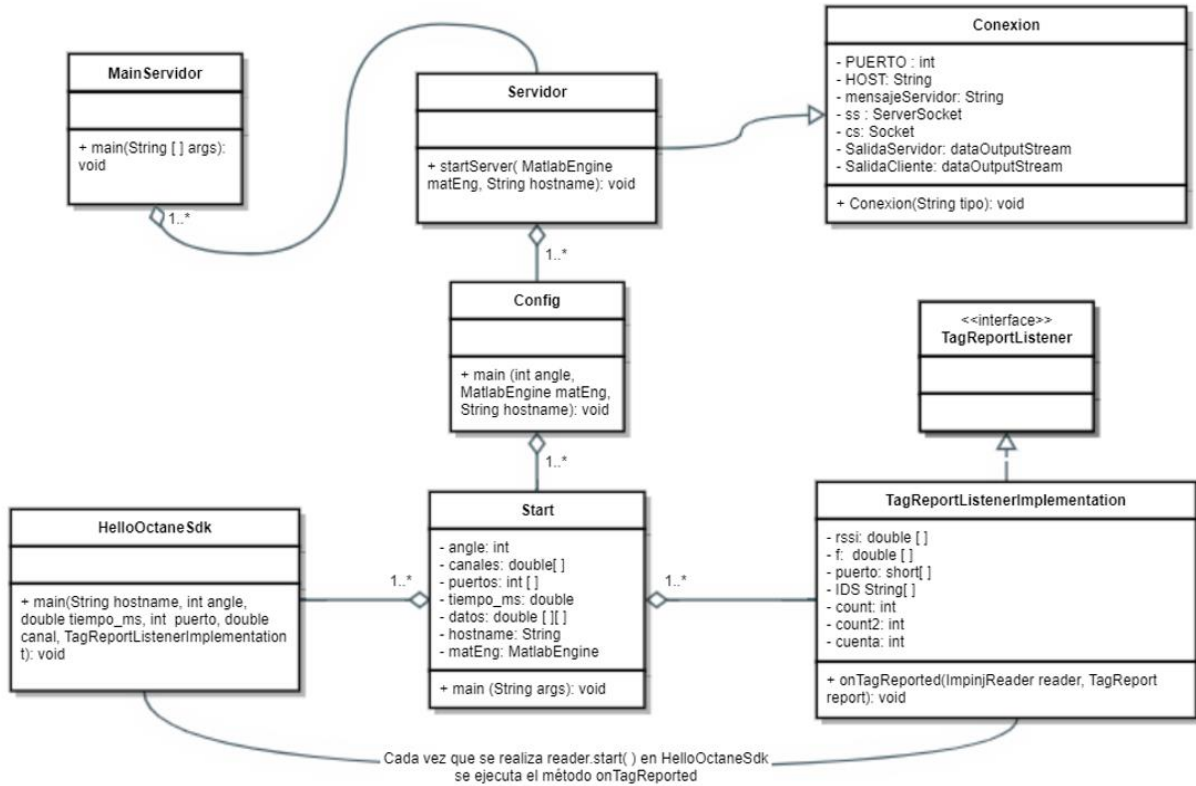


Figura 2.6. Diagrama de clases UML

El programa comenzará con la clase *MainServidor*. Previamente a la ejecución de la misma es necesario indicar en las configuraciones de Eclipse el hostname del *Reader Impinj R420*, en nuestro caso: “-Dhostname=speedwayr-10-5e-9e”. Esta clase es la encargada de realizar la correcta identificación del Reader y de abrir la conexión TCP/IP con MATLAB. Como se indica en el diagrama de clases (Figura 2.6.), otra de las funcionalidades es la de la creación de un objeto de la clase *Servidor*, al cual aplicaremos su método *startServer*. Una vez iniciada la conexión, se ejecutará el script de MATLAB llamado “*cliente.m*”, encargado de indicar a Eclipse el ángulo que se está midiendo, que se pasará como argumento al objeto de tipo *Config* que se creará en la actual clase.

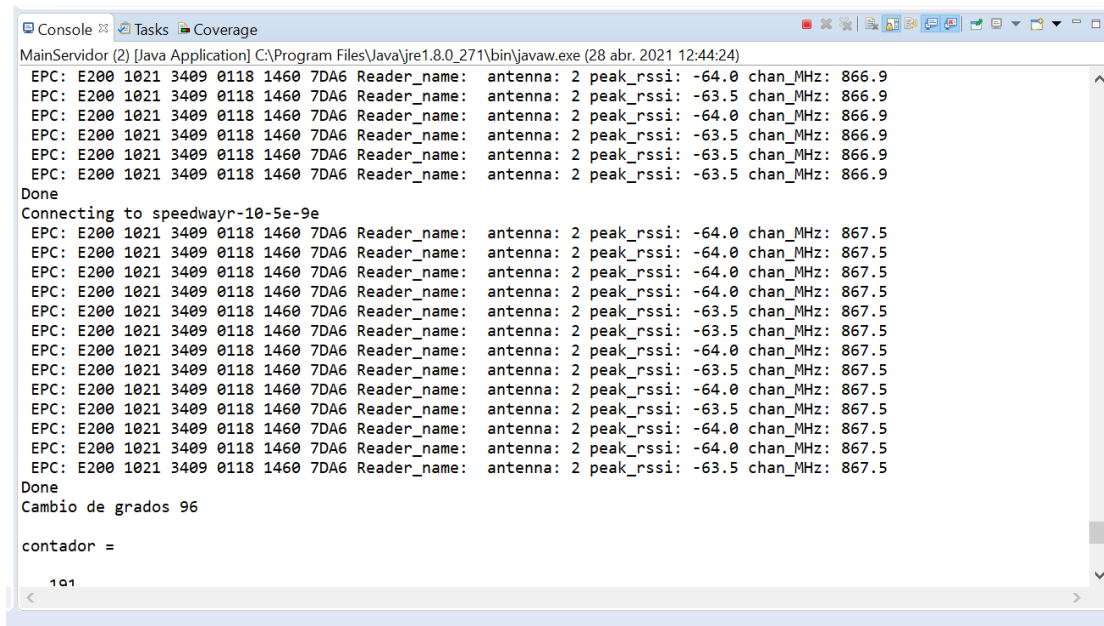
Config es la clase encargada de establecer los canales deseados, los puertos a medir, el hostname del reader y de comenzar el procedimiento de toma de datos llamando al método *main* de la clase *Start*.

Una vez realizada la llamada a *Start* se procede a la creación del objeto *HelloOctaneSdk* y *TagReportListenerImplementation*. Estos últimos se encargarán de configurar los parámetros del reader: los canales de frecuencias a los que va a transmitir, en este caso, la banda europea que incluye 865,7 MHz, 866,3 MHz, 866,9 MHz, y 867,5 MHz, la habilitación de transmisión en las 2 antenas de panel conectadas al Reader, y la duración de la transmisión (a 400 ms), que podremos variar en función de la cantidad de RSSI que queramos obtener. Cuando se ha

terminado de recibir información procedente del Reader, *Start* se encarga de almacenar los datos y de ordenarlos en matrices. Estas matrices son las mencionadas en el capítulo anterior: “*datos_medidaX*” que se pasarán directamente al Workspace de MATLAB gracias al método “*putVariable*” del paquete “*com.mathworks.engine.MatlabEngine*” aplicado sobre el objeto de comunicación de tipo *MatlabEngine*.

Una vez llegados a este punto ya hemos adquirido los datos correspondientes a un ángulo y, por tanto, conviene su representación. Esto será posible gracias al método “*feval*” que envía comandos a MATLAB con la sentencia de código que queremos ejecutar. En el caso que nos ocupa, se realizará un procesado para hallar la RSSI media gracias a la función de MATLAB “*media_RSSI.m*”, así como un plot de dicho valor. Por último, preparamos la siguiente adquisición rotando un grado la mesa giratoria.

Durante la adquisición de los datos, en la consola del entorno de trabajo se mostrará la información suministrada por el reader como se muestra en la figura 2.7, esto es, identificador del tag, antena por la que se recibe, RSSI en decibelios y frecuencia de trabajo.



```
Console
MainServidor (2) [Java Application] C:\Program Files\Java\jre1.8.0_271\bin\javaw.exe (28 abr. 2021 12:44:24)
EPC: E200 1021 3409 0118 1460 7DA6 Reader_name: antenna: 2 peak_rssi: -64.0 chan_MHz: 866.9
EPC: E200 1021 3409 0118 1460 7DA6 Reader_name: antenna: 2 peak_rssi: -63.5 chan_MHz: 866.9
EPC: E200 1021 3409 0118 1460 7DA6 Reader_name: antenna: 2 peak_rssi: -64.0 chan_MHz: 866.9
EPC: E200 1021 3409 0118 1460 7DA6 Reader_name: antenna: 2 peak_rssi: -63.5 chan_MHz: 866.9
EPC: E200 1021 3409 0118 1460 7DA6 Reader_name: antenna: 2 peak_rssi: -63.5 chan_MHz: 866.9
EPC: E200 1021 3409 0118 1460 7DA6 Reader_name: antenna: 2 peak_rssi: -63.5 chan_MHz: 866.9
Done
Connecting to speedwayr-10-5e-9e
EPC: E200 1021 3409 0118 1460 7DA6 Reader_name: antenna: 2 peak_rssi: -64.0 chan_MHz: 867.5
EPC: E200 1021 3409 0118 1460 7DA6 Reader_name: antenna: 2 peak_rssi: -64.0 chan_MHz: 867.5
EPC: E200 1021 3409 0118 1460 7DA6 Reader_name: antenna: 2 peak_rssi: -64.0 chan_MHz: 867.5
EPC: E200 1021 3409 0118 1460 7DA6 Reader_name: antenna: 2 peak_rssi: -64.0 chan_MHz: 867.5
EPC: E200 1021 3409 0118 1460 7DA6 Reader_name: antenna: 2 peak_rssi: -63.5 chan_MHz: 867.5
EPC: E200 1021 3409 0118 1460 7DA6 Reader_name: antenna: 2 peak_rssi: -63.5 chan_MHz: 867.5
EPC: E200 1021 3409 0118 1460 7DA6 Reader_name: antenna: 2 peak_rssi: -64.0 chan_MHz: 867.5
EPC: E200 1021 3409 0118 1460 7DA6 Reader_name: antenna: 2 peak_rssi: -63.5 chan_MHz: 867.5
EPC: E200 1021 3409 0118 1460 7DA6 Reader_name: antenna: 2 peak_rssi: -64.0 chan_MHz: 867.5
EPC: E200 1021 3409 0118 1460 7DA6 Reader_name: antenna: 2 peak_rssi: -64.0 chan_MHz: 867.5
EPC: E200 1021 3409 0118 1460 7DA6 Reader_name: antenna: 2 peak_rssi: -64.0 chan_MHz: 867.5
EPC: E200 1021 3409 0118 1460 7DA6 Reader_name: antenna: 2 peak_rssi: -64.0 chan_MHz: 867.5
EPC: E200 1021 3409 0118 1460 7DA6 Reader_name: antenna: 2 peak_rssi: -64.0 chan_MHz: 867.5
EPC: E200 1021 3409 0118 1460 7DA6 Reader_name: antenna: 2 peak_rssi: -63.5 chan_MHz: 867.5
Done
Cambio de grados 96

contador =

101
```

Figura 2.7. Consola de Eclipse durante la adquisición

Capítulo 3. Cámara Anecoica

La obtención de diagramas de radiación de las antenas tanto analógicos como digitales se realizó en una cámara anecoica. Una cámara anecoica es un recinto cerrado en el que se pueden simular las condiciones de espacio libre debido a su total recubrimiento con materiales aislantes que evitan las reflexiones acústicas y electromagnéticas y que la aísla totalmente de cualquier fuente interferente que se encuentre en el exterior de ésta.

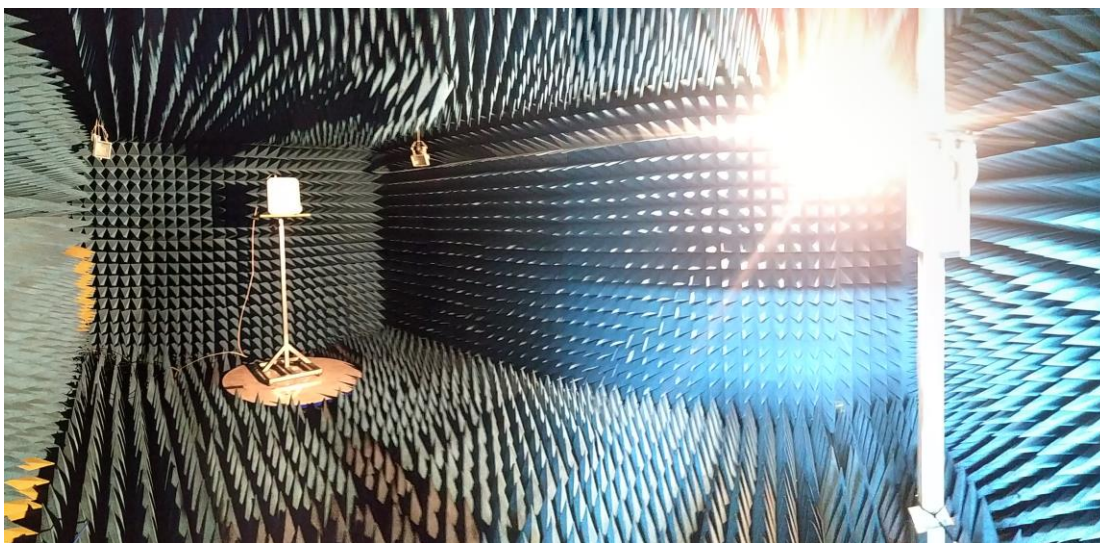


Figura 3.1. Cámara Anecoica

Con el fin de realizar los diagramas de radiación digitales de las antenas en configuración monopulso, se tuvo que realizar previamente la caracterización de un conjunto de antenas de panel por separado con el fin de encontrar aquellas cuyas características fueran más similares, concretamente, nos fijamos en su ancho de haz. A continuación, las antenas seleccionadas se posicionaron en configuración monopulso para la posterior medida analógica de los diagramas de radiación. Una vez obtenidos, se procedió a realizar las medidas digitales de los diagramas de radiación con ayuda del reader y el software implementado y expuesto en el capítulo 2 de este trabajo.

3.1. Instrumentación empleada

En esta sección se explicarán los distintos dispositivos empleados para la toma de datos analógicos y digitales.

3.1.1. Controlador y mesa giratoria

Para la correcta caracterización de las antenas por separado, así como la obtención de diagramas de radiación de las antenas de panel es necesaria una mesa giratoria que hace girar las antenas de forma automatizada en el rango de ángulos de interés. En este caso se empleó el plato de giro de la mesa giratoria *Compact Table CT 0800-P* de *Innco Systems*² que se muestra en la figura 3.1. Ésta es capaz de soportar un peso de hasta 75 kg y que puede ser controlada mediante un bus GPIB (IEEE 488).



Figura 3.2. Compact Table CT 0800-P

A su vez, este plato giratorio se controló mediante un controlador MCU (Figura 3.2) de la empresa *Maturo* mediante un bus GPIB-USB conectado directamente al ordenador que contiene el software de adquisición.

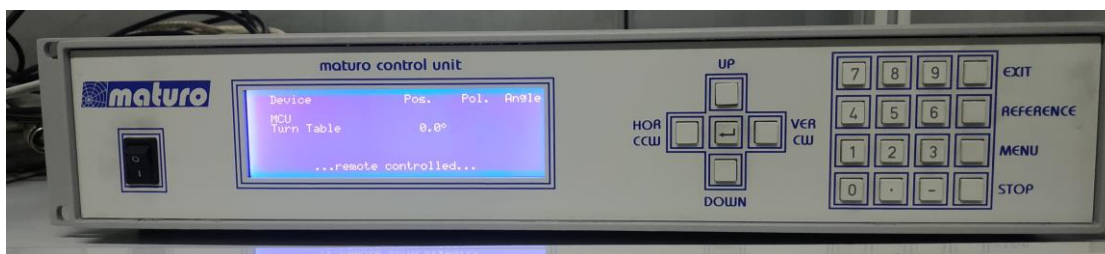


Figura 3.3. Controlador MCU Maturo

Para la correcta conexión con el ordenador portátil es necesaria la instalación del programa “*Keysight Connection Expert*”, desde el cuál podremos controlar la instrumentación conectada.

² <https://innco-systems.de/de/produkte/drehtische/ct0800-compact-table-for-freestanding-installation>

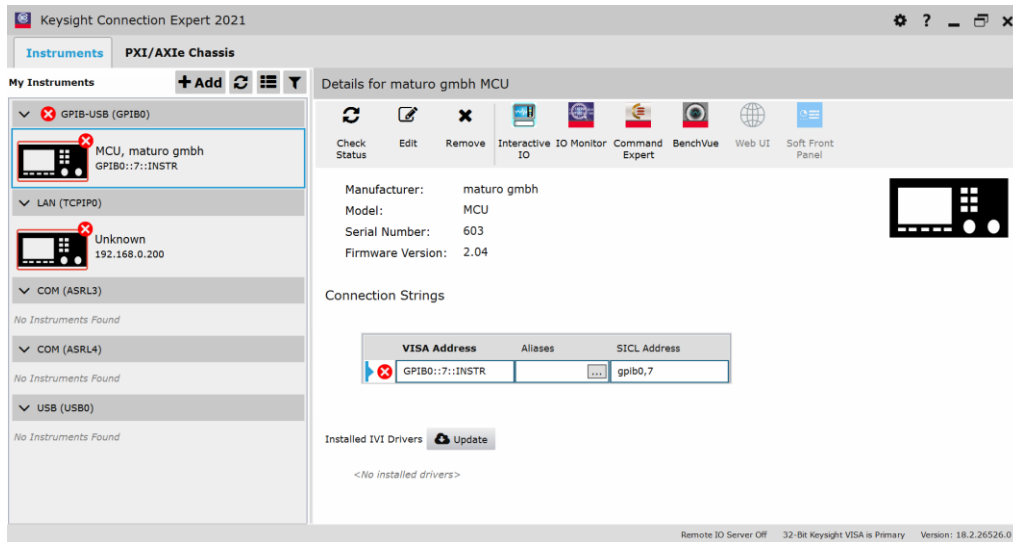


Figura 3.4. Interfaz de Keysight Connection Expert

3.1.2. Analizador de redes

Otro de los instrumentos empleados para la obtención de los diagramas analógicos fue un analizador de redes, en concreto, el analizador de *redes E571B ENA Series* de *Agilent Technologies* que funciona en la banda de 300 kHz hasta 8,5 GHz. Gracias a él podremos medir los parámetros S de las antenas de panel con ayuda de la interfaz de conexión mostrada en la Figura 3.4. de *Keysight Connection Expert*.

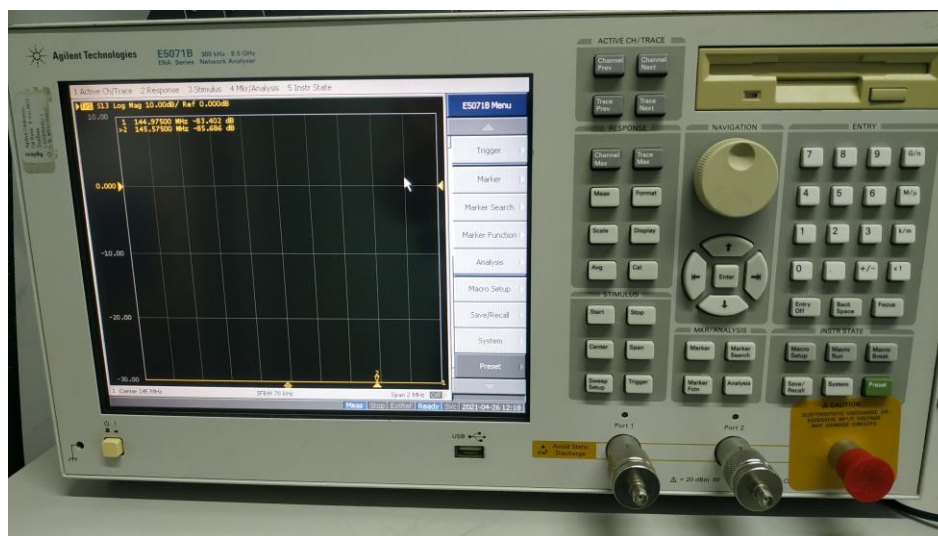


Figura 3.5. Analizador de Redes Agilent Technologies

3.1.3. Antenas de panel

Las antenas empleadas en este experimento son las antenas direccionales de panel de la empresa Laird. Éstas serán caracterizadas por separado y empleadas posteriormente para la localización angular de los tags.



Figura 3.6. Antena de panel Rain RFID de Laird

Las especificaciones de estas antenas se muestran en la siguiente tabla:

ESPECIFICACIONES ELÉCTRICAS	
Frecuencia de operación	865 – 868 MHz
Ganancia (dBic)	8,5
Impedancia nominal (Ω)	50
Máxima potencia (25°C) (W)	10
Polarización	Circular
Ancho de haz	70°

Tabla 3.1. Especificaciones Antena de panel

3.1.4. Antena bicónica

Emplear una antena bicónica fue fundamental para la correcta medición de los diagramas analógicos, tanto de las antenas de panel por separado como en configuración monopulso. La antena bicónica empleada fue de la empresa *Schwarzbeck Mess – Elektronik*, en concreto la *SBA 9113*.

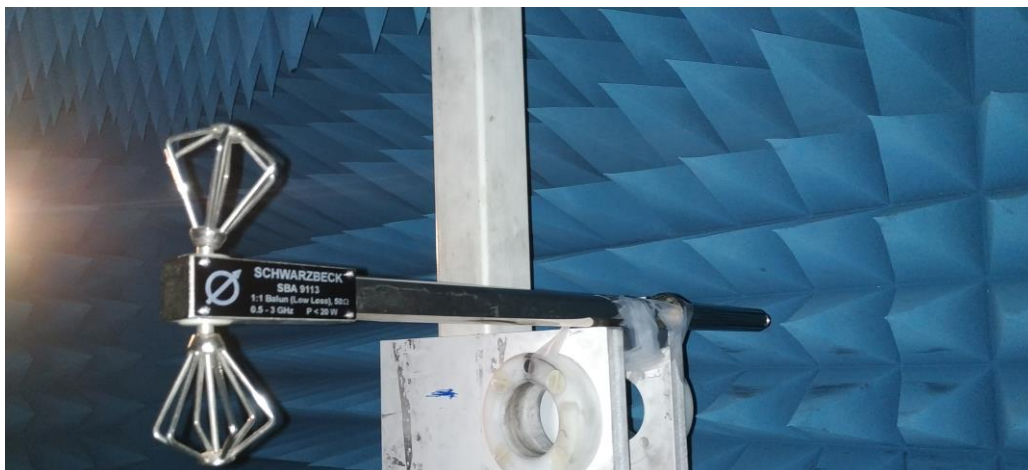


Figura 3.7. Antena bicónica SBA 9113

Algunas de sus características principales se ven reflejadas en la siguiente tabla.

ESPECIFICACIONES	
Frecuencia de operación	500 MHz – 3 GHz
Ganancia Isotrópica (dBi)	-1,5...+1,5
Impedancia nominal (Ω)	50
Máxima potencia de entrada (W)	20
Polarización	Lineal
Ancho de haz (Plano E)	80° - 50°
Ganancia en la frecuencia de operación (dBi)	0,59

Tabla 3.2. Especificaciones técnicas de la Antena bicónica

3.1.5. Reader Impinj R420

El instrumento fundamental para la toma de medidas digitales es el lector *RFID UHF Impinj Speedway Revolution R420*. Como se ha explicado en el capítulo 2 de este trabajo, este lector será configurado en la banda de frecuencias de interés para que sea capaz de transmitir las ráfagas deseadas por cada antena de panel, siendo, a su vez, capaz de detectar el nivel de potencia RSSI procedente de vuelta de los tags con la información pertinente al mismo.



Figura 3.8. Reader Impinj Speedway Revolution R420

En la tabla 3.3. se muestran algunas de las características de su ficha técnica.

FICHA TÉCNICA	
Frecuencia de trabajo (MHz)	865 – 868 (ETSI) / 902 -928 (FCC)
Alimentación	24 Vdc / 2,1 A
Grado de protección	IP 52
Entradas y Salidas	4 puertos R-TNC hembra
Potencia de transmisión	+10,0 / 32,5 dBm
Interfaz de comunicación	RS-232, Ethernet, USB
Protocolo de Tags	GS1/EPCglobal UHF Class 1 Gen 2 (ISO 18000-6C)
Temperatura de funcionamiento	-20°C a +50°C
Dimensiones (mm)	190,0 x 175,3 x 30,5
Peso (kg)	0,7

Tabla 3.3. Ficha técnica del Reader Impinj R420

3.1.6. Tags

Los tags empleados en las medidas digitales y en los posteriores experimentos de localización se tratan de tags pasivos de la marca *Omni-ID*.



Figura 3.9. Tag pasivo Omni-ID

Estos tags funcionan en el rango de frecuencias que incluye tanto la banda europea (866 – 868 MHz), que será la banda de interés, como la banda americana (902 – 928 MHz), con un rango de detección mayor de 3,5 metros. Al ser tags pasivos, no necesitan ningún tipo de fuente de alimentación, basta con que éstos reciban una señal desde el lector para generar una respuesta.

3.2. Obtención de los diagramas analógicos

3.2.1. Caracterización de las antenas por separado

Como se indicó en el punto 3.1. de este trabajo, el primer paso para la correcta caracterización de las antenas en configuración monopulso fue buscar aquellas antenas de panel cuyos anchos de haz fueran más similares. Para ello, se colocaron cada una de las antenas a caracterizar en un soporte sobre la mesa giratoria a 3,1 metros de distancia de una antena bicónica, ambas a la misma altura sobre el suelo, dentro de la cámara anecoica. Éstas se conectaban mediante un cable coaxial al analizador de redes. Dicho analizador se configuró para tomar la medida de los parámetros S de la antena de panel y a su vez éste se conectó vía Ethernet a un ordenador portátil que contenía el software de adquisición de los datos analógicos y se encargaba de hacer girar remotamente la mesa giratoria (mediante conexión GPIB-USB con el controlador de *Maturó*). En la siguiente figura podemos ver la disposición dentro de la cámara anecoica.

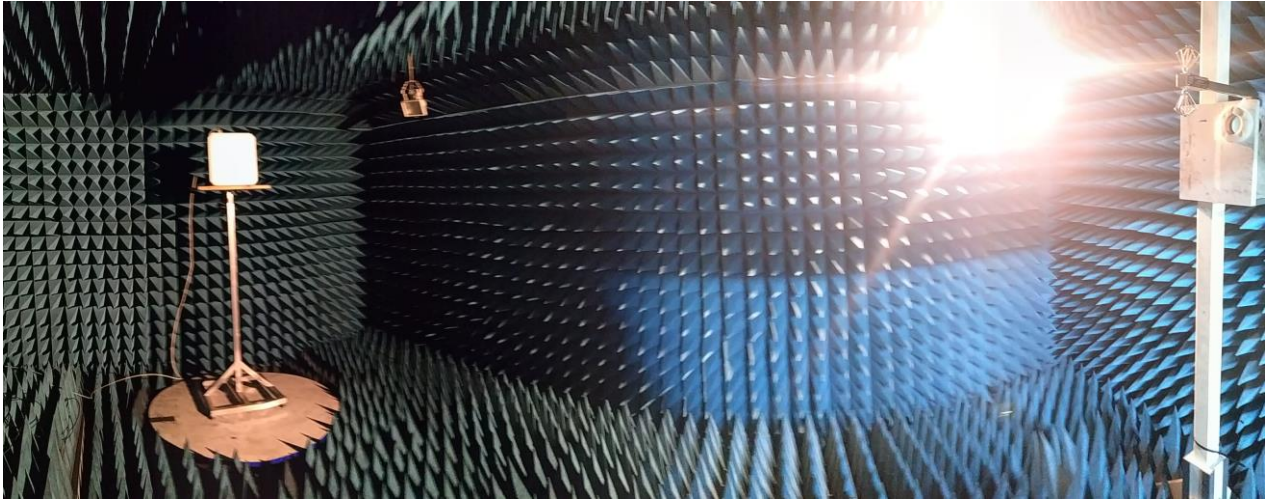


Figura 3.10. Disposición para medida analógica de una antena de panel

Las frecuencias a la que se realizó la obtención de parámetros S fue 856,7 MHz, 866,3 MHz, 866,9 MHz y, finalmente, 867,5 MHz, que son las frecuencias de la banda europea que emplearemos. Además, este proceso se repitió cuatro veces, para 4 antenas de panel diferentes y los resultados de sus diagramas de radiación analógicos fueron los siguientes:

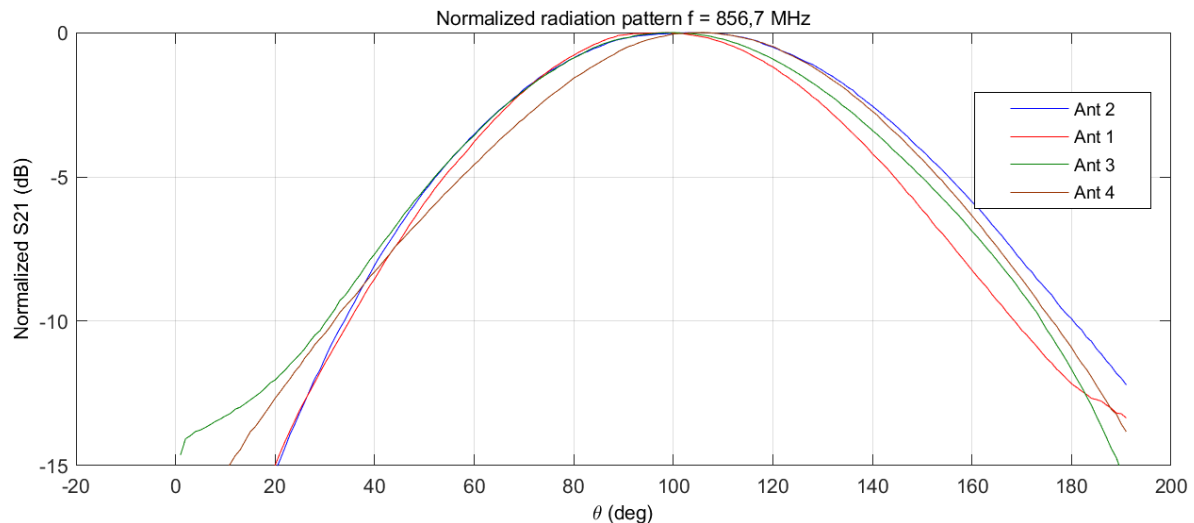


Figura 3.11. Diagramas de radiación analógicos de las 4 antenas de panel en el canal 1

Si representamos los diagramas de radiación de las antenas por separado en sus 4 canales de frecuencia, podremos obtener el ancho de haz de cada una de ellas, de tal forma que se puede determinar cuáles son más aptas para disponerse en configuración monopulso.

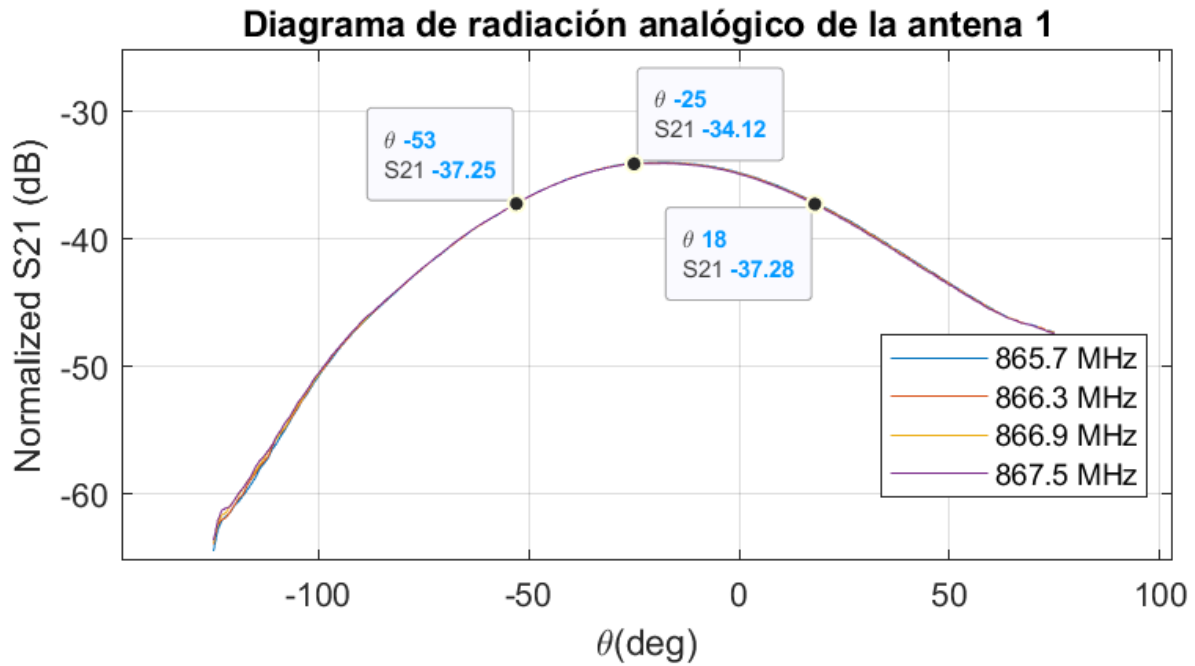


Figura 3.12. Diagramas de radiación analógicos de la antena 1

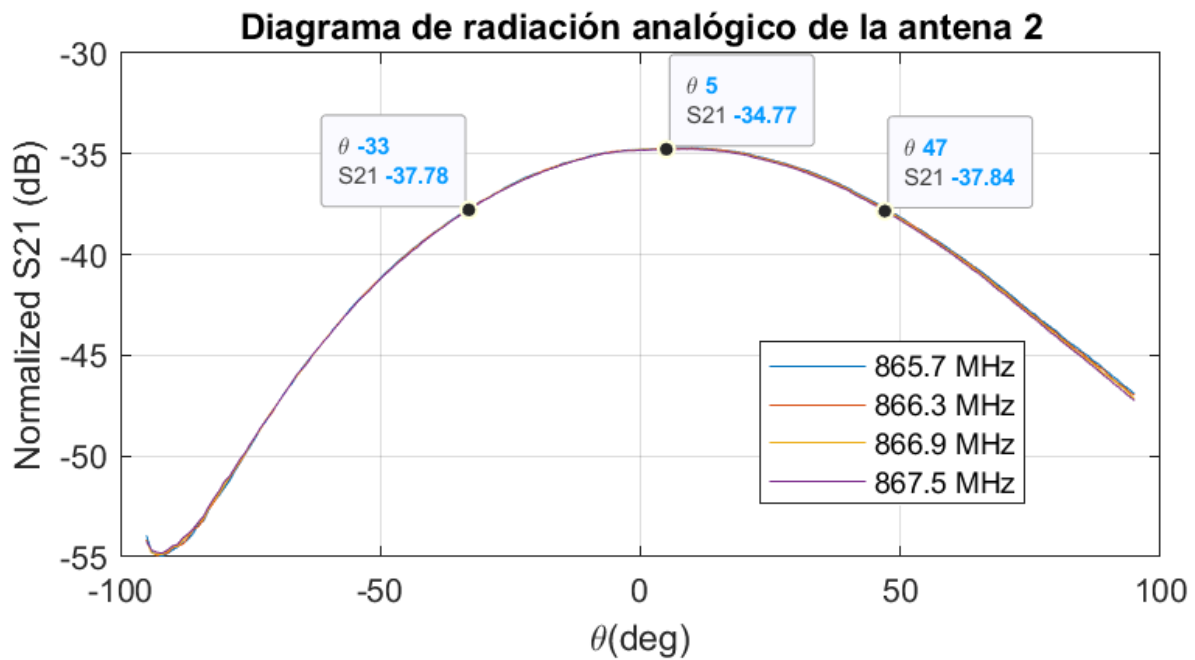


Figura 3.13. Diagramas de radiación analógicos de la antena 2

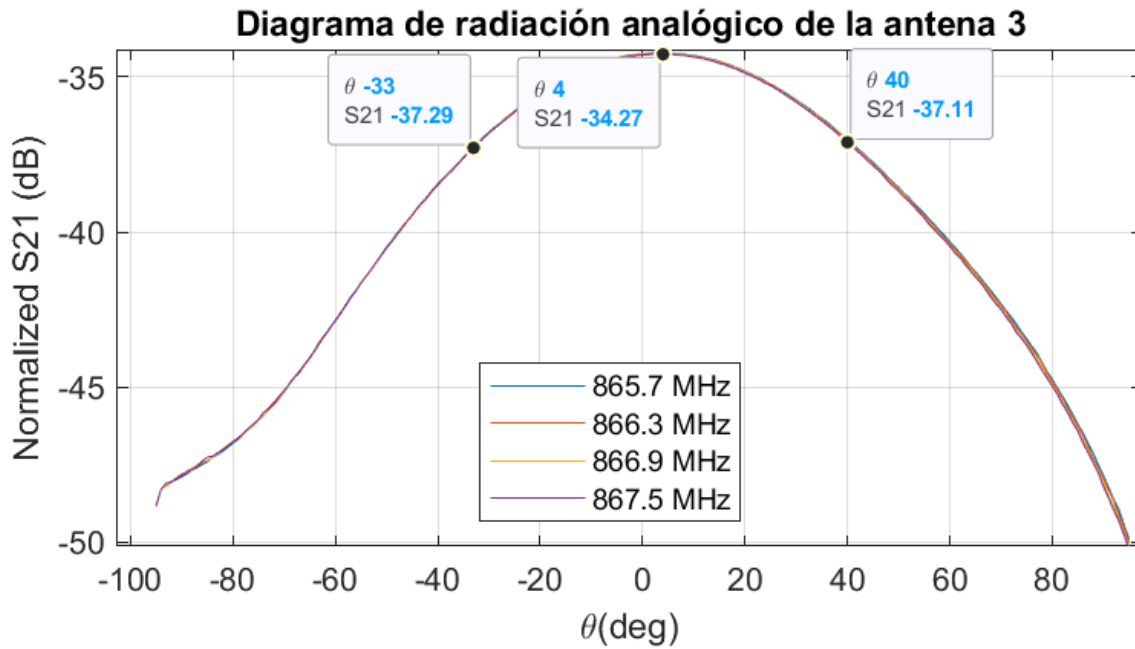


Figura 3.14. Diagramas de radiación analógicos de la antena 3

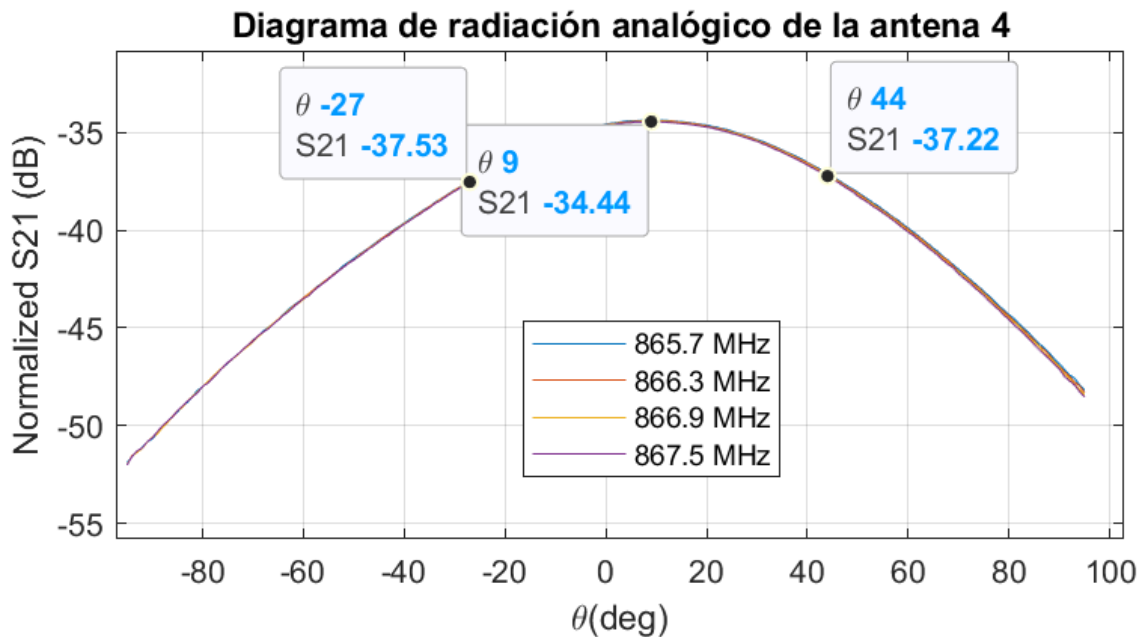


Figura 3.15. Diagramas de radiación analógicos de la antena 4

Los anchos de haz para cada una de las antenas se midieron en función de a qué ángulo su diagrama de radiación caía 3 dB con respecto a su ganancia máxima, de tal forma que los resultados fueron los siguientes.

Para la antena 1: $\Delta\theta_{-3dB} = 18 - (-53) = 71^\circ$

Para la antena 2: $\Delta\theta_{-3dB} = 47 - (-33) = 80^\circ$

Para la antena 3: $\Delta\theta_{-3dB} = 40 - (-33) = 73^\circ$

Para la antena 4: $\Delta\theta_{-3dB} = 44 - (-27) = 71^\circ$

A la vista de los resultados, podemos concluir que las antenas que más se parecen son las marcadas como antena 1 y antena 4. Estas serán las elegidas para disponerse en configuración monopulso y hallar los diagramas de radiación conjuntos.

3.2.2. Caracterización de las antenas en configuración monopulso

El objetivo de disponer las antenas en configuración monopulso es que sus diagramas de radiación queden dispuestos de tal forma que la caída de ambos a -3dB de potencia con respecto a su ganancia máxima corte a 0 grados en theta.

Para disponer las antenas en configuración monopulso se hizo uso de un soporte para las mismas capaz de disponerlas en los grados de interés. A la vista de que el ancho de haz de cada una de ellas es de 71° , el tilt lateral de las antenas se deduce de aproximadamente 35° . Sin, embargo, en virtud de pruebas realizadas para obtener los diagramas de radiación conjuntos, se decidió que la mejor disposición para que ambos diagramas cortaran en -3dB fue de 30° . Podemos verlo representado en la figura 3.16.



Figura 3.16. Antenas 1 y 4 en configuración monopulso

Una vez colocadas las antenas en la cámara anecoica, la disposición del resto de elementos queda reflejada en la siguiente ilustración. Análogamente al paso previo, se empleó una antena bicónica de referencia para la medida de los parámetros S.

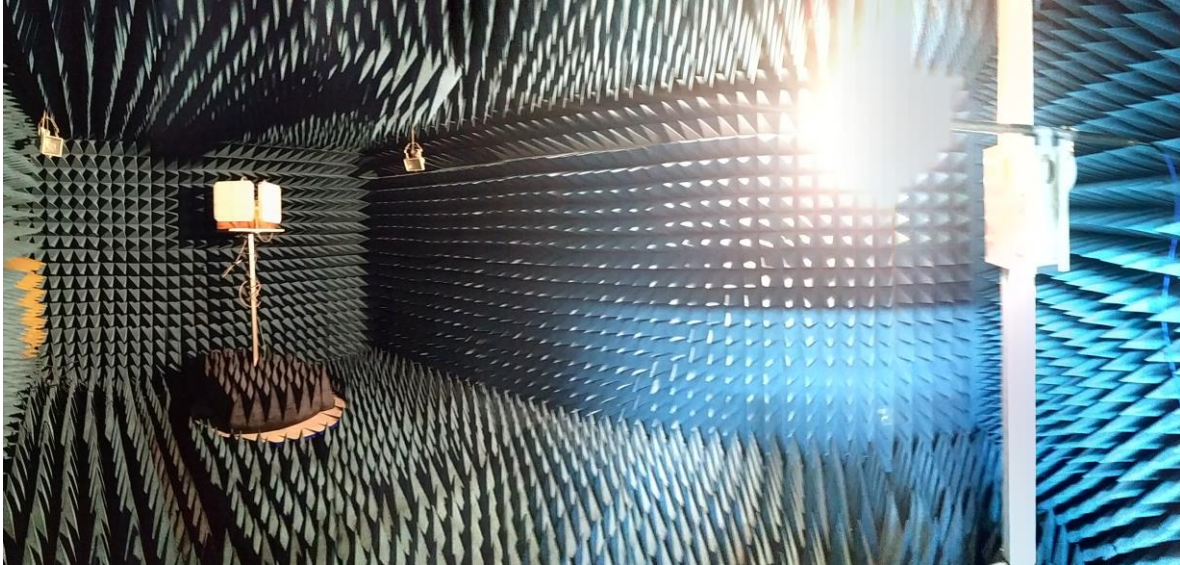


Figura 3.17. Disposición de los elementos para la obtención de los diagramas en configuración monopulso

Al igual que para las antenas por separado, se tuvo que medir el diagrama de radiación de cada antena individualmente girando el soporte de la mesa giratoria desde los -90° hasta los 90° . Puede parecer un proceso similar al anterior, pero en este caso los diagramas pueden variar debido a la influencia de la antena que no se está midiendo con respecto a la misma y su cambio de disposición, ya que en esta ocasión el máximo de ganancia de cada una de ellas apunta a un ángulo diferente. Los resultados obtenidos para los diagramas de radiación conjuntos quedan dispuestos en la figura 3. 18. y nos serán útiles para compararlos con los resultados obtenidos en las medidas digitales y comprobar que la adquisición se ha realizado correctamente.

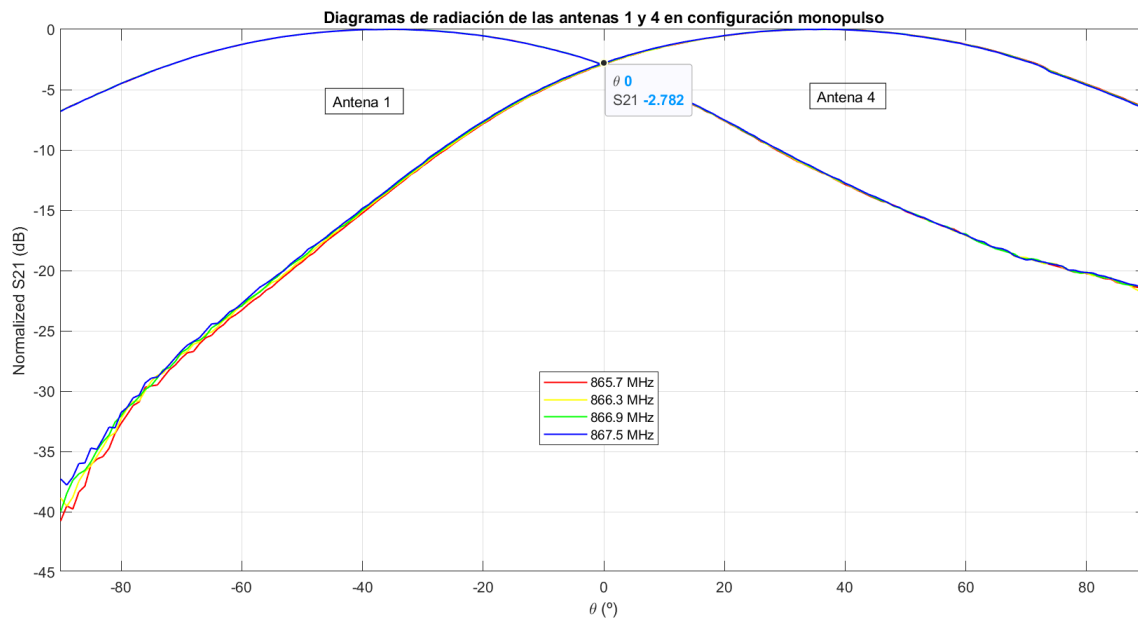


Figura 3.18. Diagramas analógicos en configuración monopulso

Como se puede observar, ambos diagramas cortan a 0 grados y justo cuando la potencia máxima decae aproximadamente -3dB (-2,782 dB concretamente). Los diagramas son prácticamente iguales a pesar de haber sido tomados en diferentes frecuencias debido a la proximidad de las mismas. Esto también se puede observar en la representación polar de los diagramas de radiación conjuntos.

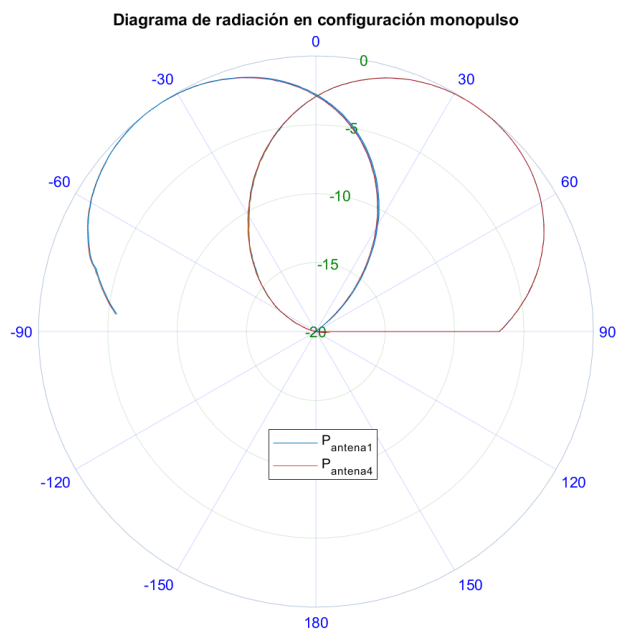


Figura 3.19. Diagramas de radiación analógicos en forma polar

3.3. Obtención de los diagramas digitales

Los componentes que integran el sistema de localización se distribuyen tanto dentro de la cámara anecoica como fuera de ella. Dentro de la cámara anecoica se dispusieron las antenas en configuración monopulso sobre la mesa giratoria frente al tag pasivo RFID a una distancia de 3,1 metros. A su vez se incluyó dentro de la misma el lector, ya que se conecta directamente a las antenas y a una toma de corriente que se encuentra en el interior de la cámara. Fuera de la cámara se encontraba el controlador de la mesa giratoria y el ordenador con el software de adquisición de datos.

Las conexiones entre la instrumentación del interior y el exterior de la cámara anecoica fueron posibles gracias a los pequeños accesos que hay en una de las paredes de la misma, por donde tuvo que pasar el cable ethernet que conecta el lector con el ordenador portátil y el cable de conexión GPIB que conecta la mesa giratoria y el controlador de la misma.

Podemos ver reflejadas estas conexiones en el esquema de la figura 3.20.

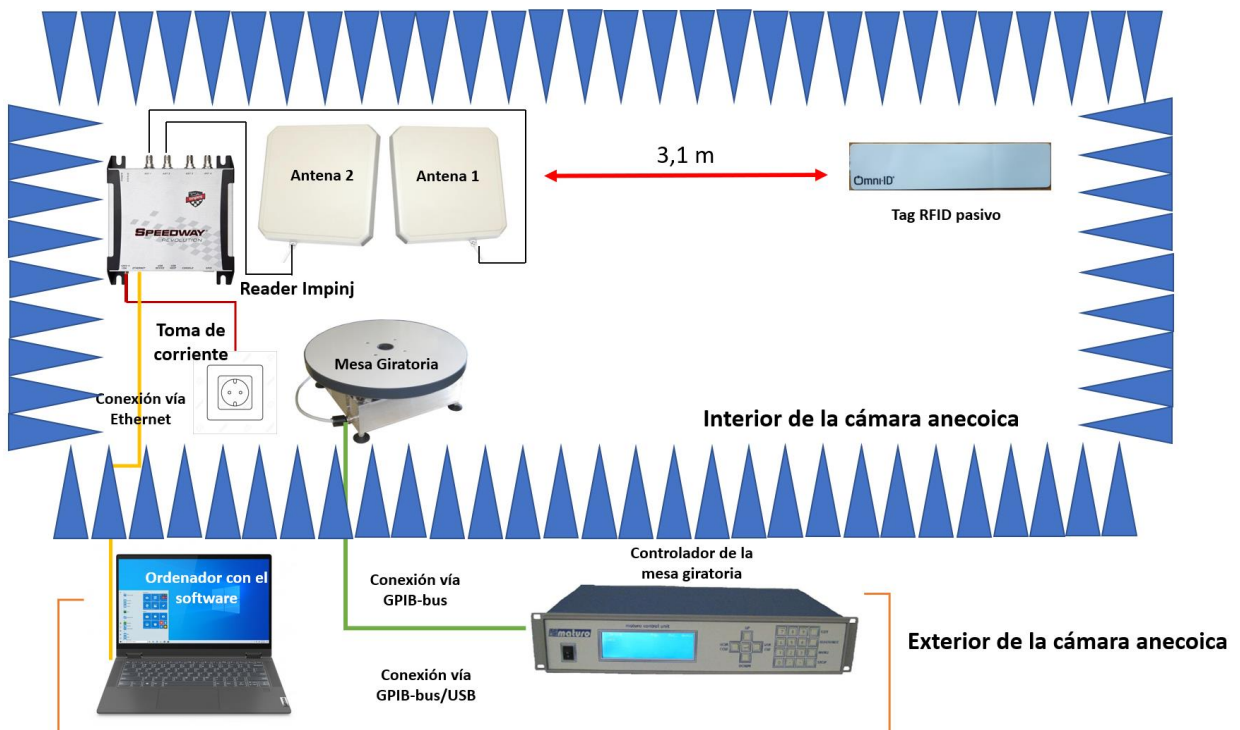


Figura 3.20. Disposición de los instrumentos para la toma de datos digitales

El interior de la cámara anecoica durante la adquisición se puede ver en la figura 3.21.



Figura 3.21. Interior de la cámara anecoica

En este punto de la adquisición, solamente fue necesario realizar el diagrama digital de las antenas en configuración monopulso directamente. A partir de ahora, y a lo largo del resto del trabajo, no se hablará de antena 1 y antena 4 sino de antena 1 y antena 2, siendo la antena 1 la que apunta hacia ángulos negativos y la antena 2 la que apunta a ángulos positivos considerando el ángulo 0 la posición en la que se encuentra el tag.

Los diagramas digitales obtenidos para un tilt lateral igual al empleado en las medidas analógicas fueron los mostrados en la figura 3.22.

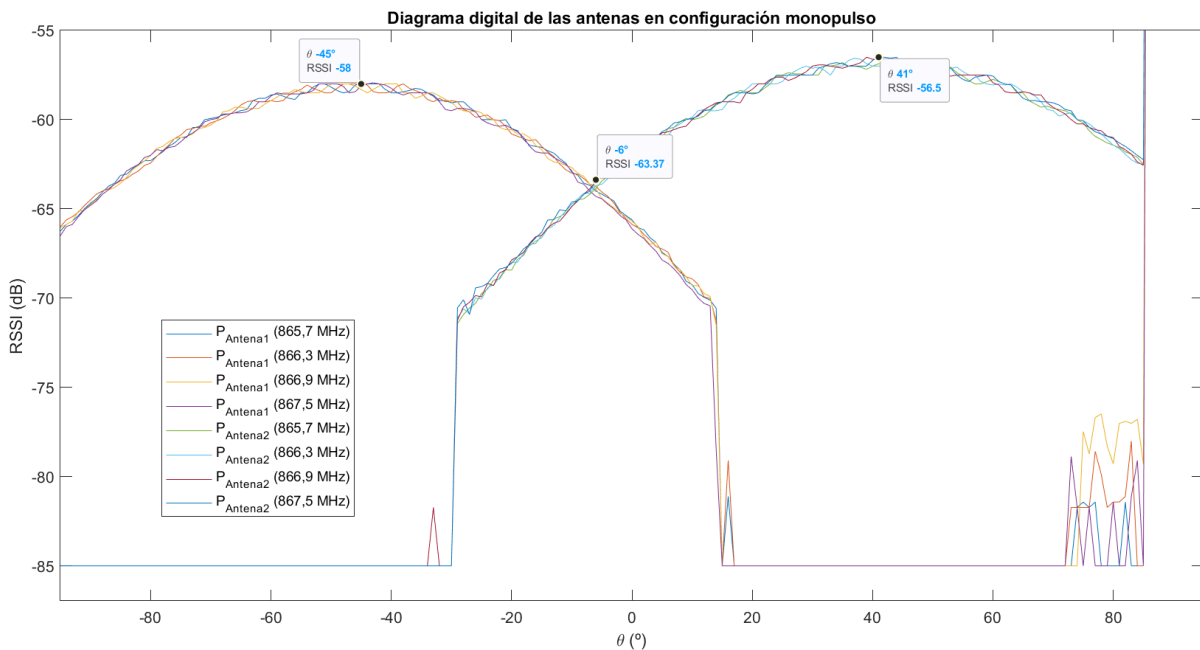


Figura 3. 22. Diagramas de radiación digitales para un tilt lateral de 30°

Más allá de que el corte entre los diagramas de radiación no se produce a 0° , cosa que fácilmente se pudo solventar, es interesante el hecho de que el corte no se produce a -3 dB sino a alrededor de unos -6 dB. En analógico, los diagramas de radiación se obtuvieron directamente al medir desde las antenas de panel hasta una antena de referencia de tipo bicónica que se encontraba a una distancia de $3,1$ m de la anterior, esto es, los diagramas se miden en una dirección. En esta ocasión, el diagrama de radiación digital se mide empleando las antenas de panel y observando la información de RSSI que les llega de vuelta tras incidir sobre un objeto reflectante, que en este caso es el tag pasivo. Esto hace que el sistema de funcionamiento sea similar al de un radar biestático, lo cual implica que la fuente emisora y receptora estén en la misma posición y se deban considerar tanto el camino de ida como el de vuelta (lo cual implica que se recorren $6,2$ metros hasta obtener la información de RSSI). Por esta razón, empleando un tilt lateral igual que para la obtención de los diagramas analógicos, se ha obtenido en la perpendicular un nivel de RSSI que es la mitad de lo esperado, ya que al aplicar 2 veces el filtro espacial hemos obtenido la ganancia multiplicada por un factor de 2 en escala logarítmica, lo cual también implica un cambio en el ancho de haz de los diagramas digitales y, por tanto, el Field Of View.

Tras realizar varias pruebas se concluyó que el tilt lateral de las antenas en configuración monopulso debía modificarse entorno a los 15° para que los diagramas de radiación digitales de ambas antenas cortaran a -3 dB en 0° , quedando en la posición que se observa en la figura 3.23.



Figura 3.23. Antenas en configuración monopulso para las medidas digitales

Las herramientas empleadas para la toma de datos se han expuesto previamente en el capítulo 2 de este trabajo. Una vez puesto en marcha todo el sistema de localización se obtuvo en el entorno de MATLAB el conjunto de datos correspondientes a 20 RSSIs por antena y por

canal referentes a los distintos grados desde -95° hasta 95° . Tras realizar la media de dichas potencias para cada antena y canal, pudimos obtener el siguiente diagrama de radiación.

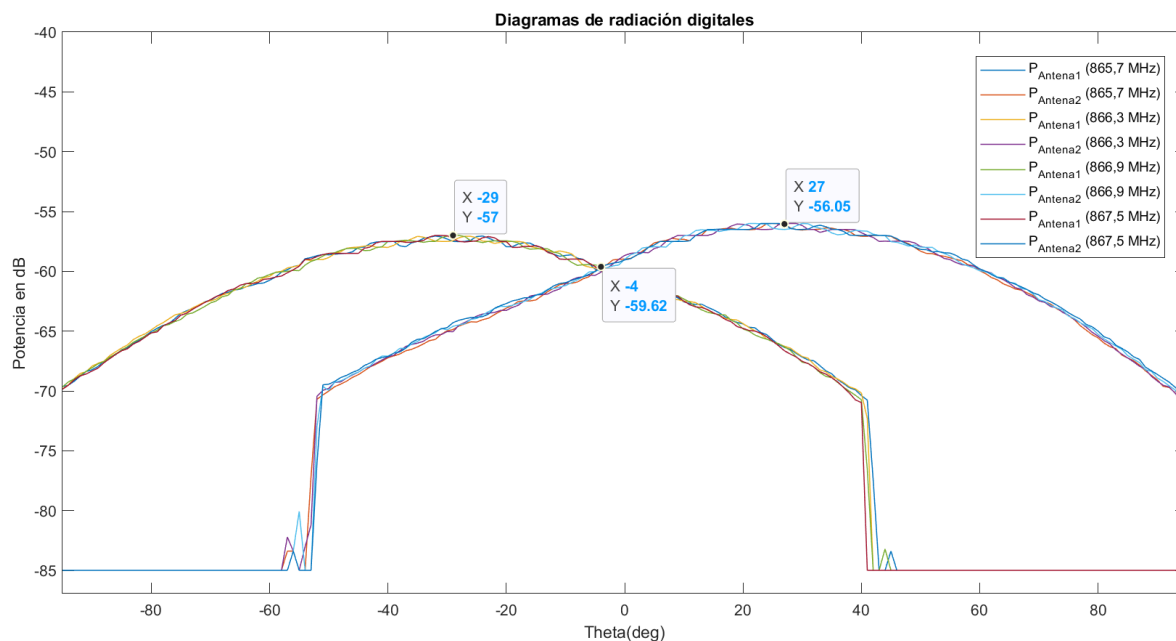


Figura 3.24. Diagramas de radiación digitales

Como podemos ver en la figura 3.24, el corte de los diagramas de radiación se produce en -4° , sin embargo, los niveles de RSSI máxima varían 1 dB, siendo menor para la antena 1 que para la antena 2, de forma que, al realizar posteriormente el procesado para igualar esta diferencia de ganancia, el corte de los diagramas de las antenas se centrará más en 0° . En la representación normalizada de los diagramas de radiación se puede apreciar con más detalle. Se puede observar, que la RSSI mínima recibida se corresponde con la sensibilidad del reader, correspondiente a -85 dB.

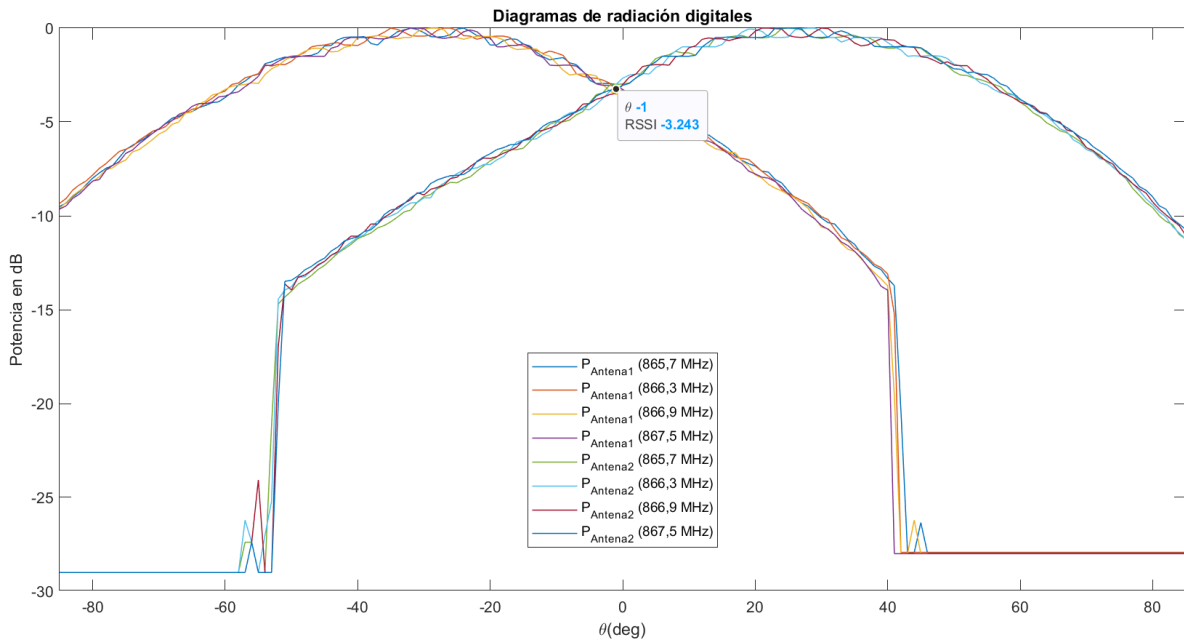


Figura 3.25. Diagramas de radiación digitales normalizados

También cabe destacar que se ha aumentado más de lo previsto el FoV debido a que existe información de RSSI de la antena 2 desde -51° y de la antena 1 hasta, aproximadamente, 40° . Esto nos deja un margen de estimación de la DoA de los tags de 91° , probablemente debido a que las antenas tienen un único lóbulo. No obstante, el cálculo exacto del Field Of View se detallará en el capítulo del procesado para la localización angular (Capítulo 4).

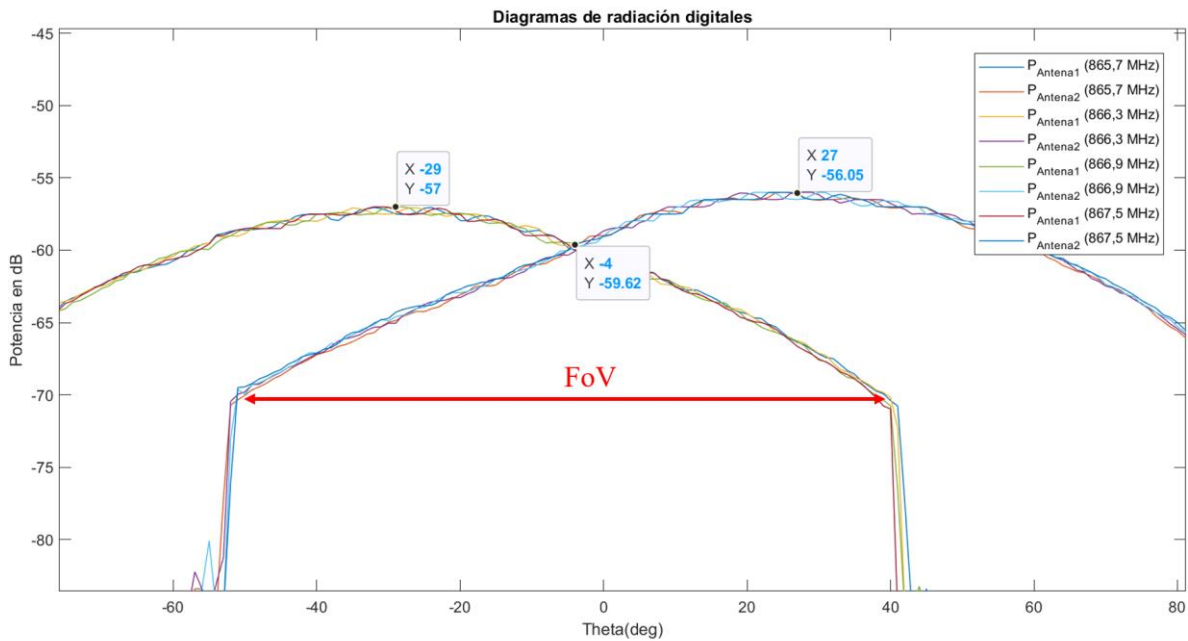


Figura 3.26. Field Of View obtenido

Capítulo 4. Aplicación de técnicas monopulso y procesado para localización angular

Tras la obtención de los diagramas digitales de las antenas en configuración monopulso, se pudo proceder a la aplicación de técnicas monopulso para estimar la localización angular de los tags.

4.1. Función monopulso

Como ya se ha expuesto previamente en este trabajo, el sistema monopulso empleado está basado en la comparación de la amplitud de la señal recibida. Queda de manifiesto que se han obtenido patrones de radiación de las dos antenas que quedan superpuestos y que son totalmente análogos en el rango de frecuencias de trabajo. No obstante, para cada una de las frecuencias se construirá una función monopulso diferente.

La función monopulso se define empleando la amplitud detectada por cada antena en valores lineales, ésta es el resultado de dividir un array compuesto por la diferencia de los diagramas de radiación de la antena 2 (que apunta hacia ángulos positivos) y la antena 1 (que apunta hacia ángulos negativos, conocido como patrón diferencia, y un array de valores lineales resultado de la suma de los diagramas de radiación de ambas antenas, conocido como patrón suma.

$$\Psi_D(\theta) = \frac{\Delta_D(\theta)}{\Sigma_D(\theta)} = \frac{RSSI_2(\theta) - K_D \cdot RSSI_1(\theta)}{RSSI_2(\theta) + K_D \cdot RSSI_1(\theta)} \quad (1)$$

El término K_D hace referencia al coeficiente de corrección digital que se extrae fácilmente como resultado de los niveles de RSSI medidos para cada antena en la dirección perpendicular, esto es, $\theta = 0^\circ$.

$$K_D = \frac{RSSI_2(\theta = 0^\circ)}{RSSI_1(\theta = 0^\circ)} \rightarrow K_D(dB) = RSSI_2(\theta = 0^\circ) - RSSI_1(\theta = 0^\circ) \quad (2)$$

El valor del coeficiente de corrección digital, si nos fijamos en los diagramas de radiación digitales sin normalizar (figura 4.1) podemos concluir que se trata de, aproximadamente, $K_D=1$ dB.

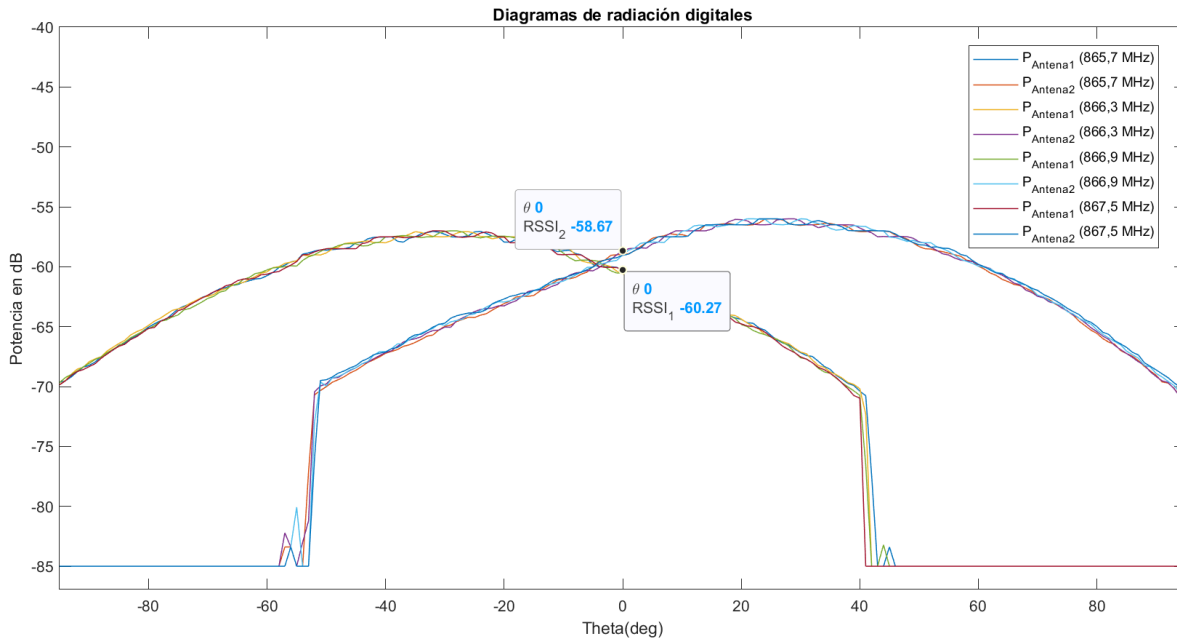


Figura 4.1. Obtención del coeficiente de corrección digital

El procesado para realizar la función monopulso, dará como resultado una curva aproximadamente lineal en un rango de theta para el cual se define el FoV y en el cual, la función monopulso toma valores entre -1 y 1, correspondiéndose este rango angular con la zona en la que se puede asociar de forma biunívoca el valor de la función monopulso con un ángulo de llegada único de un tag. Como se ha mencionado anteriormente, tendremos 4 funciones monopulso correspondientes a los diferentes canales en los que trabajamos. Podemos ver las funciones monopulso en la siguiente figura.

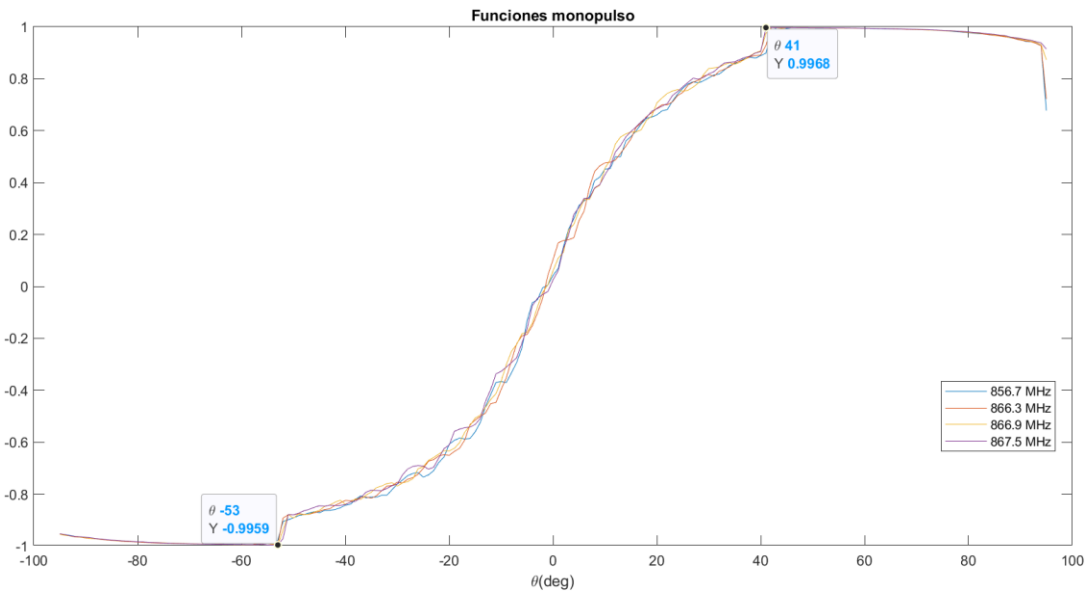


Figura 4.2. Funciones monopulso

Se observa que el FoV comprende el rango angular desde los -53° hasta los 41° .

4.2. Valor monopolso y error monopolso

El valor monopolso tiene una definición análoga a la función monopolso, consiste en un valor numérico obtenido a partir de dos medidas de amplitud procedentes de, en este caso, un tag por la antena 2 y antena 1, resultado de la división entre la diferencia entre ambos y la suma de los mismos. Hay que considerar también el factor de corrección para el cálculo de este valor al igual que para la función monopolso, en nuestro caso, aplicado a la RSSI de la antena 1.

Cuando obtenemos dos medidas de RSSI cualesquiera procedentes de cada antena y siempre y cuando se encuentren dentro del Field of View, podremos obtener su valor monopolso. Este valor nos permitirá compararlo con la función monopolso correspondiente a la frecuencia empleada para su obtención, de tal forma que dentro de la función monopolso encontraremos un único valor que se corresponde con dicho valor monopolso, y que irá asociado a un único valor de theta. Ese valor de theta nos da lo que se conoce como Direction of Arrival.

$$\Psi_{Dato} = \frac{\Delta_D}{\Sigma_D} = \frac{RSSI_2 - K_D \cdot RSSI_1}{RSSI_2 + K_D \cdot RSSI_1} \quad (3)$$

Una vez conocido el valor monopolso, podemos proceder a estimar el error monopolso. El error monopolso se trata de una función no lineal definida para todo el rango de theta, que nos muestra la diferencia entre la función monopolso y el valor monopolso obtenido para un par de RSSIs en valor absoluto. Esta función presentará un mínimo en el valor de theta que se corresponde con el ángulo en el que se encuentra el tag.

$$E_D(\theta) = |\Psi_D(\theta) - \Psi_{Dato}| \quad (4)$$

Conviene definir, llegados a este punto, el concepto de pseudoespectro angular como el inverso del error monopolso y expresado en escala logarítmica. Este pseudoespectro será una función definida para todo valor de theta y presentará un máximo en el valor de theta en el que se encuentre el tag. A lo largo de los próximos ejemplos, se mostrarán tanto funciones de error monopolso como pseudoespectros angulares.

$$PEA(\theta) = 10 \cdot \log_{10} \left(\frac{1}{E_D(\theta)} \right) \quad (5)$$

4.3. Pruebas de localización angular

Gracias al entorno de MATLAB podemos procesar algunos de los datos obtenidos para los diagramas de radiación y así determinar si se está realizando una correcta estimación de los ángulos de llegada de los tags. Dado que los valores de RSSI medidos son conocidos para cada valor de theta, algunos de éstos pueden ser escogidos al azar para ser procesados y aplicar sobre ellos la estimación del DoA.

Como se mencionó anteriormente, para cada valor de theta se obtuvieron 20 RSSI por antena y por canal, a las cuales posteriormente se le realizó una media con el fin de obtener el diagrama de radiación. Una buena práctica que pone de manifiesto el funcionamiento del sistema monopulso es escoger parejas de RSSI procedentes de las dos antenas en distintos canales y aplicar el procesamiento de estimación del ángulo de llegada. Por ello, a continuación, se mostrarán resultados obtenidos para distintos ángulos de testeo conocidos.

- Para el ángulo 0° en el canal 1:

Para uno de los pares de valores de RSSI se obtiene el siguiente error monopulso:

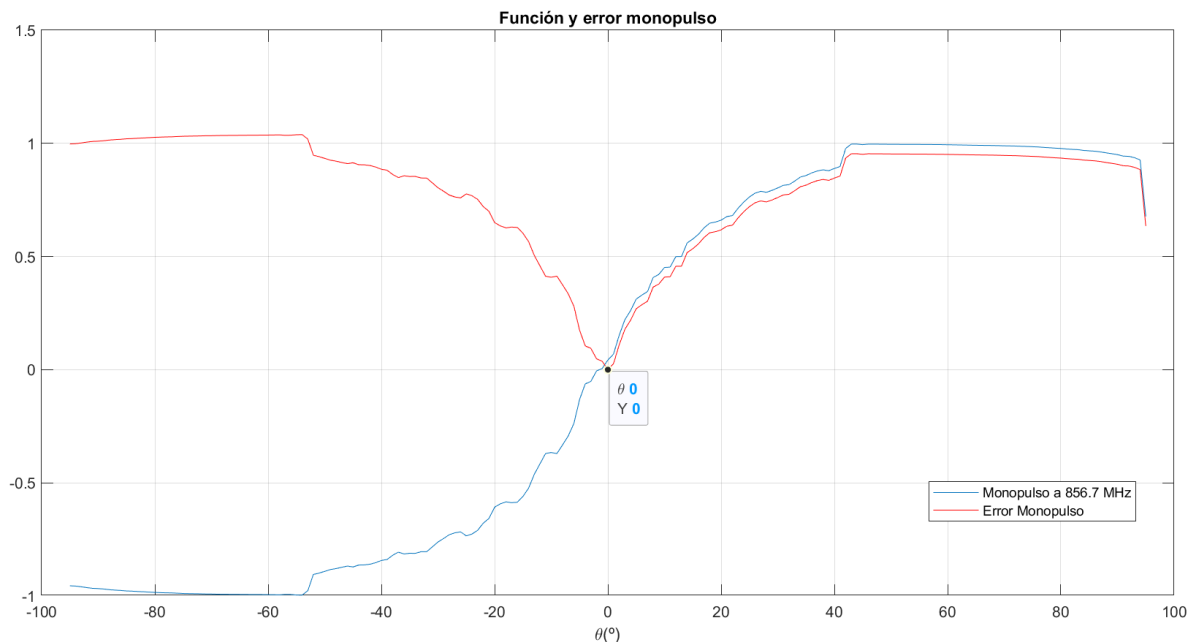


Figura 4.3. Error monopulso en el ángulo 0°

Los valores monopulso obtenidos en cada muestra se pueden ver a continuación.


```

monopulso_datos =
Columns 1 through 11
    0    0.0575    0.0575    0.0575    0.0575    0.0575    0    0.0575    0.0575    0.0575    0
Columns 12 through 21
    0    0.0575    0.0575    0.1146    0.0575    0.0575    0.0575    0    0    0.0426

```

Figura 4.4. Valores monopulso en el ángulo 0°

Los distintos pseudoespectros angulares para las 21 muestras testeadas son los siguientes:

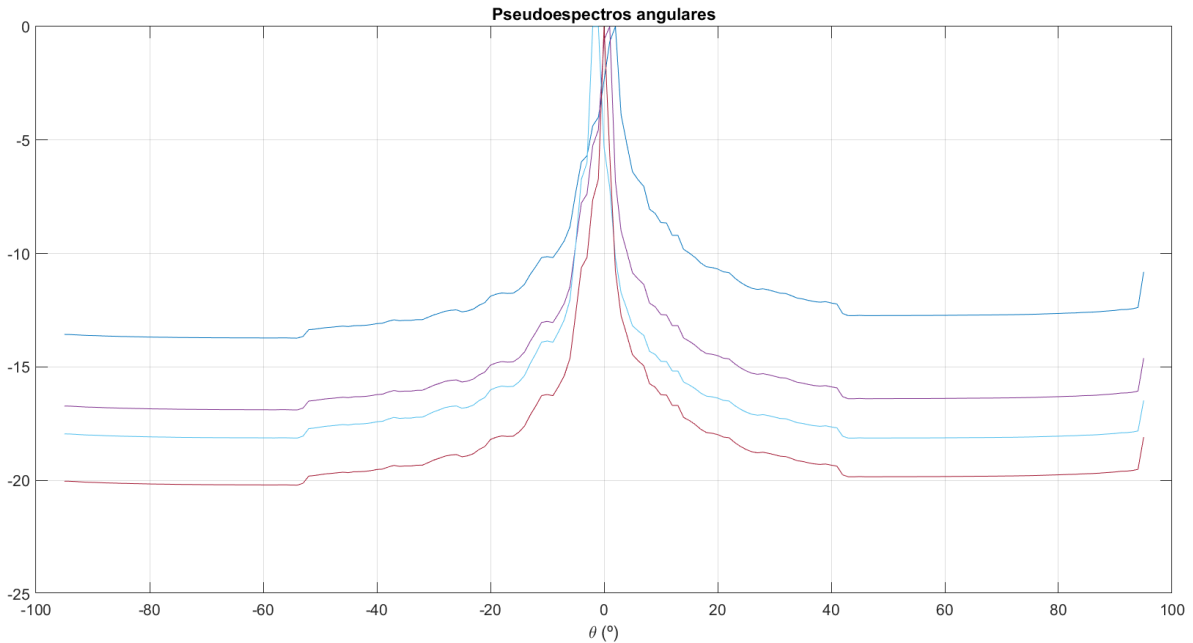


Figura 4.5. Pseudo espectros angulares en el ángulo 0°

Como se observa, la mayoría de los valores monopulso obtenidos, al compararlos con la función monopulso nos indican que el tag, en el momento de la toma de la RSSI, se encontraba en el ángulo 0°. Esto se pone de manifiesto en la representación de los pseudoespectros angulares, que en muchos de los casos son coincidentes, de ahí que solo obtengamos 4 curvas diferentes.

Finalmente, se pueden observar los errores cometidos, dado que sabemos que el ángulo de testeo es 0°, restando a dicho valor el DoA estimado, dando lugar a la siguiente representación. Teniendo en cuenta que las DoA en cada muestra han resultado ser:

```

doa_estimadas =
Columns 1 through 19
-1  1  1  1  1  1 -1  1  1  1 -1 -1  1  1  2  1  1  1 -1
Columns 20 through 21
-1  0

```

Figura 4.6. DoA estimadas en el ángulo 0°

Los errores de estimación son los mostrados en la figura 4.7.

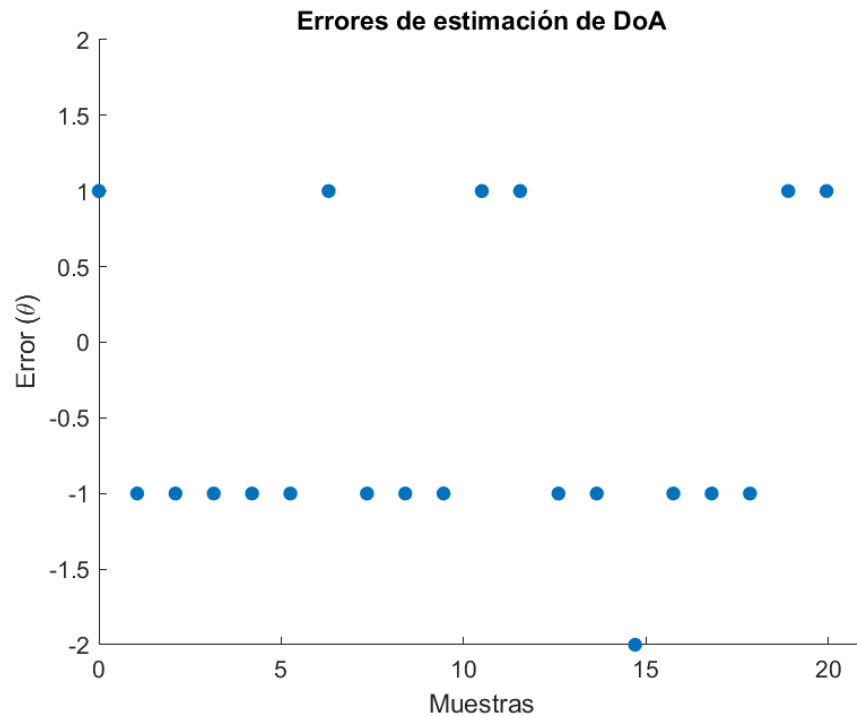


Figura 4.7. Errores de estimación del DoA en el ángulo 0°

Cabe destacar que la última muestra representada se trata de los pares de valores de RSSIs resultados de la media de los anteriores, esto es, aquellos que dieron cabida a la representación de los diagramas de radiación digitales. Por esta razón, resulta lógico que el ángulo estimado sea de 0°, ya que la función monopulso es fruto del mismo.

- Para el ángulo -15° en el canal 2:

El procedimiento es análogo al caso anterior, pero esta vez, empleando la función monopoluso que corresponde, es decir, la del canal 2. Los resultados obtenidos se mostrarán a continuación.

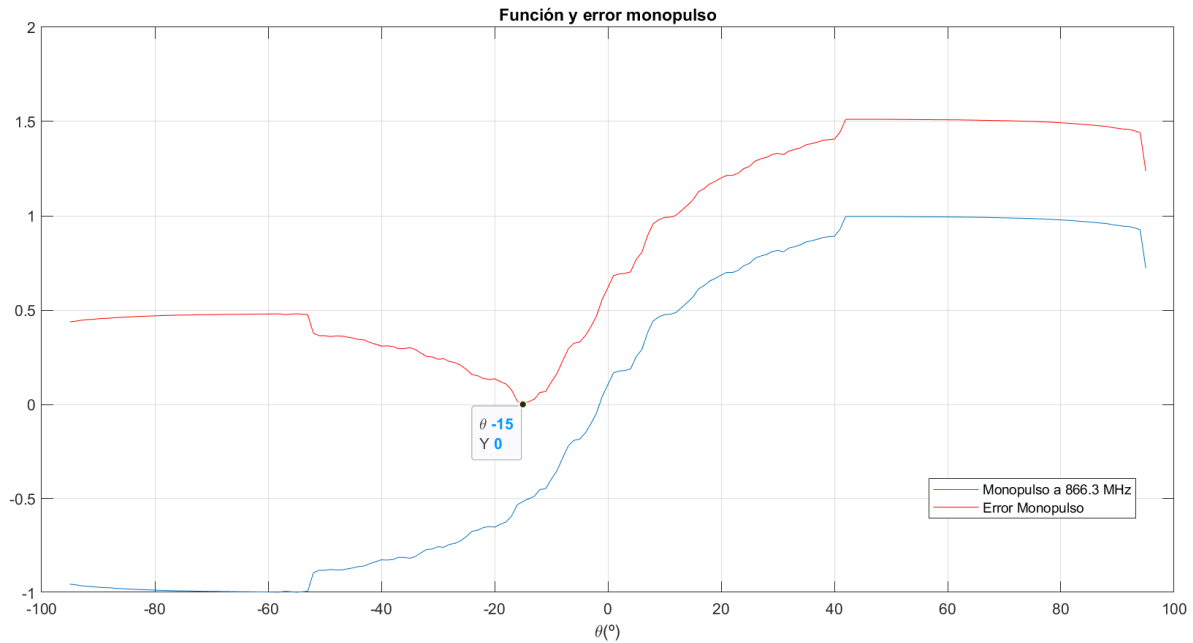


Figura 4.8. Error monopoluso en el ángulo -15°

Los valores monopoluso para cada par de RSSIs son los siguientes:

```
monopulso_datos =
Columns 1 through 12
-0.5195 -0.5195 -0.5195 -0.5195 -0.5195 -0.5195 -0.5195 -0.4762 -0.5195 -0.5195 -0.5195 -0.5195
Columns 13 through 21
-0.4762 -0.5195 -0.5195 -0.5195 -0.5195 -0.5195 -0.5195 -0.5195 -0.5150
```

Figura 4.9. Valores monopoluso en el ángulo -15°

Como se observa, el error monopoluso presenta un mínimo en el ángulo -15° , tal y como se esperaba. Los valores monopoluso son negativos, lo cual es lógico y de esperar, porque la función monopoluso tiene valores negativos para ángulos negativos. Al igual que antes, los distintos pseudoespectros (figura 4.10) angulares rondarán este valor.

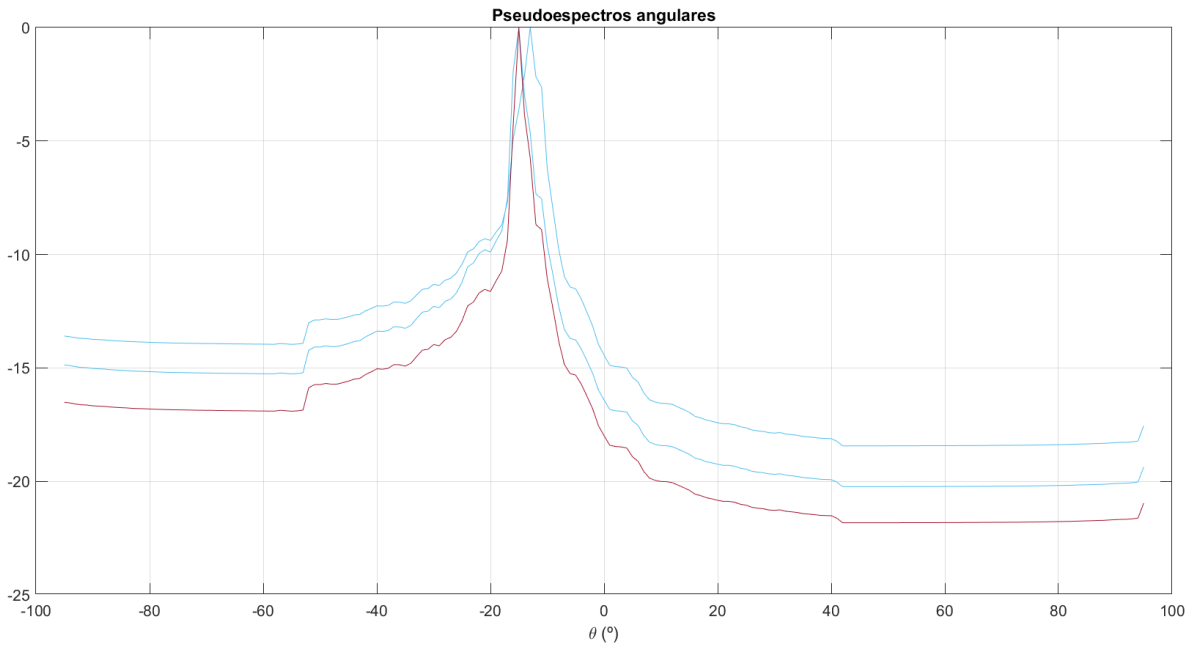


Figura 4.10. Pseudoespectros angulares para el ángulo -15°

Los distintos DoA estimados en este testeo se muestran en la siguiente figura.

```

doa_estimadas =
Columns 1 through 20
-15 -15 -15 -15 -15 -15 -15 -13 -15 -15 -15 -15 -13 -15 -15 -15 -15 -15 -15
Column 21
-15

```

Figura 4.11. DoA estimadas para el ángulo -15°

Los errores de estimación quedan representados en la figura 4. 12, donde se puede observar que el método empleado para la localización es bastante fiable.

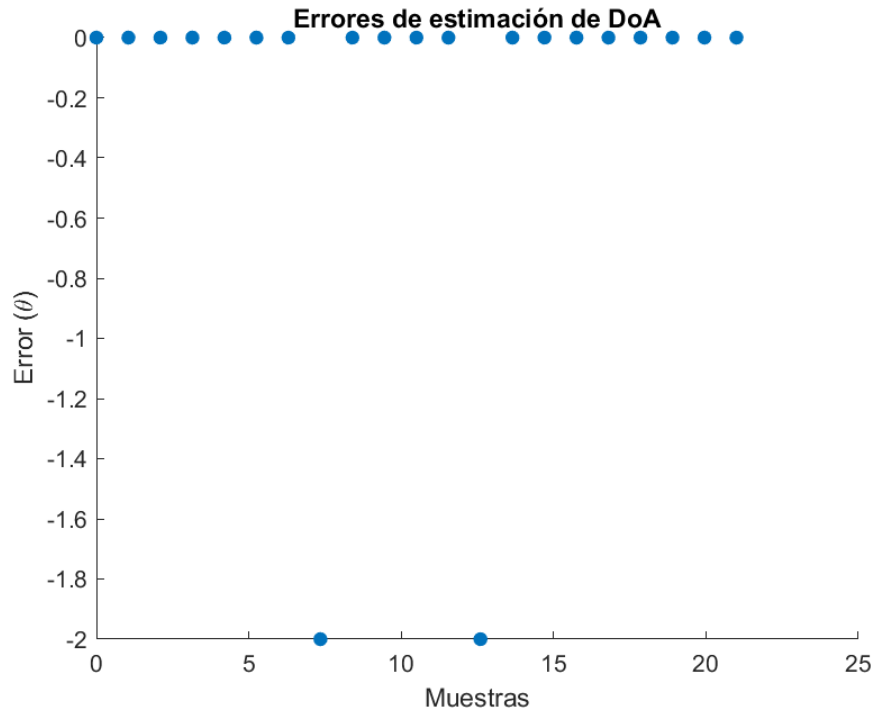


Figura 4.12. Errores de estimación del DoA en el ángulo -15°

- Para el ángulo 27° en el canal 3:

En este caso, la función monopulso empleada es la correspondiente a la frecuencia 866,9 MHz, es decir, la del tercer canal. El mínimo de la función de error monopulso nos mostrará el DoA estimado en este caso, al igual que para los anteriores.

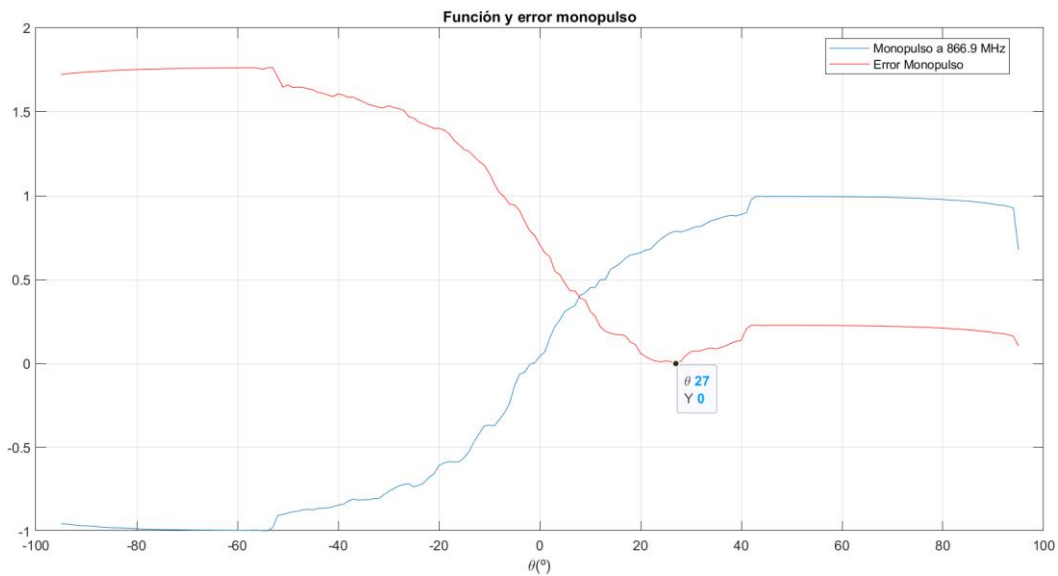


Figura 4.13. Error monopulso en el ángulo 27°

En esta ocasión, hemos obtenido los siguientes valores monopulso.

```
monopulso_datos =  
  
Columns 1 through 11  
    0.7525    0.7764    0.7764    0.7764    0.7764    0.7764    0.7764    0.7764    0.7525    0.7525    0.7764  
  
Columns 12 through 21  
    0.7764    0.7525    0.7764    0.7525    0.7525    0.7525    0.7764    0.7764    0.7764    0.7679
```

Figura 4.14. Valores monopulso en el ángulo 27°

En esta ocasión, los valores monopulso son positivos, correspondiéndose con la zona angular positiva como se puede ver en la función monopulso.

Podemos observar que los distintos pseudoespectros angulares rondan el valor de los 27° en la figura 4.15.

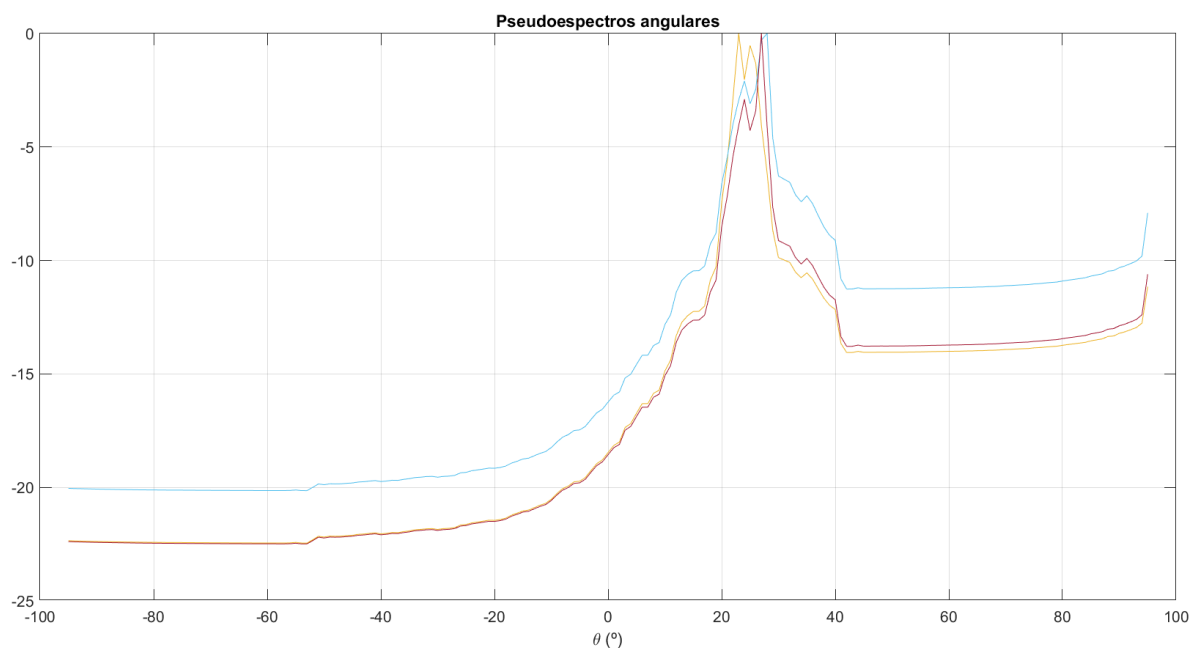


Figura 4.15. Pseudoespectros angulares para el ángulo 27°

Las distintas DoA estimadas para cada pareja de valores de RSSI se pueden ver a continuación.

```

doa_estimadas =
Columns 1 through 19
    23    28    28    28    28    28    28    28    23    23    28    28    23    28    23    23    23    28    28
Columns 20 through 21
    28    27

```

Figura 4.16. DoA estimadas para el ángulo 27°

Los errores cometidos como vemos han sido pequeños en algunos casos, gráficamente se muestra esta evolución en la estimación en la figura 4.17.

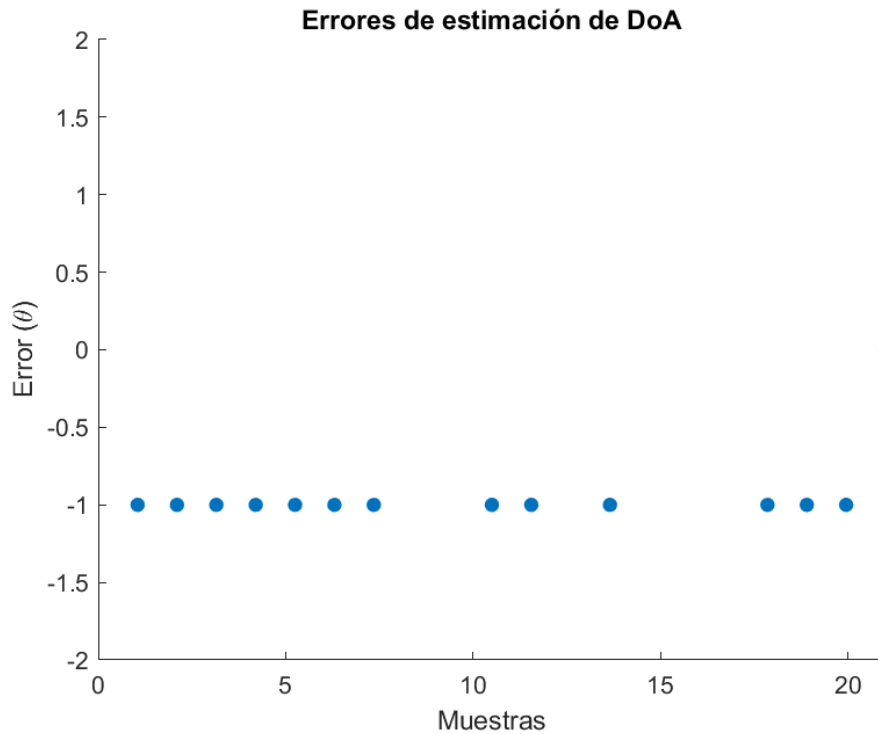


Figura 4.17. Errores de estimación del DoA en el ángulo 27°

Finalmente, veamos un caso en el que se emplea la función monopulso del canal 4, para comprobar el correcto funcionamiento y procesado de todos los datos obtenidos. En este caso el testeo puede realizarse sobre un caso límite en la parte negativa, como por ejemplo -50°.

- Para el ángulo -50° en el canal 4:

Como podemos observar, se sigue haciendo una correcta estimación del ángulo de llegada de la información del tag. El mínimo de la función de error se encuentra en -50° .

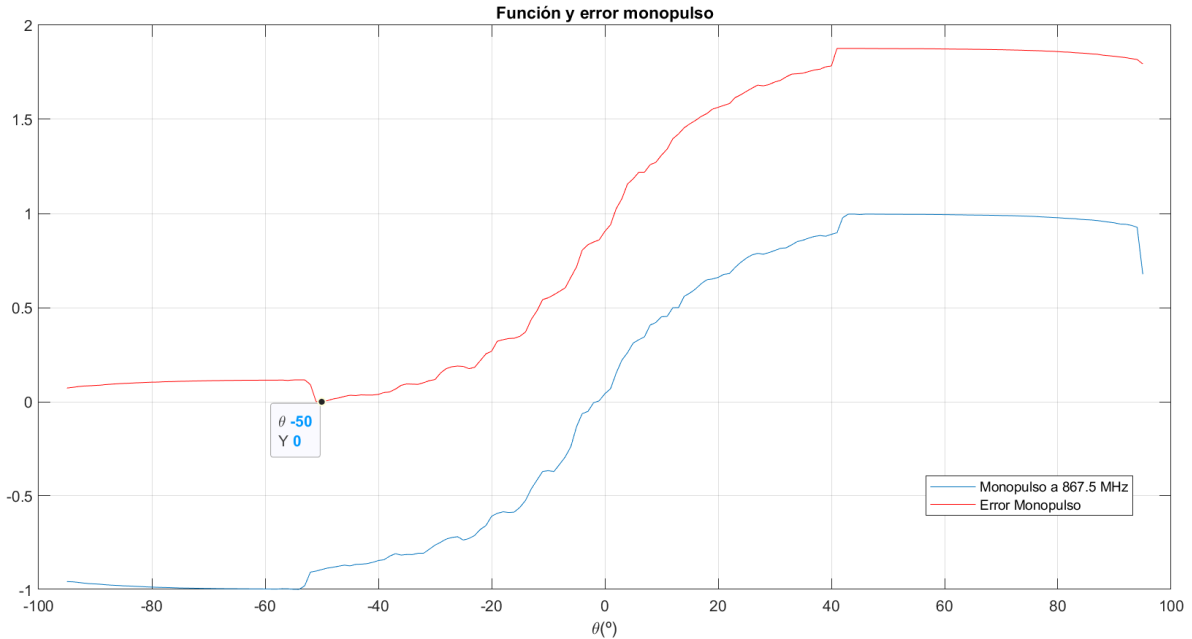


Figura 4.18. Error monopulso para el ángulo -50°

Como es de esperar, los valores monopulsos obtenidos son negativos y de valor más bajo y próximo a -1 que en el primer caso expuesto en esta memoria.

```
monopulso_datos =
Columns 1 through 12
-0.8813 -0.8813 -0.8813 -0.9387 -0.8935 -0.8813 -0.8678 -0.8813 -0.8678 -0.8813 -0.8678 -0.8528
Columns 13 through 21
-0.9045 -0.8813 -0.8678 -0.8813 -0.8813 -0.8528 -0.8678 -0.8813 -0.8795
```

Figura 4.19. Valores monopulso para el ángulo -50°

Los distintos pseudoespectros también rondan, al igual que el error monopolso, el valor de -50° .

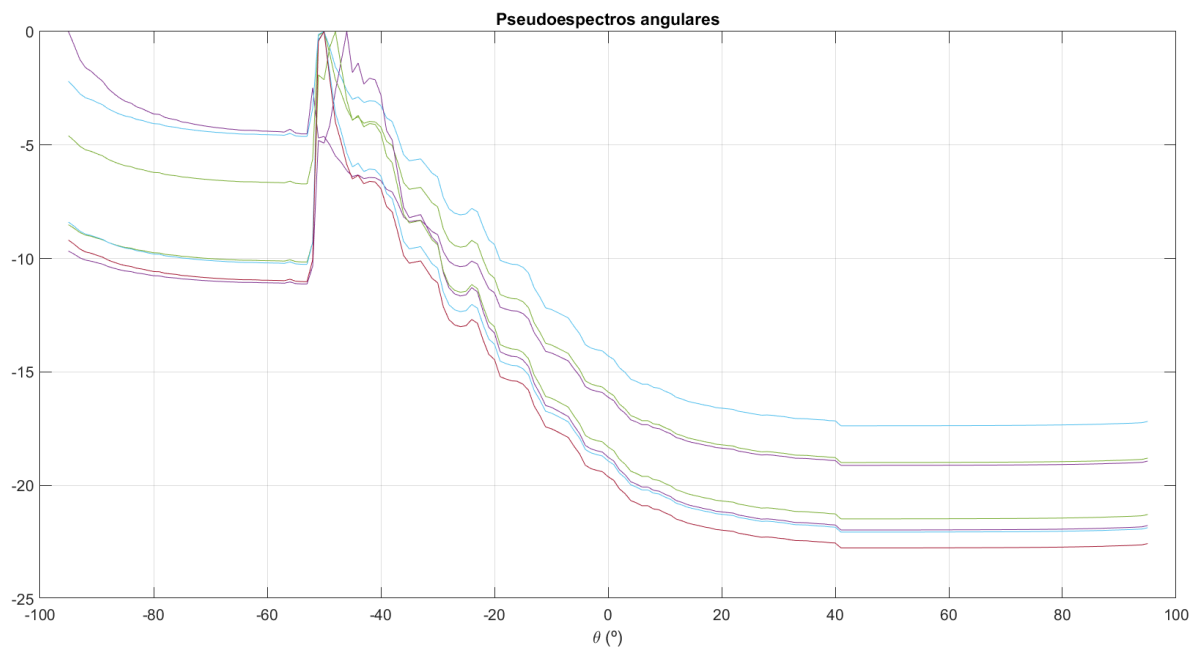


Figura 4.20. Pseudoespectros angulares para el ángulo -50°

Veamos ahora las DoA estimadas en este caso.

```

doa_estimadas =
Columns 1 through 20
-50 -50 -50 -95 -50 -50 -48 -50 -48 -50 -48 -46 -50 -50 -48 -50 -50 -46 -48 -50
Column 21
-50

```

Figura 4.21. DoA estimadas para el ángulo -50°

Como vemos, en la mayoría de los casos el error es bastante pequeño, salvo alguna excepción, como para el segundo par de muestras. La estimación sigue siendo fiable en la mayoría de los casos, pero dado que la zona angular en la que estamos estimando comienza a dejar de ser lineal para la función monopolso, podemos encontrar algunos pares de valores para los cuales el error es alto.

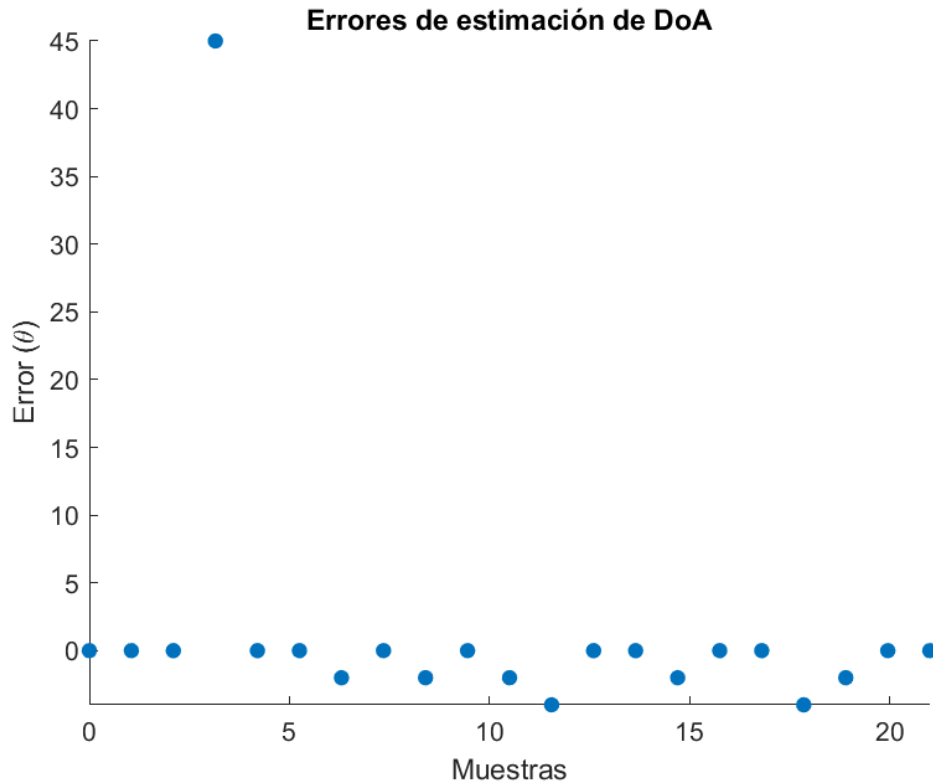


Figura 4.22. Error de estimación del DoA para el ángulo -50°

Algunos de estos resultados como por ejemplo la cuarta muestra de este último caso, ponen de relieve el hecho de que solo podemos estimar dentro del Field Of View. Fuera del mismo, la estimación no será precisa, y no podremos asociar el valor monopolso de forma inequívoca a un valor de la función monopolso para obtener a qué ángulo está asociado.

También es importante mencionar que los errores están cuantizados a 1° debido a que la función monopolso se ha construido a partir de una serie de valores que se han ido adquiriendo de grado en grado con la mesa giratoria para la obtención de los diagramas digitales. Si la adquisición de los datos se hubiera realizado con un paso de grados menor, el mínimo error que podríamos haber obtenido habría sido menor.

En términos generales, se puede concluir que el sistema monopolso funciona correctamente, a juzgar por los datos obtenidos. No obstante, esto puede considerarse un mero indicador de que el procesado de datos es el adecuado, ya que como se ha explicado al principio de este capítulo, los valores de RSSI que se han testado han sido los que han dado lugar a la función monopolso.

Capítulo 5. Experimento de localización angular de 3 tags

Una de las formas de comprobar que el procesado realizado para la estimación de la localización angular de los tags es acertado es mediante la puesta en marcha de alguna comprobación experimental. En este capítulo, se expondrán los resultados de un ensayo realizado dentro de la cámara anecoica, en el cual la adquisición de datos de amplitud procedente de los tags ha sido totalmente independiente de la usada para generar la función monopulso, cosa de la que carecían las pruebas anteriores.

En este experimento, se colocaron 3 tags RFID pasivos frente a las antenas en configuración monopulso, con una orientación con respecto a la perpendicular que era conocida previamente. La posición relativa de los mismos era de -10° , 0° y 10° y la mesa giratoria que soportaba las antenas se mantuvo fija durante todo el proceso en 0° . Conocidos estos datos, el procesado posterior puede darse o no por válido. La posición de los tags queda representada en la siguiente figura, donde podemos ver que el identificador de cada uno de ellos también es conocido.

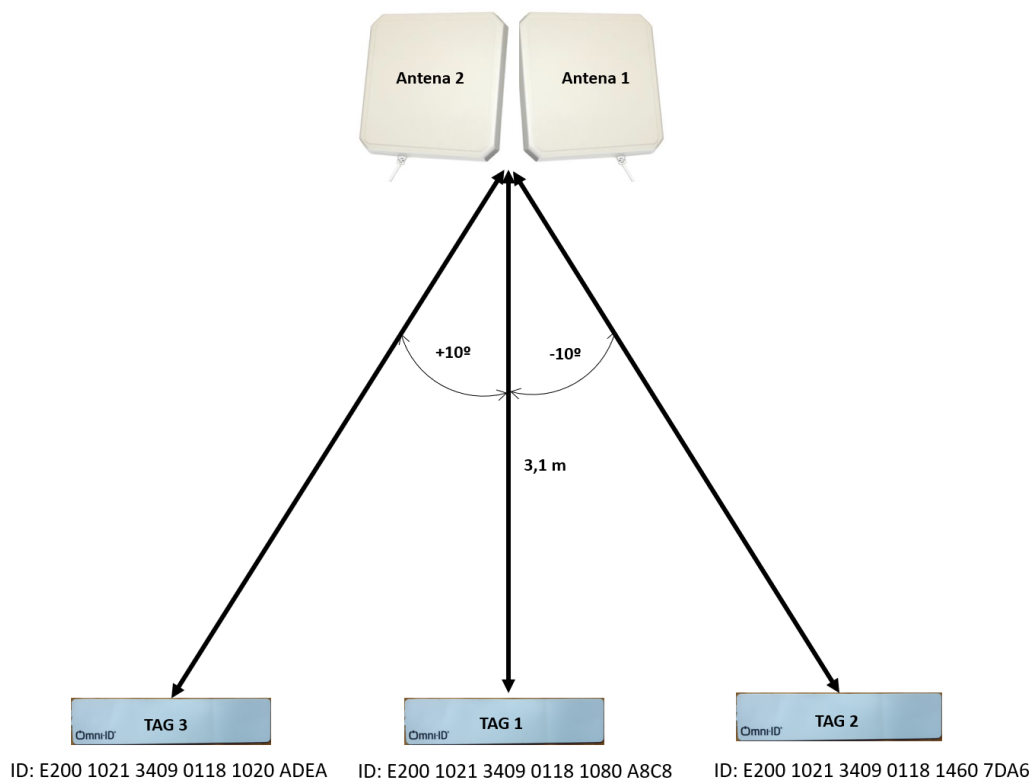


Figura 5.1. Posición de los tags en el experimento de localización

Conviene recordar que la antena 2 es la que apunta hacia ángulos positivos y la antena 1 la que apunta hacia ángulos negativos. Así como que la adquisición de las RSSIs de cada tag va a llegar secuencialmente conforme se vayan enviando tramas gracias al reader. En este caso, se obtendrán, aproximadamente, 10 RSSIs por antena, canal y tag, de tal forma que podremos emparejar los valores de RSSI de cada antena según el orden de llegada de la información.

A continuación, se mostrarán los resultados obtenidos para cada uno de los tags en todas las frecuencias de trabajo, se representarán los distintos valores monopulso y los pseudoespectros angulares, así como la DoA estimada para cada par de valores.

5.1. Localización angular del tag 1

En primer lugar, realizaremos la adquisición del tag 1 que se encuentra a 0° . Se mostrarán los distintos valores de RSSI obtenidos por cada antena en cada uno de los canales para cada una de las 10 muestras tomadas. Como se podrá observar en todos los canales, el nivel de RSSI obtenido es muy similar entre ambas antenas, ya que el tag se encuentra a la misma distancia de la antena 1 que de la antena 2, teniendo en cuenta que a la antena 1 hay que sumarle 1 dB de coeficiente de corrección a la hora de procesar los datos.

A continuación, se realizará la estimación para cada uno de los canales.

5.1.1. Canal 1 (865,7 MHz)

La evolución temporal de las RSSI obtenidas se aprecia en la siguiente figura.

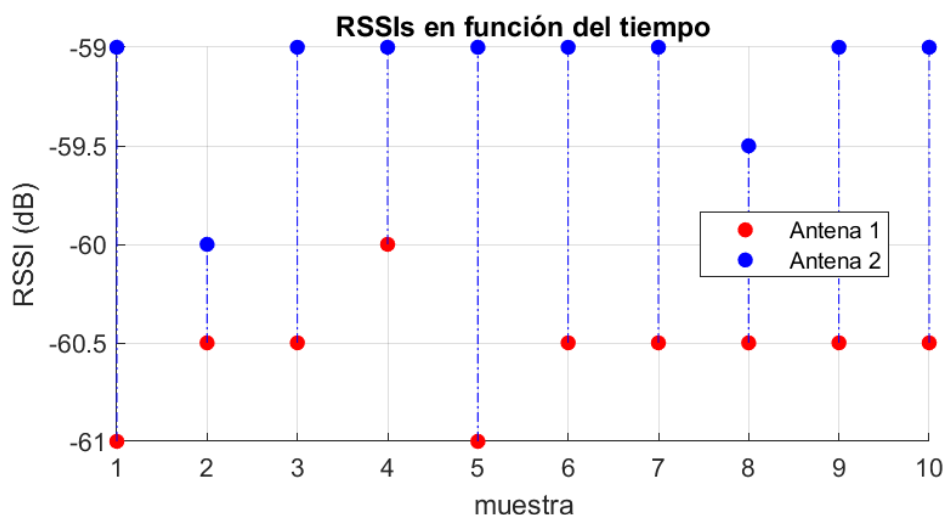


Figura 5.2. Evolución de las RSSI del tag 1 en el canal 1

Las distintas curvas de error monopulso junto a la función monopulso empleadas nos indican que la estimación del DoA se encuentra entorno al ángulo 0° .

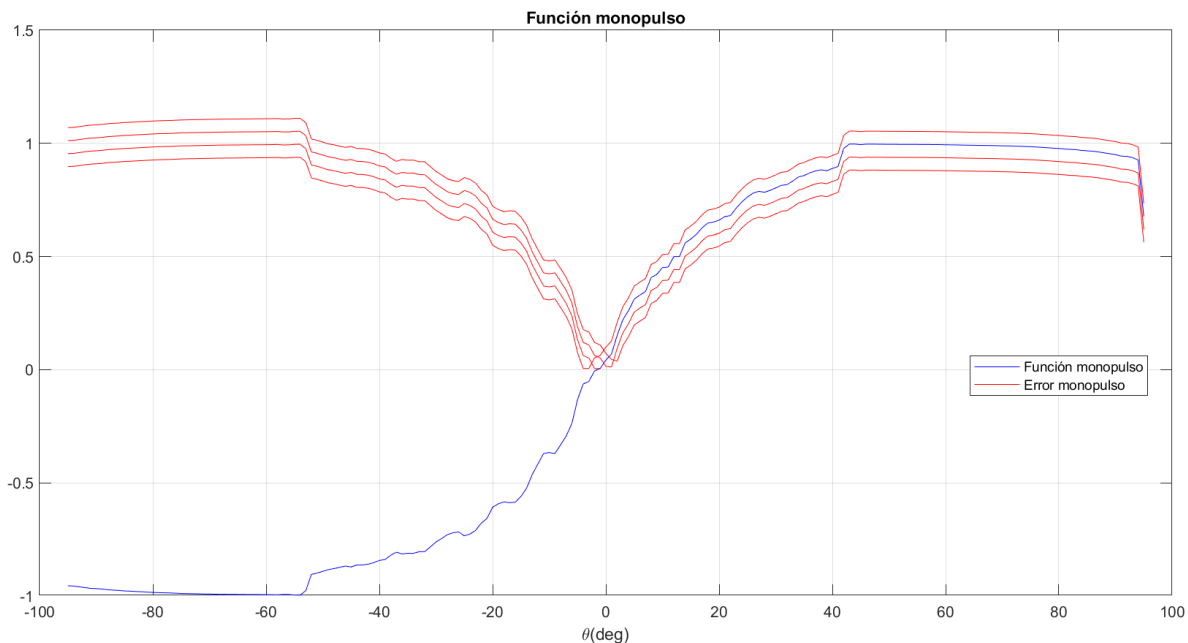


Figura 5.3. Errores monopulso del tag 1 en el canal 1

Se aprecia también que los valores monopulso de cada pareja de valores oscilan entorno al valor 0 , que es el valor que adquiere la función monopulso en el ángulo $\theta = 0^\circ$.

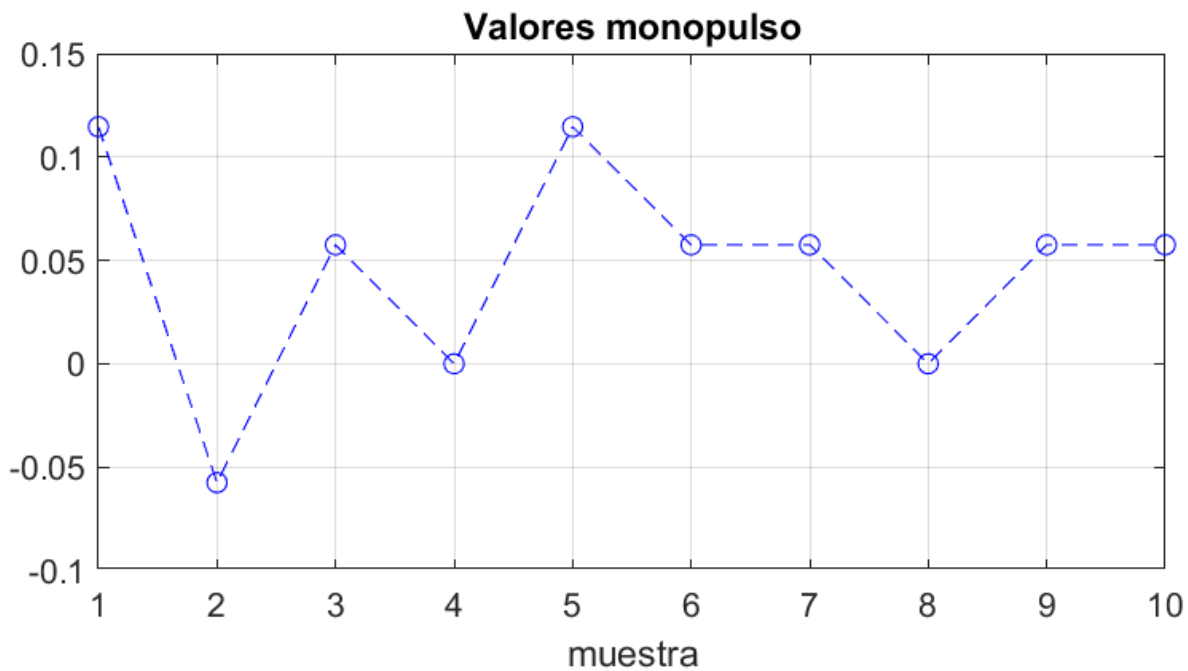


Figura 5.4. Valores monopulso del tag 1 en el canal 1

En la siguiente captura del Workspace de MATLAB podemos ver las cifras exactas de los distintos valores monopulso.

```
valores_monopulso =
    0.1146   -0.0575    0.0575         0    0.1146    0.0575    0.0575         0    0.0575    0.0575
```

Figura 5.5. Valores monopulso exactos del tag 1 en el canal 1

Por último, queda representar los distintos pseudoespectros angulares que se han obtenido, que al igual que la función de error, presentará máximos donde la anterior tenía mínimos, justo en las DoA estimadas en este caso.

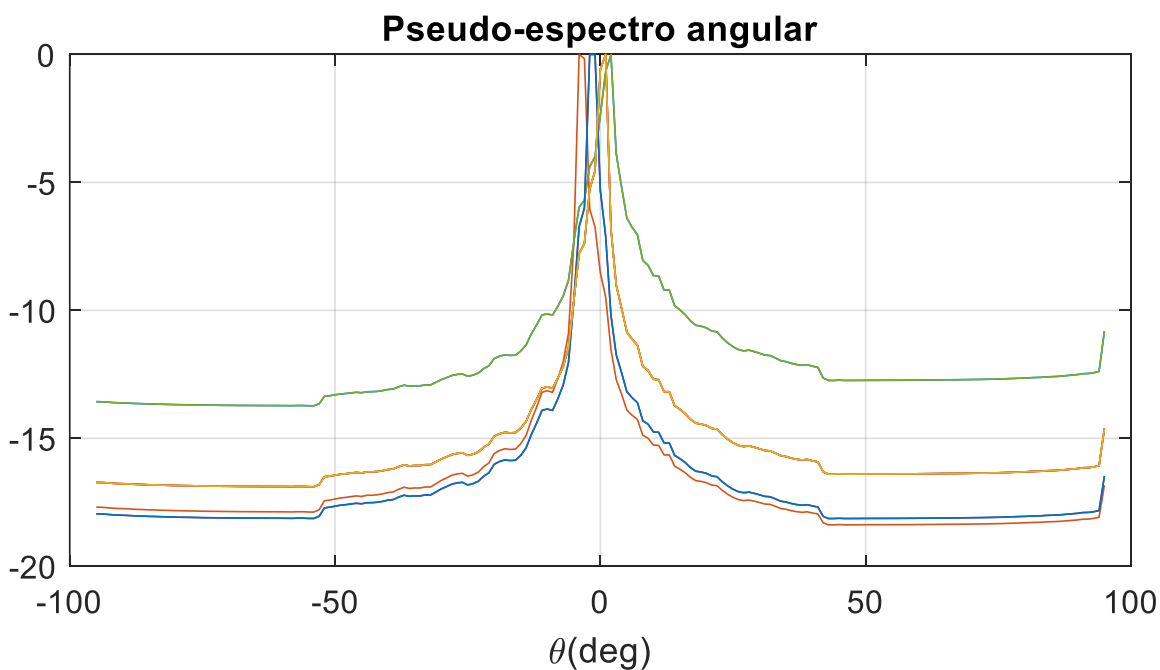


Figura 5.6. Pseudoespectros angulares para el tag 1 en el canal 1

Las distintas DoA estimadas exactas fueron las siguientes:

```
doa_estimados =
    2    -4     1    -1     2     1     1    -1     1     1
```

Figura 5.7. DoA estimadas para el tag 1 en el canal 1

Podemos observar también los errores de estimación cometidos gráficamente, teniendo en cuenta que el ángulo en el que se colocó el tag es conocido.

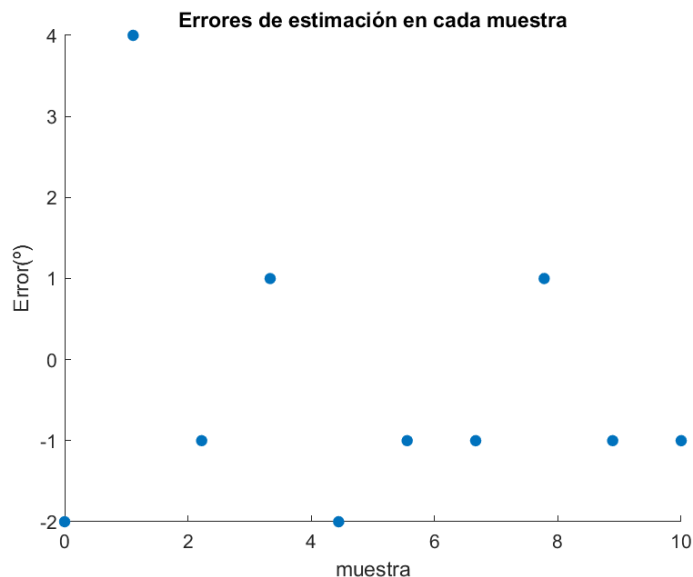


Figura 5.8. Errores cometidos para el tag 1 en el canal 1

5.1.2. Canal 2 (866,3 MHz)

En este canal de frecuencia, se obtuvieron resultados análogos a los anteriores, debido a la gran cercanía entre las frecuencias de trabajo empleadas.

Comenzaremos como antes, representando la evolución temporal de las RSSI obtenidas en la adquisición.

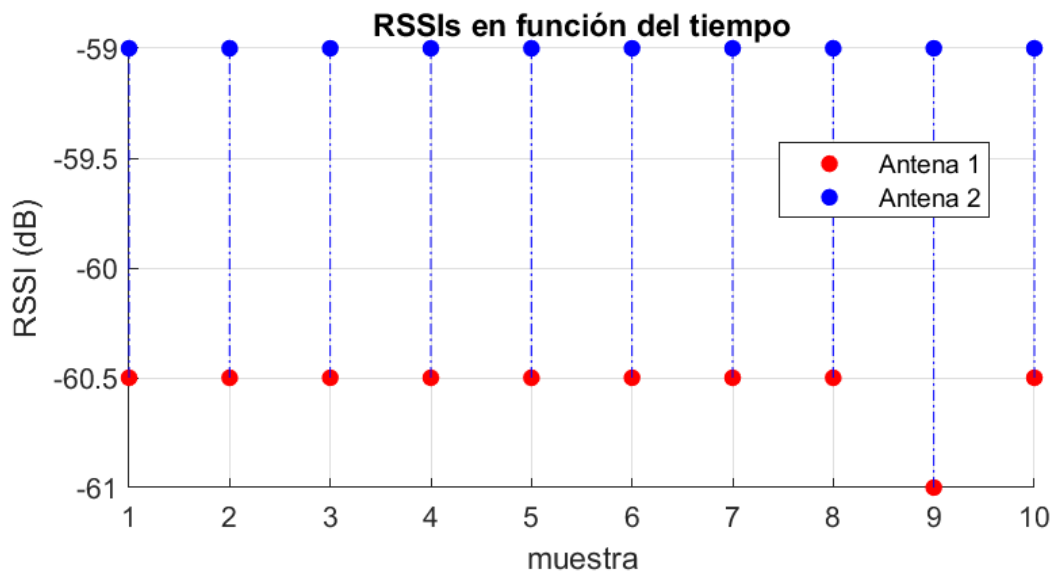


Figura 5.9. Evolución de la RSSI del tag 1 en el canal 2

Al igual que en el canal 1, la potencia de RSSI recibida entre las antenas difiere muy poco, lo cual es buena señal y un indicador de que el tag 1 se encuentra en el centro de la perpendicular.

La función monopulso empleada en este caso es la del canal 2 y las distintas funciones de error son las siguientes.

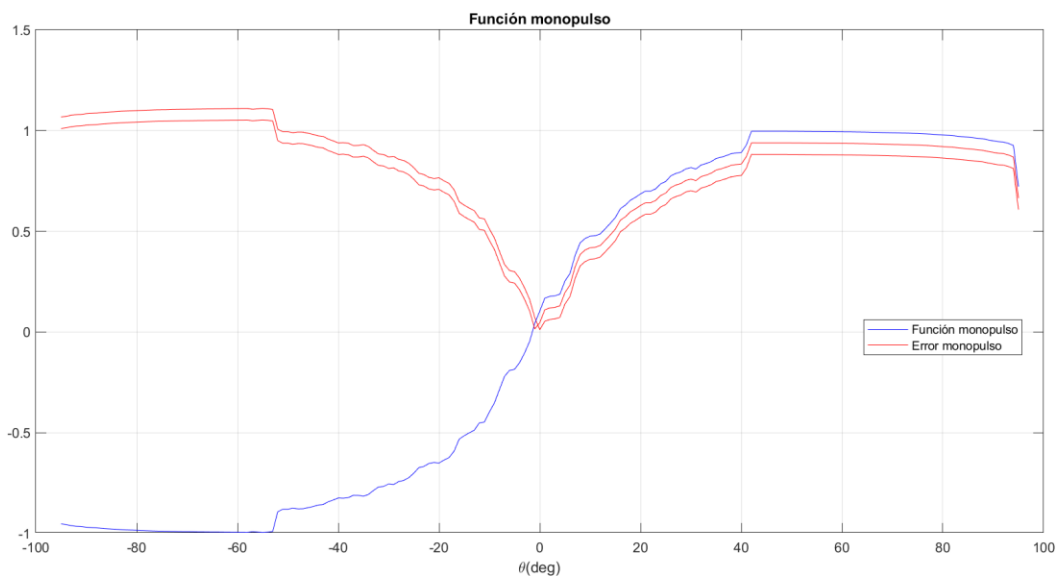


Figura 5.10. Errores monopulso para el tag 1 en el canal 2

Solo hemos obtenido dos curvas de error debido a que de entre los 10 pares de datos solo hemos obtenido dos DoA diferentes. Esto queda reflejado en los distintos valores monopulso obtenidos, donde solamente hay uno que difiere al resto.



Figura 5.11. Valores monopulso para el tag 1 en el canal 2

Los valores, a priori, parecen ser mejores que en el canal 1, dado que son más próximos al valor 0, que se corresponde con $\theta = 0^\circ$.

```
valores_monopulso =  
    0.0575    0.0575    0.0575    0.0575    0.0575    0.0575    0.0575    0.0575    0.1146    0.0575
```

Figura 5.12. Valores monopulso exactos para el tag 1 en el canal 2

Los distintos pseudoespectros angulares son coherentes con los resultados obtenidos.

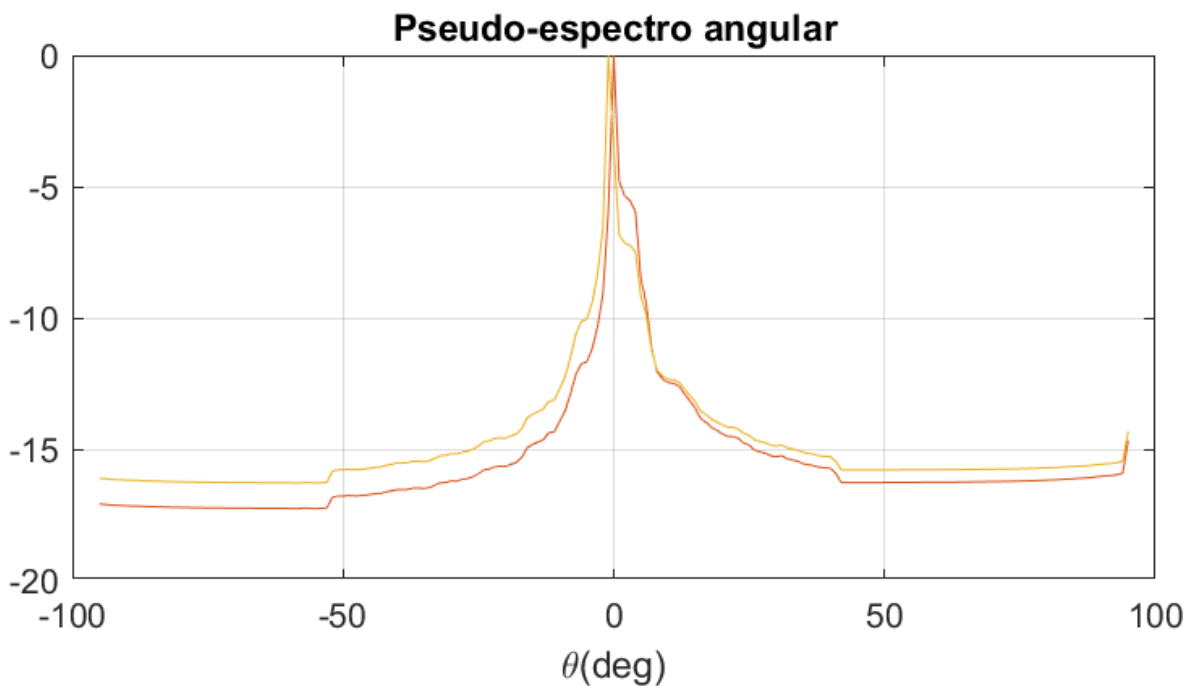


Figura 5.13. Pseudoespectros angulares para el tag 1 en el canal 2

Las distintas DoA estimadas para este canal empleando el tag 1, como era de esperar, son mucho más precisos.

```
doa_estimados =  
    -1    -1    -1    -1    -1    -1    -1    -1    0    -1
```

Figura 5.14. DoA estimadas para el tag 1 en el canal 2

Los errores de estimación en este caso son bastante más bajos que en el caso anterior como podemos apreciar en la siguiente figura.

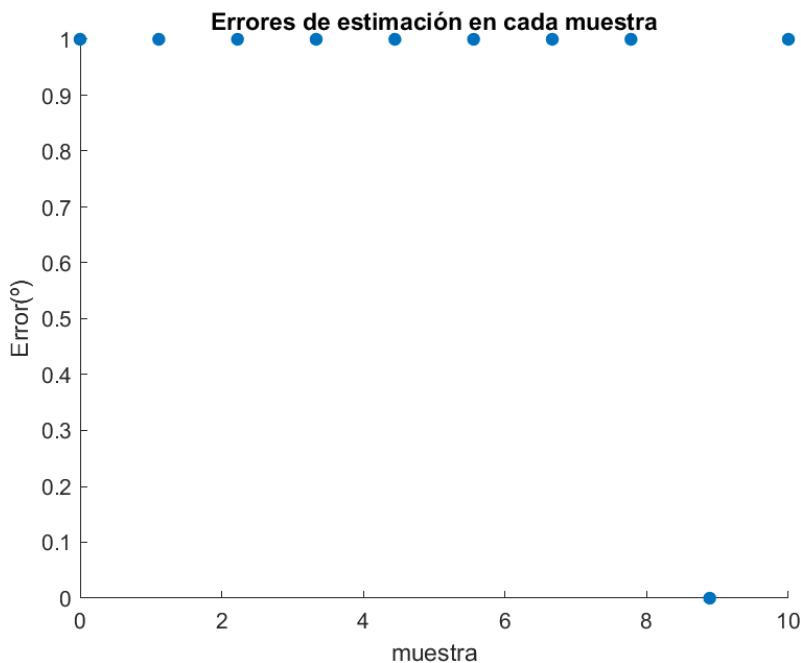


Figura 5.15. Errores de estimación para el tag 1 en el canal 2

5.1.3. Canal 3 (866,9 MHz)

En este canal, comenzaremos también por la representación de la RSSI recibida para cada muestra por ambas antenas.

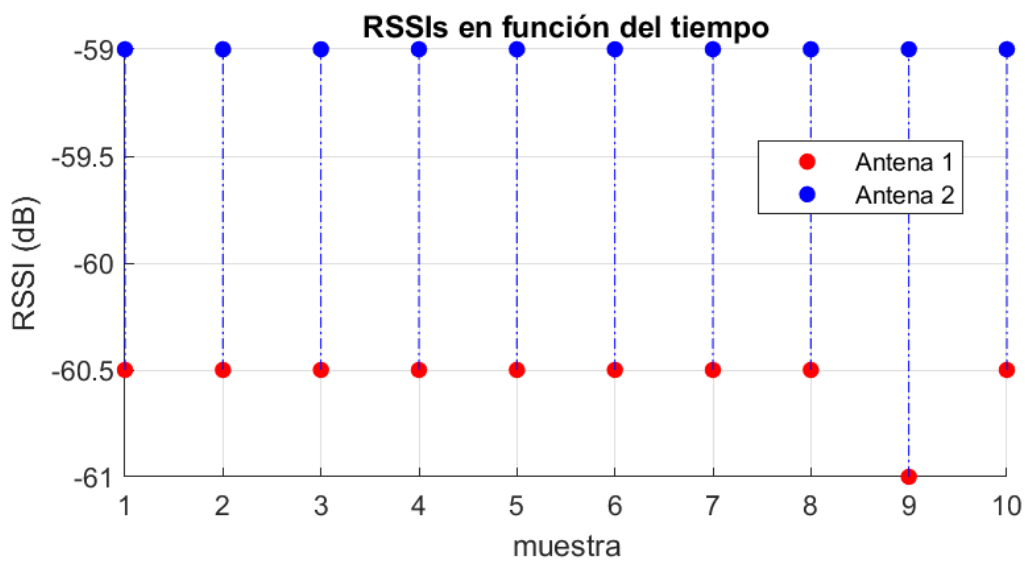


Figura 5.16. Evolución de las RSSI para el tag 1 en el canal 3

Los resultados obtenidos son exactamente los mismos que para el canal 2, no obstante, los ángulos de estimación de llegada del tag serán diferentes debido a que emplearemos la función monopulso correspondiente al canal 3 para hacer el procesado. Esta función, junto con las funciones de error monopulso, se muestran en la siguiente figura.

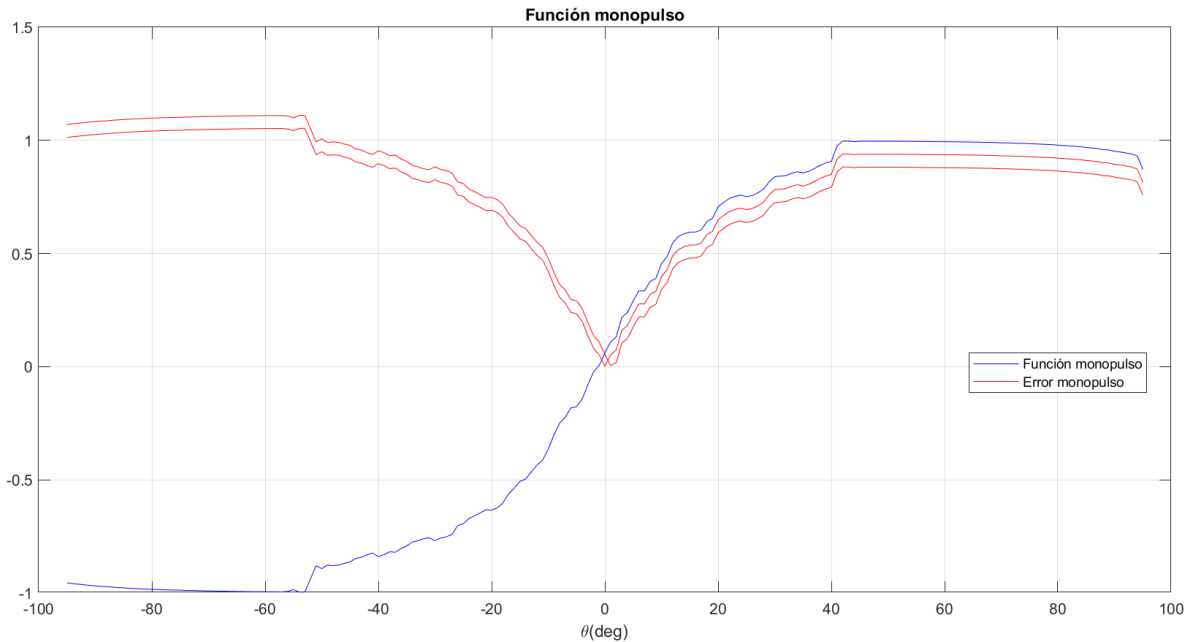


Figura 5.17. Error monopulso para el tag 1 en el canal 3

Los distintos valores monopulso se muestran en la siguiente figura. Al igual que para el canal anterior, solo hemos obtenido dos valores diferentes para cada muestra.



Figura 5.18. Valores monopulso para el tag 1 en el canal 3

Exactamente, los valores monopulso obtenidos son los siguientes.

```
valores_monopulso =  
0.0575  0.0575  0.0575  0.0575  0.0575  0.0575  0.0575  0.0575  0.1146  0.0575
```

Figura 5.19. Valores monopulso exactos para el tag 1 en el canal 3

El pseudoespectro angular obtenido en cada muestra esta vez varía por la razón mencionada anteriormente. En virtud de los resultados, podemos concluir que en este canal se ha realizado mejor estimación que en los anteriores.

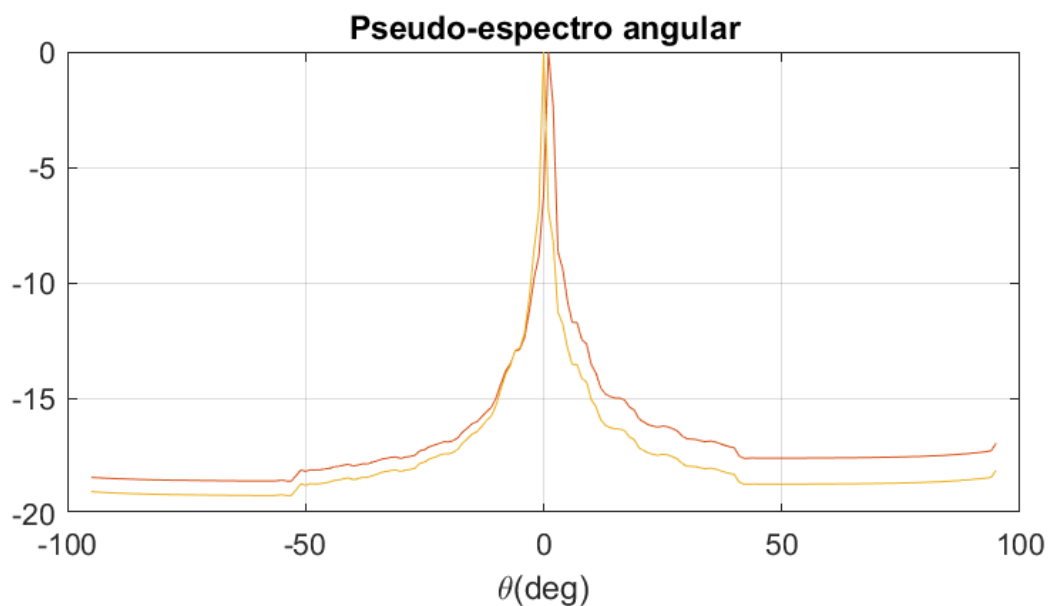


Figura 5.20. Pseudoespectros angulares para el tag 1 en el canal 3

```
doa_estimados =  
0 0 0 0 0 0 0 0 1 0
```

Figura 5.21. DoA estimadas para el tag 1 en el canal 3

A excepción de una de las muestras, hemos obtenido un DoA de 0° para todos los casos.

En este canal, la estimación se ha producido prácticamente sin errores, a excepción de una de las muestras. Esto lo convierte, de momento, en el mejor canal de estimación para este tag.

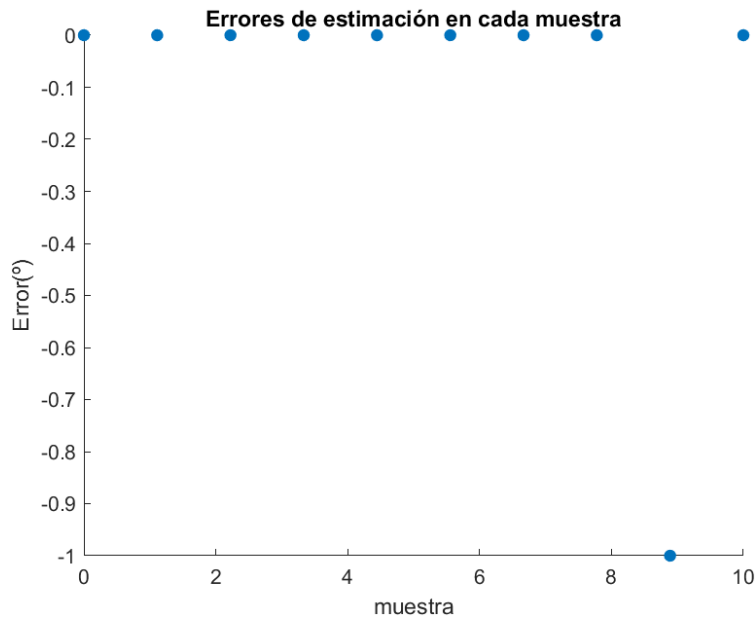


Figura 5.22. Errores de estimación para el tag 1 en el canal 3

5.1.4. Canal 4 (867,5 MHz)

El último de los canales a comprobar para este tag es el de la frecuencia de 867,5 MHz. Se procederá a la muestra de los resultados al igual que se ha venido haciendo para el resto de canales. En primer lugar, se muestra la evolución de las RSSI en el tiempo para cada par de antenas.

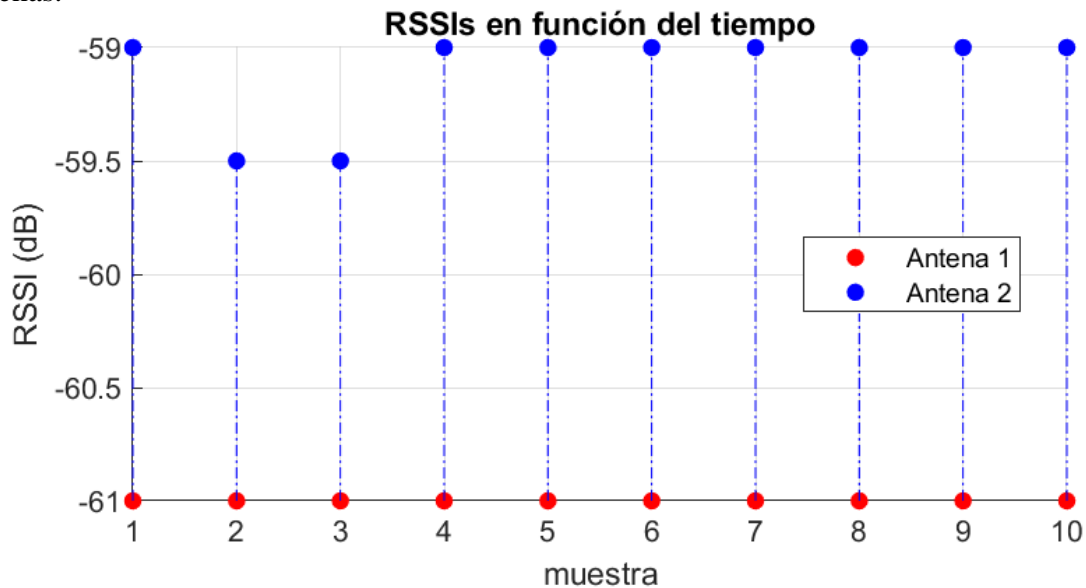


Figura 5.23. Evolución de la RSSI para el tag 1 en el canal 4

En este canal, se sigue obteniendo una RSSI mayor en la antena 2, lo cual es coherente con la posterior corrección de ganancia digital.

Las distintas funciones de error monopulso son las siguientes.

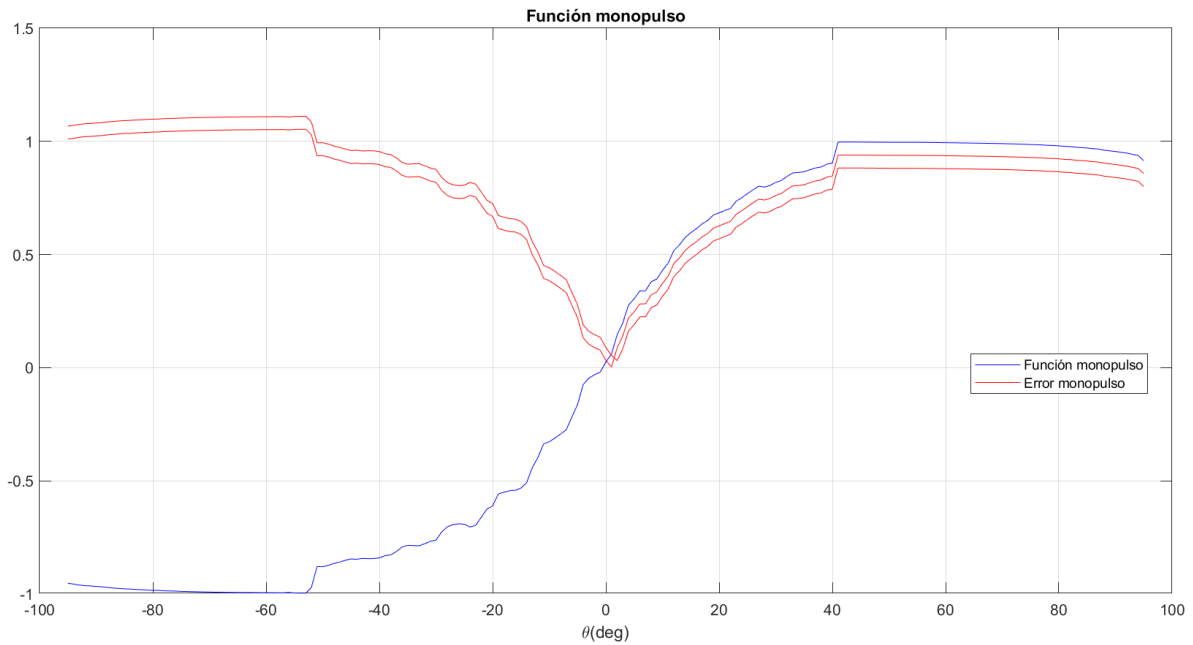


Figura 5.24. Errores monopulso para el tag 1 en el canal 4

Vemos que hay 2 valores diferentes de estimación del ángulo a pesar de tener 10 pares de valores de RSSI. Habrá, por tanto, dos valores monopulso diferentes.

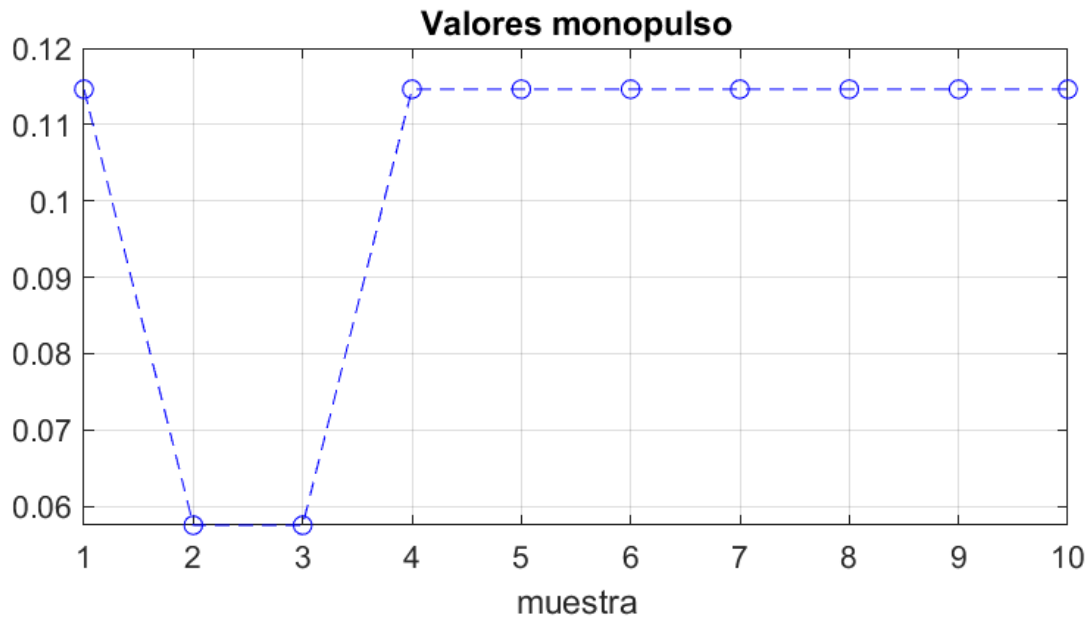


Figura 5.25. Valores monopulso para el tag 1 en el canal 4

valores_monopulso =

0.1146 0.0575 0.0575 0.1146 0.1146 0.1146 0.1146 0.1146 0.1146 0.1146

Figura 5.26. Valores monopulso exactos para el tag 1 en el canal 4

Los distintos pseudoespectros angulares obtenidos se muestran en la figura 5.27.

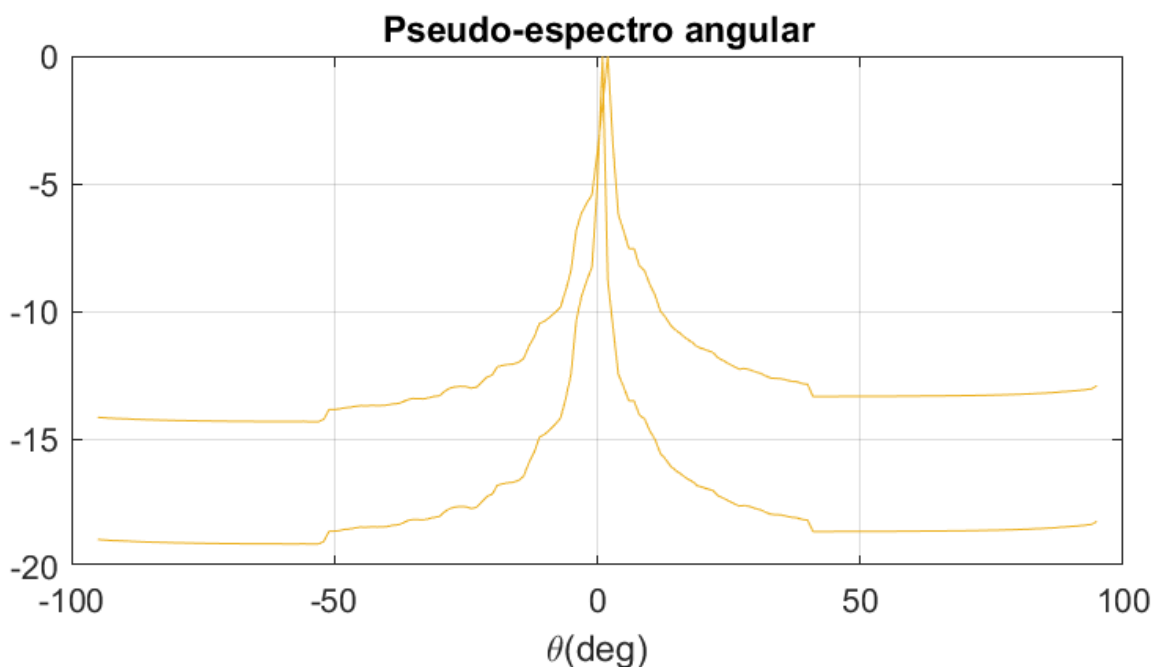


Figura 5.27. Pseudoespectros angulares para el tag 1 en el canal 4

Las direcciones de llegada estimadas de los tags también se han acercado mucho a los 0° esperados. No obstante, como la RSSI obtenida en la antena 2 es mayor que la de la antena 1 incluso aplicando el coeficiente de corrección, el procesado monopulso nos está indicando que el tag se encuentra más en la zona angular positiva, que es la correspondiente a la antena 2.

```
doa_estimados =  
  
    2    1    1    2    2    2    2    2    2    2
```

Figura 5.28. DoA estimadas para el tag 1 en el canal 4

Definitivamente, para este tag, el canal que mejor ha localizado angularmente su posición ha sido el canal 3. Ya que en el canal 4 hemos obtenido el siguiente error.

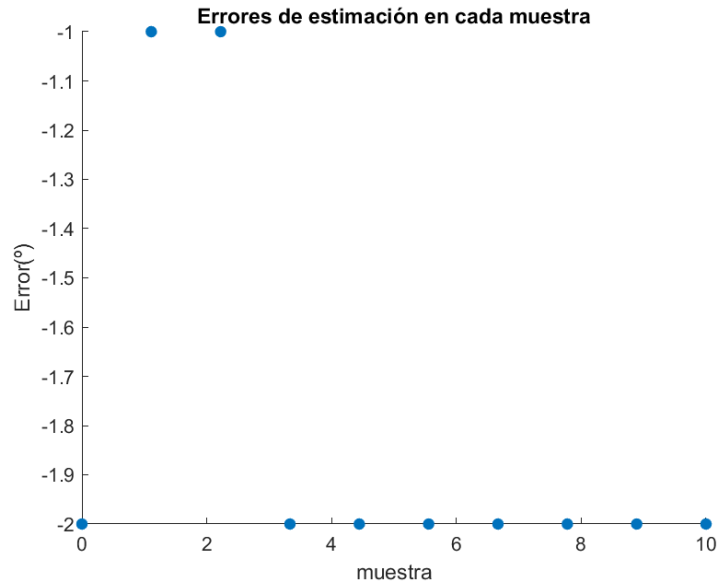


Figura 5.29. Errores de estimación para el tag 1 en el canal 4

5.2. Localización angular del tag 2

Si recuperamos la figura 5.1 se puede observar que el tag del que nos disponemos a hacer la estimación de localización se encuentra en una zona angular negativa. Concretamente, se espera un resultado de en torno a -10° en la DoA de este tag. Se procede al análisis de resultados en cada uno de los canales en las siguientes secciones.

5.2.1. Canal 1 (865,7 MHz)

Análogamente al tag anterior comenzaré por mostrar la evolución de las RSSI adquiridas durante la ejecución del experimento.

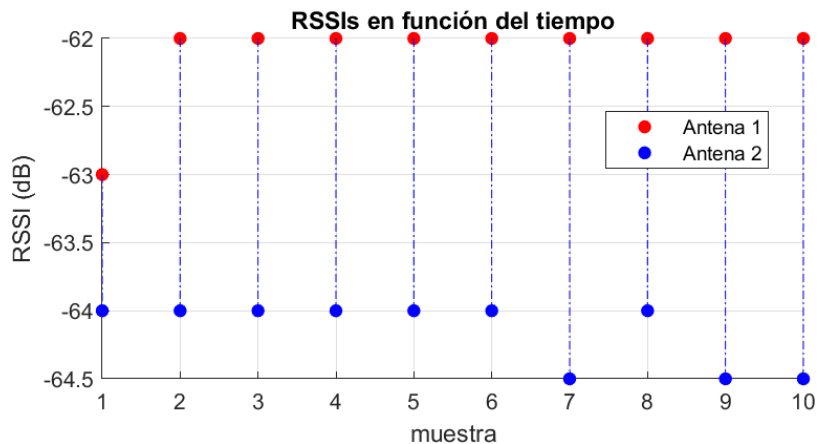


Figura 5.30. Evolución de la RSSI para el tag 2 en el canal 1

Se observa que, en esta ocasión, el nivel de RSSI recibido por la antena 1 es mayor que en la antena 2. Esto se debe a que el tag se encuentra en una zona angular negativa, hacia donde apunta la antena 1, por lo tanto, la amplitud de señal obtenida será mayor desde esta antena. Además de que habrá que aplicar el coeficiente de corrección y la diferencia entre las RSSIs será mayor entonces.

Las funciones de errores obtenidas rondan el valor de los -10° .

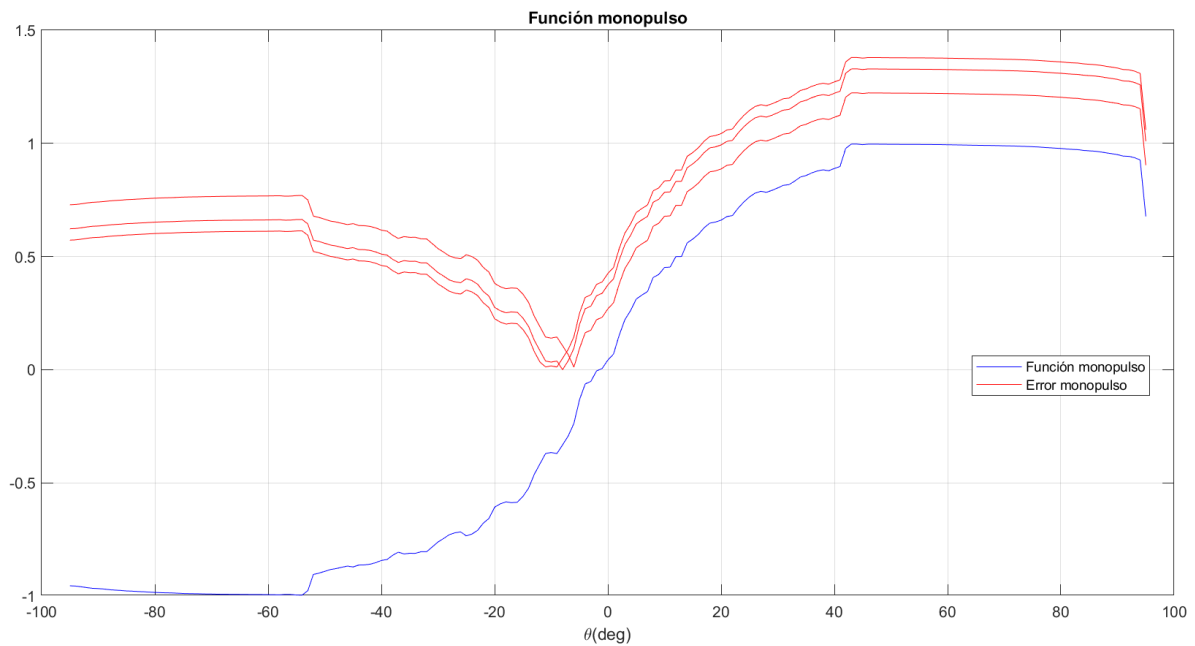


Figura 5.31. Funciones de error para el tag 2 en el canal 1

Los valores monopulso obtenidos, como era de esperar, también son negativos, ya que se corresponden con ángulos negativos de la función monopulso.

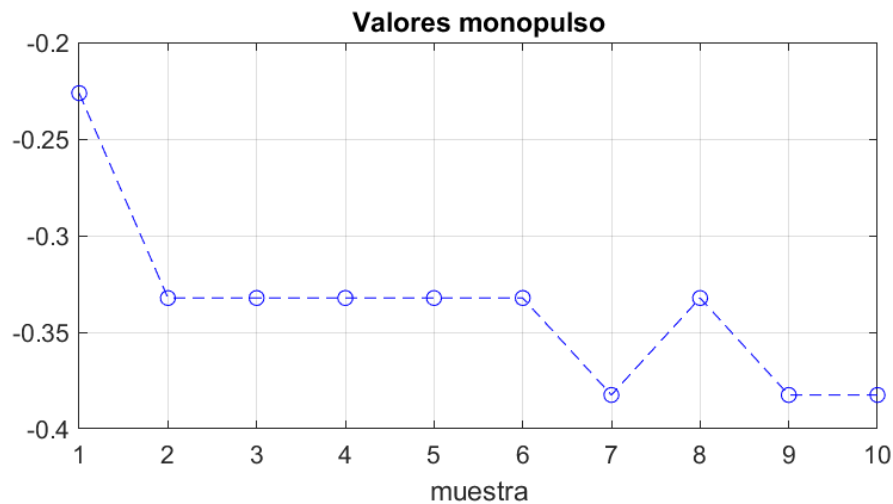


Figura 5.32. Valores monopulso para el tag 2 en el canal 1

```

valores_monopulso =
    -0.2263  -0.3323  -0.3323  -0.3323  -0.3323  -0.3323  -0.3825  -0.3323  -0.3825  -0.3825

```

Figura 5.33. Valores monopulso exactos para el tag 2 en el canal 1

Los distintos pseudoespectros angulares obtenidos, al igual que las funciones de error, presentan un máximo en los valores de theta esperados.

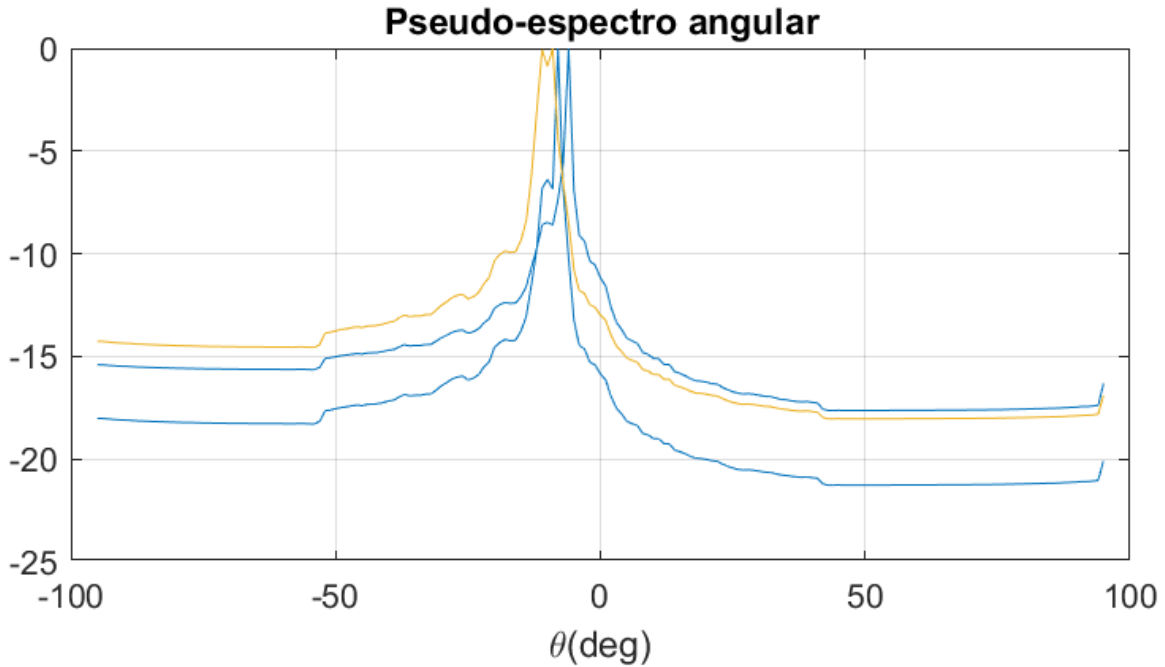


Figura 5.34. Pseudoespectros angulares para el tag 2 en el canal 1

Los ángulos en los que se ha localizado el tag para este canal se muestran en la figura 5. 31.

```

doa_estimados =
    -6  -8  -8  -8  -8  -8  -9  -8  -9  -9

```

Figura 5.35. DoA estimadas para el tag 2 en el canal 1

Como se esperaba, los resultados son bastante próximos a -10° . Y el error cometido se muestra a continuación.

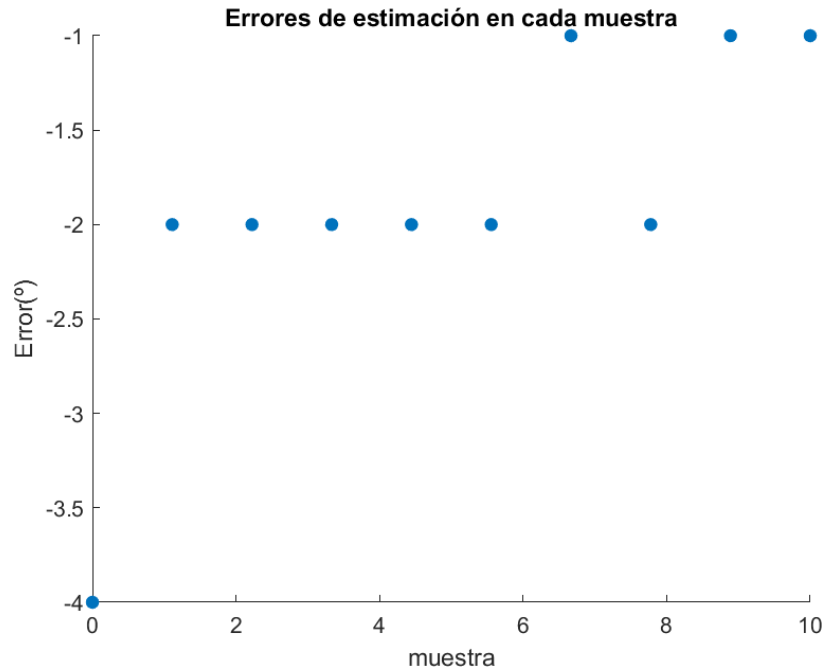


Figura 5.36. Errores de estimación para el tag 2 en el canal 1

5.2.2. Canal 2 (866,3 MHz)

En el canal 2, los resultados han sido más estables que para el resto de canales. Esto se manifiesta en la evolución de la RSSI recibida, que podremos ver en la siguiente figura, en donde la proporción entre la RSSI recibida por la antena 1 y la antena 2 prácticamente se mantiene a lo largo de todas las muestras.

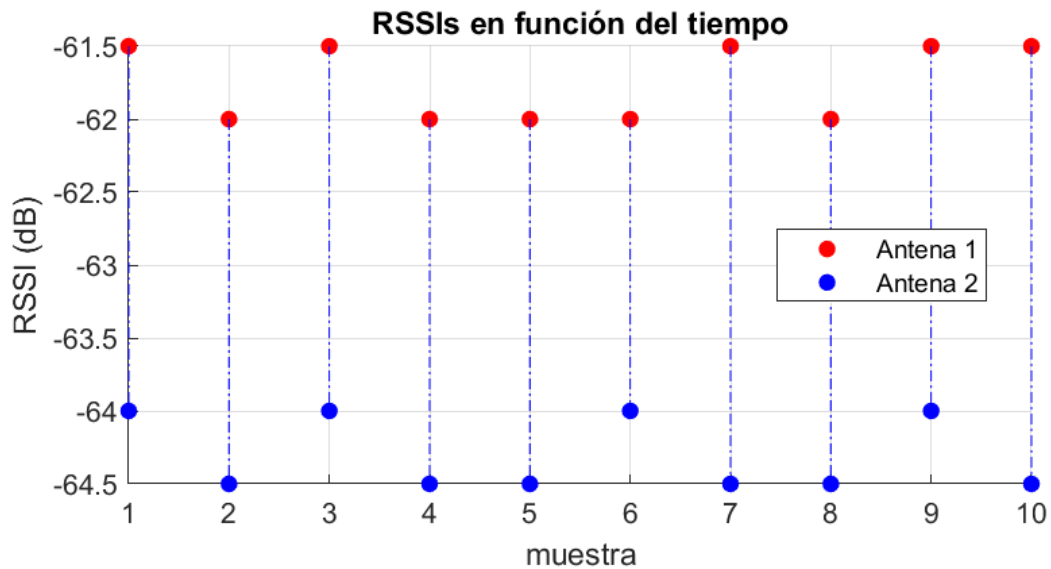


Figura 5.37. Evolución de la RSSI para el tag 2 en el canal 2

La función monopulso empleada en este caso es la del canal 2 y los distintos errores obtenidos se corresponden con $\theta = -10^\circ$.

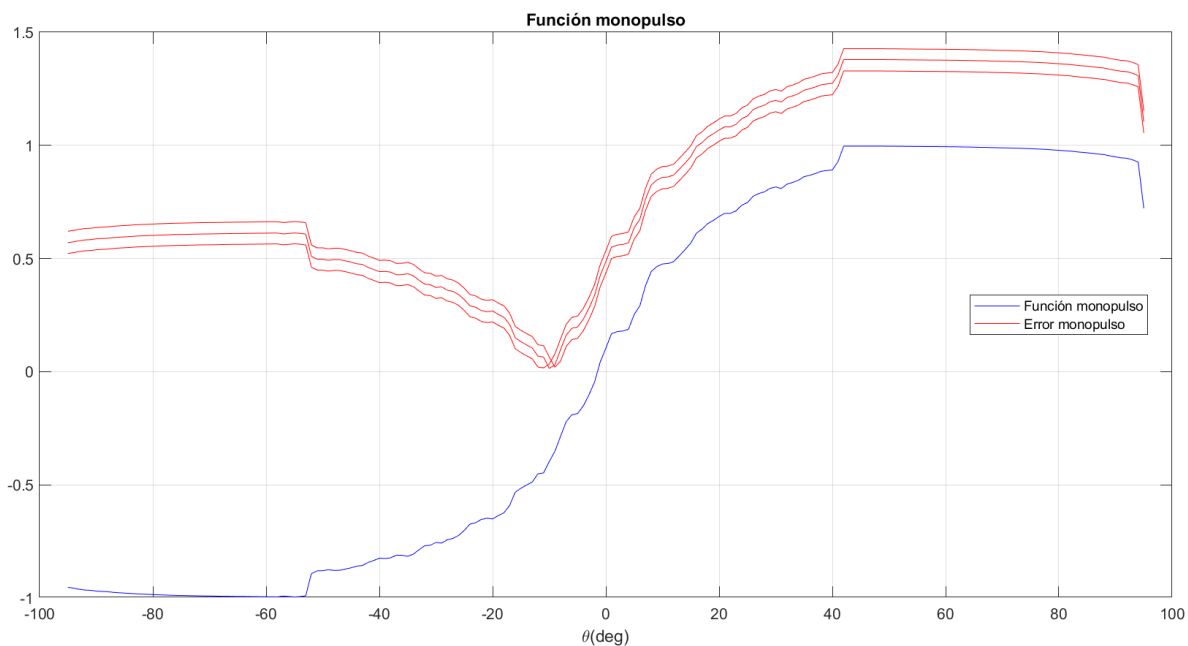


Figura 5.38. Errores monopulso para el tag 2 en el canal 2

Los valores monopulso que se han obtenido en esta ocasión también son negativos.

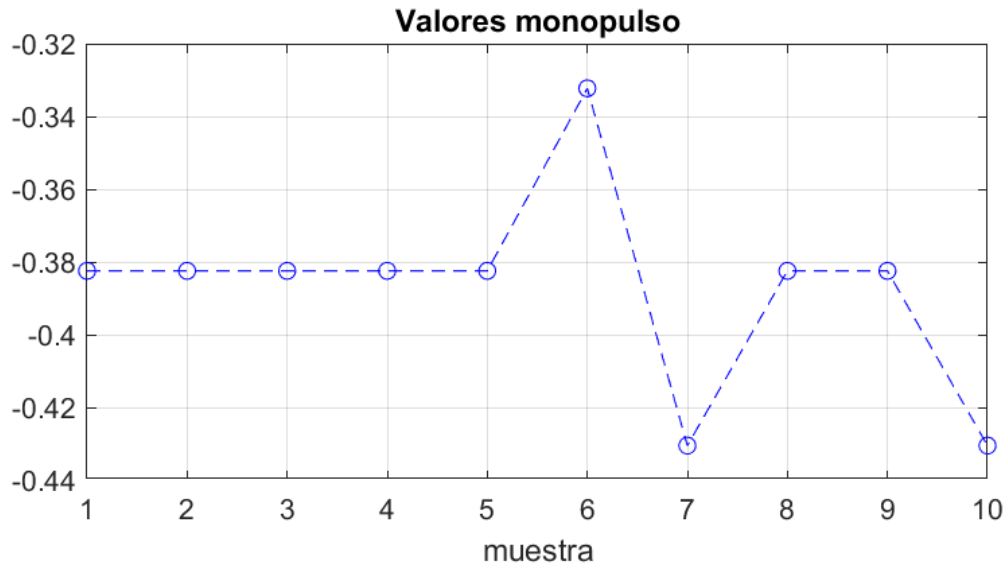


Figura 5.39. Valores monopulso para el tag 2 en el canal 2

```
valores_monopulso =
-0.3825 -0.3825 -0.3825 -0.3825 -0.3825 -0.3323 -0.4305 -0.3825 -0.3825 -0.4305
```

Figura 5.40. Valores monopulso exactos para el tag 2 en el canal 2

Por último, el pseudoespectro angular en este canal se muestra en la figura 5.41.

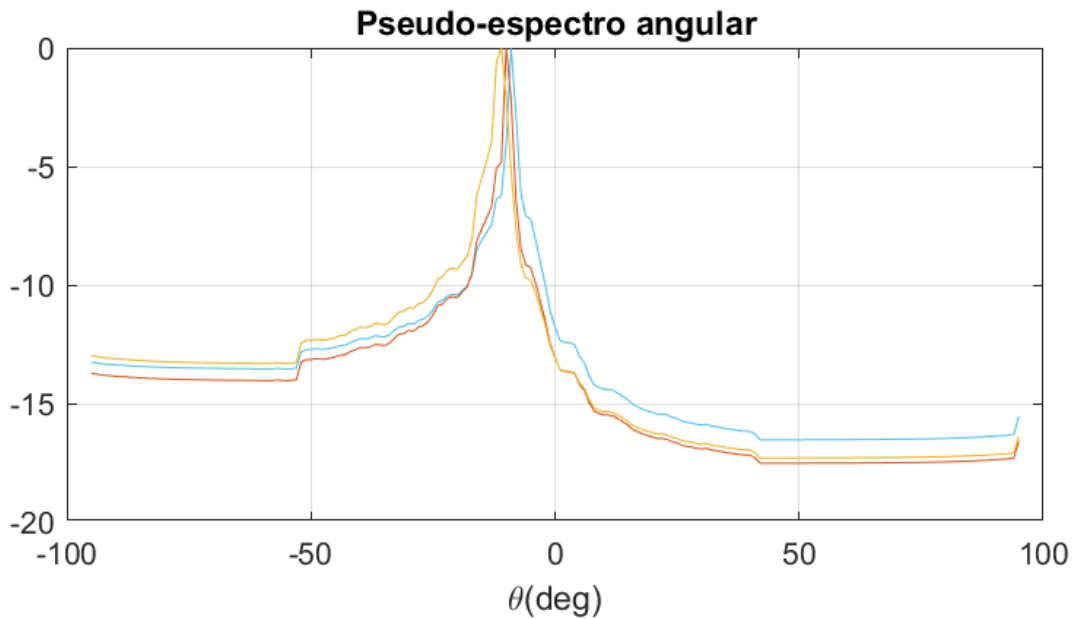


Figura 5.41. Pseudoespectros angulares para el tag 2 en el canal 2

Las DoA estimadas son, por tanto, bastante similares al valor esperado de -10° . Podemos apreciar que el error de localización es prácticamente nulo.

```
do_a_estimados =
-10  -10  -10  -10  -10  -9  -11  -10  -10  -11
```

Figura 5.42. DoA estimadas para el tag 2 en el canal 2

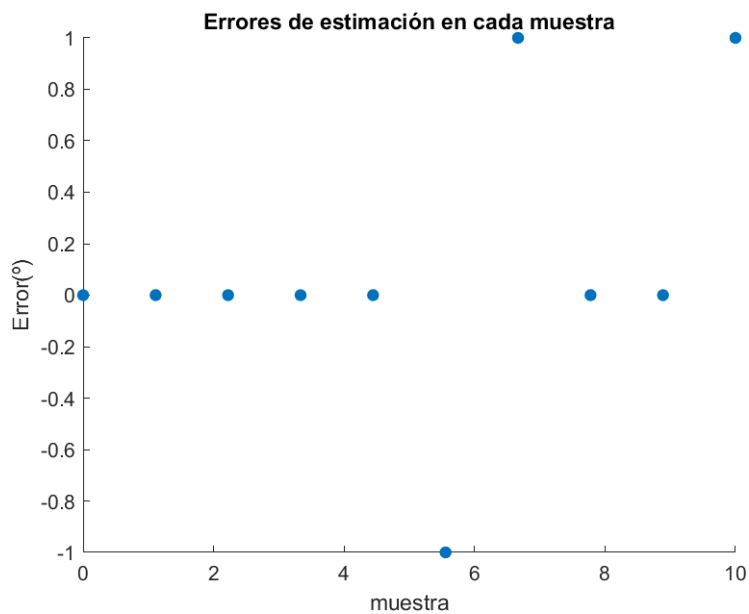


Figura 5.43. Errores de estimación para el tag 2 en el canal 2

5.2.3. Canal 3 (866,9 MHz)

Procedemos a realizar los mismos pasos para el canal de frecuencia de 866,9 MHz. Como se espera, la evolución de las RSSI a lo largo de las distintas muestras tomadas no es muy cambiante.

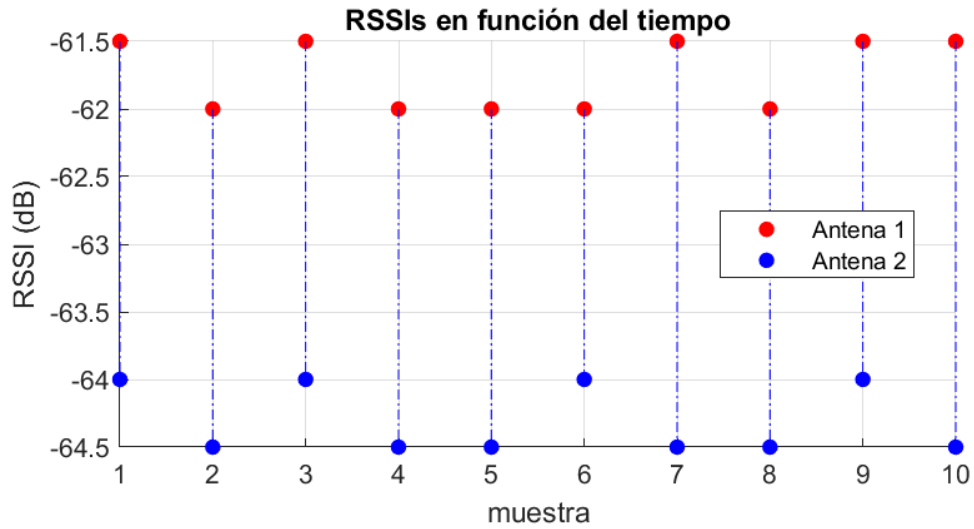


Figura 5.44. Evolución de la RSSI para el tag 2 en el canal 3

Se aprecia que la RSSI recibida por la antena 1 sigue siendo mayor que la de la antena 2.

Gracias a las funciones de error obtenidas, también se observa que la estimación de posición angular del tag será buena para este canal.

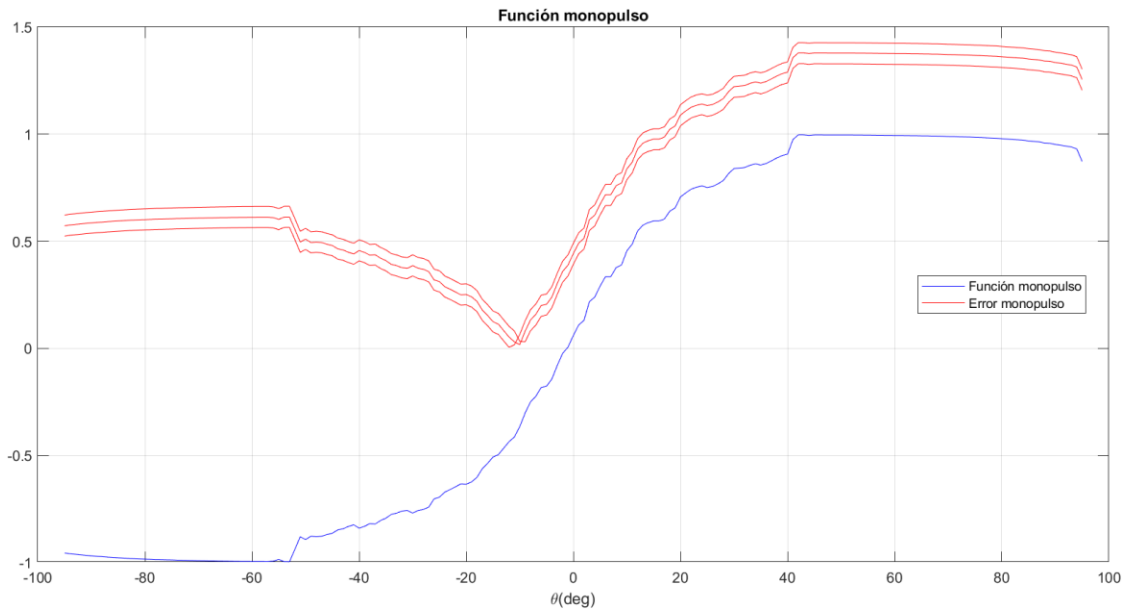


Figura 5.45. Errores monopulso para el tag 2 en el canal 3

Los errores monopulso siguen siendo, por tanto, negativos y para las 5 primeras muestras presentan el mismo valor.

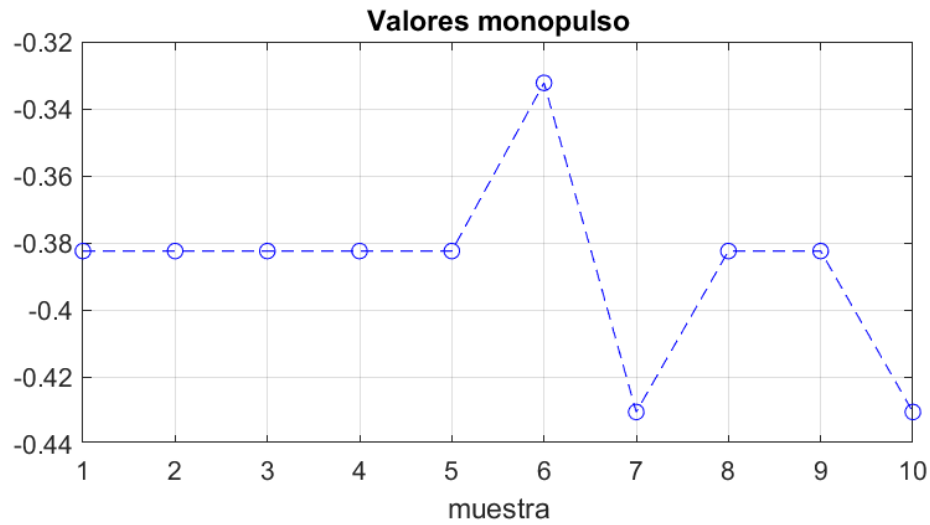


Figura 5.46. Valores monopulso para el tag 2 en el canal 3

```
valores_monopulso =
-0.3825  -0.3825  -0.3825  -0.3825  -0.3825  -0.3323  -0.4305  -0.3825  -0.3825  -0.4305
```

Figura 5.47. Valores monopulso exactos para el tag 2 en el canal 3

El pseudoespectro angular mostrará 3 valores diferentes para los tres datos distintos de valor monopulso que se han obtenido.

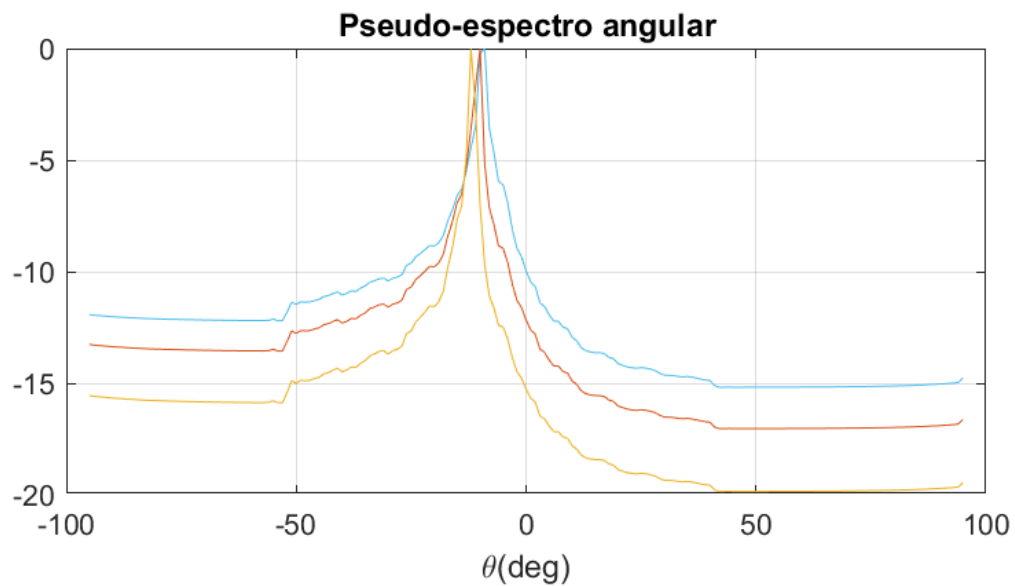


Figura 5.48. Pseudoespectros angulares para el tag 2 en el canal 3

La estimación es bastante aceptable, aunque no tan buena como para el caso del canal 2, ya que éstas presentaban un error de 1° como máximo y en este caso los errores son de hasta 2°, como podemos apreciar en la figura 5.50.

```
doa_estimados =
-10  -10  -10  -10  -10  -9  -12  -10  -10  -12
```

Figura 5.49. DoA estimadas para el tag 2 en el canal 3

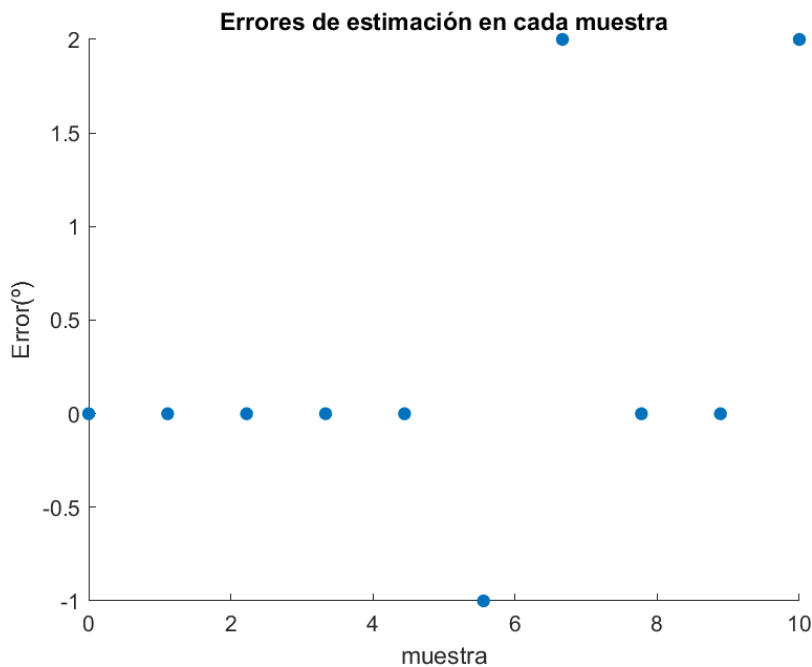


Figura 5.50. Errores de estimación para el tag 2 en el canal 3

5.2.4. Canal 4 (867,5 MHz)

Para concluir con el procesado correspondiente a este tag, falta el último de los canales de frecuencia empleados a lo largo de todo el trabajo, el de 867,5 MHz. Para este canal, se prevé que los resultados no serán tan precisos como en los anteriores, a juzgar por la evolución de la RSSI, la cual debería ser mayor para la antena 1 en todo momento. A pesar de ello, la estimación de localización seguirá siendo negativa para aquellas muestras en las que la antena 2 tiene mayor valor de RSSI, dado que a la antena 1 hay que aplicarle un factor de corrección cuyo valor es mayor que la diferencia entre las RSSI de las antenas.

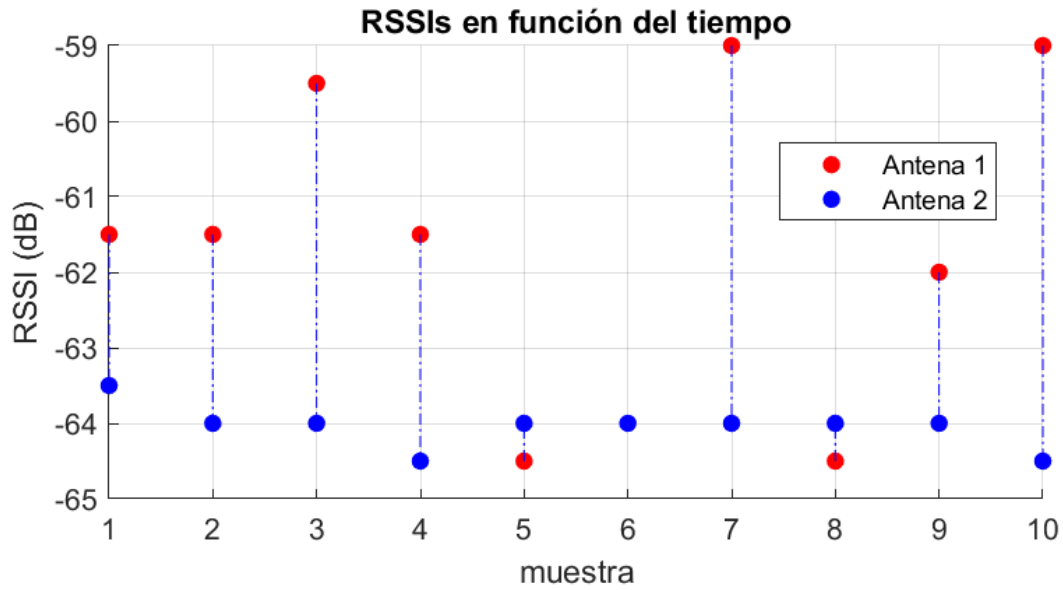


Figura 5.51. Evolución de las RSSI para el tag 2 en el canal 4

Las funciones de error, en este caso, serán diferentes para todas las muestras obtenidas, es por ello que en la representación aparecen 10 curvas diferentes.

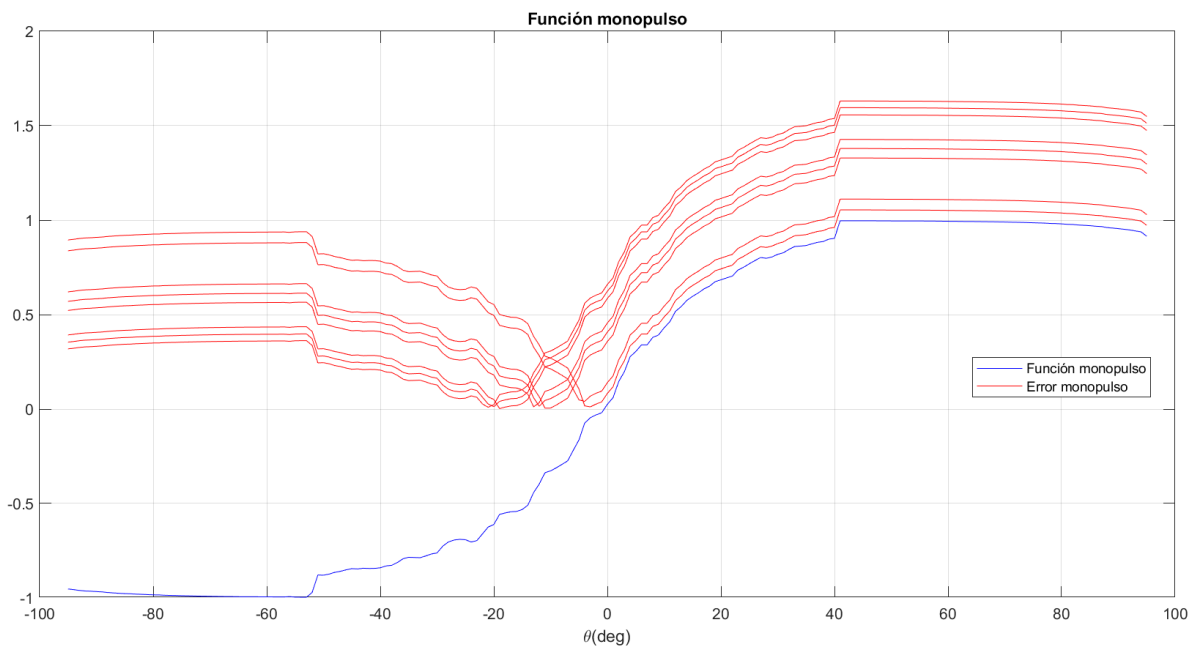


Figura 5.52. Funciones de error para el tag 2 en el canal 4

Los valores monopulso serán también diferentes para cada adquisición.

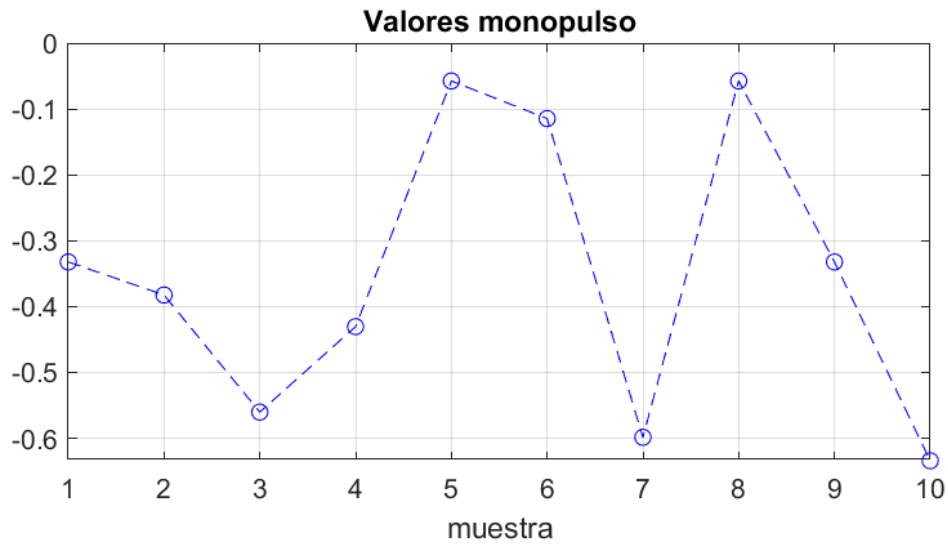


Figura 5.53. Valores monopolso para el tag 2 en el canal 4

Exactamente, los valores son los siguientes.

```
valores_monopulso =
-0.3323  -0.3825  -0.5603  -0.4305  -0.0575  -0.1146  -0.5985  -0.0575  -0.3323  -0.6342
```

Figura 5.54. Valores monopolso exactos para el tag 2 en el canal 4

Por último, las funciones de pseudoespectro angular serán tan inestables como ocurría con las funciones de error.

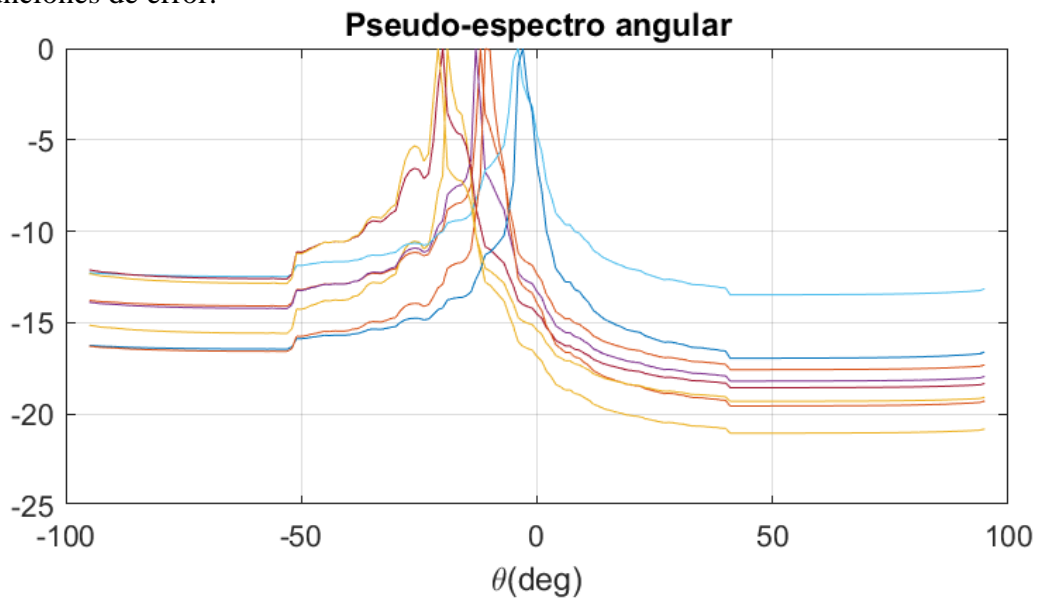


Figura 5.55. Pseudoespectros angulares para el tag 2 en el canal 4

Se aprecia que la estimación ha sido la peor en comparación al resto de canales, produciéndose un error bastante elevado para la mayoría de las muestras.

```
doa_estimados =  
-11 -12 -19 -13 -3 -4 -20 -3 -11 -21
```

Figura 5.56. DoA estimadas para el tag 2 en el canal 4

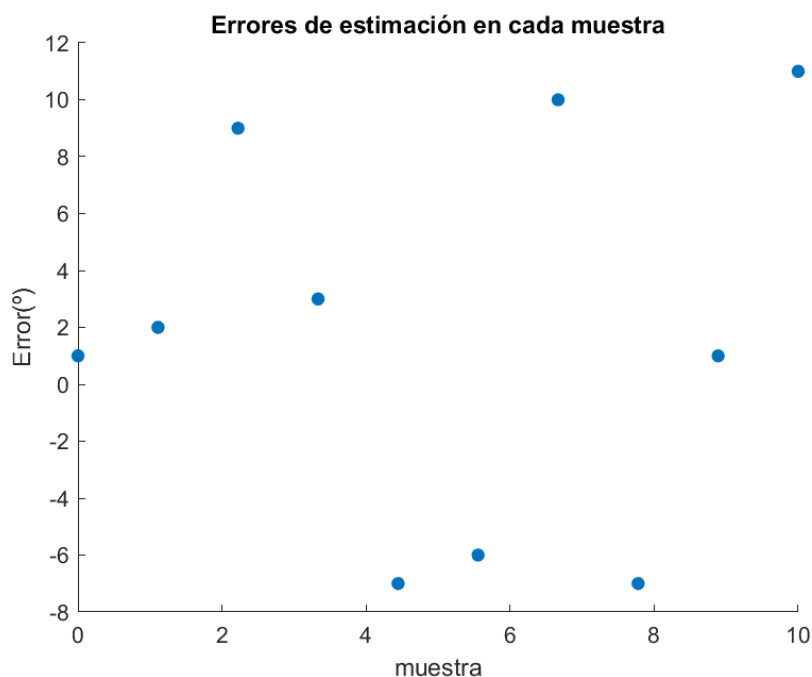


Figura 5.57. Errores de estimación para el tag 2 en el canal 4

5.3. Localización angular del tag 3

El último de los tags que queda por analizar se trata del número 3, que se posicionó en la franja de ángulos positiva, exactamente a 10°. Como es de esperar, en cada uno de los canales se recibió más potencia por la antena 2 que por la antena 1, ya que ésta es la que apunta a ángulos positivos. Procedemos pues, al análisis en cada uno de los canales.

5.3.1. Canal 1 (865,7 MHz)

La evolución de las RSSI en este canal se mantuvo bastante estable para cada una de las muestras obtenidas.

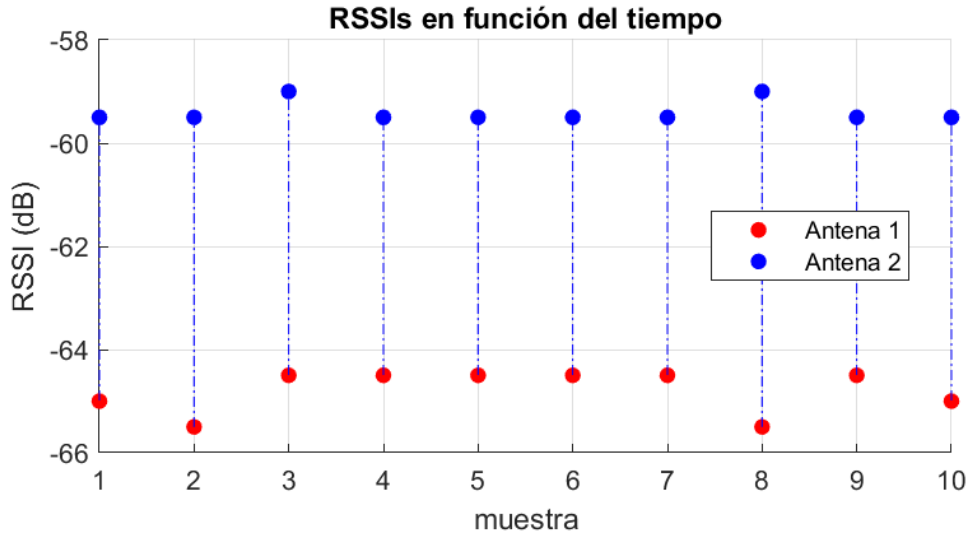


Figura 5.58. Evolución de las RSSI para el tag 3 en el canal 1

Las distintas funciones de error monopulso nos indican que la DoA del tag 3 en cada muestra es próxima a los 10° .

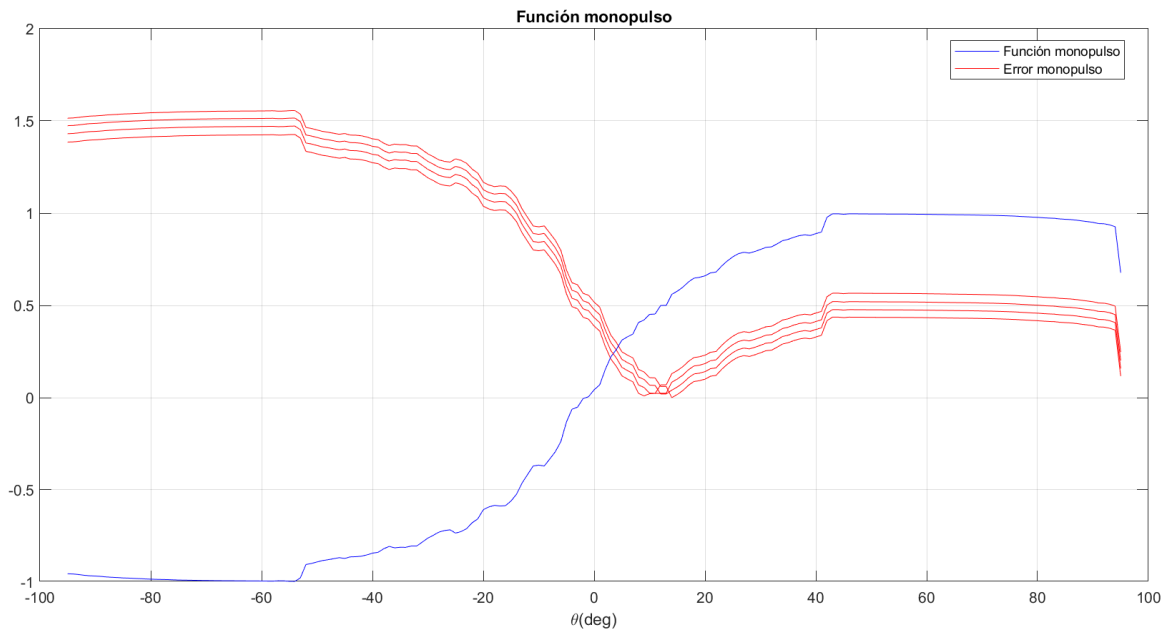


Figura 5.59. Errores monopulso para el tag 3 en el canal 1

Los distintos valores monopulso obtenidos tienen valores positivos, correspondiéndose con valores de theta positivos en la función monopulso.

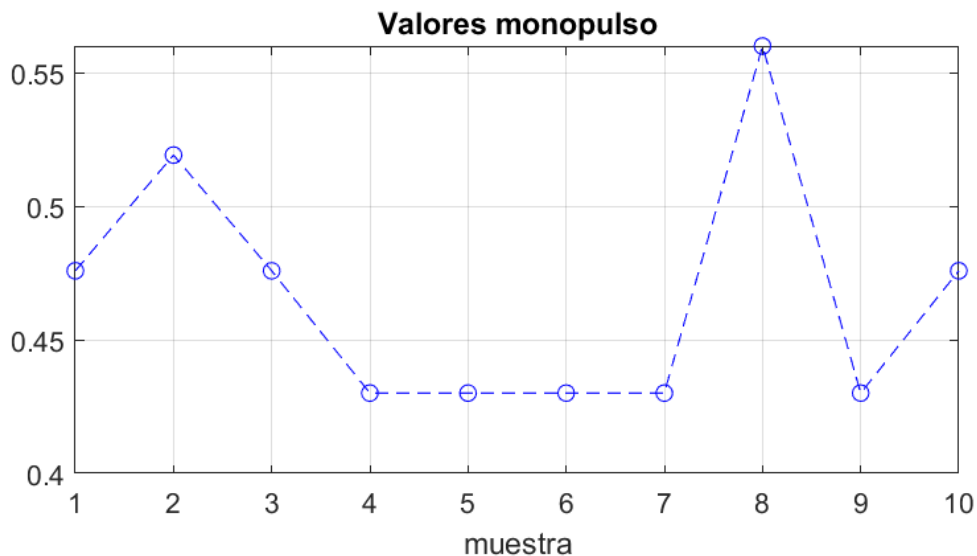


Figura 5.60. Valores monopulso para el tag 3 en el canal 1

Siendo, los valores exactos los mostrados en la figura 5.61.

valores_monopulso =

0.4762 0.5195 0.4762 0.4305 0.4305 0.4305 0.4305 0.5603 0.4305 0.4762

Figura 5.61. Valores monopulso exactos para el tag 3 en el canal 1

Los distintos pseudoespectros angulares, al igual que las funciones de error, nos indican que la localización se ha estimado en un valor próximo a 10°.

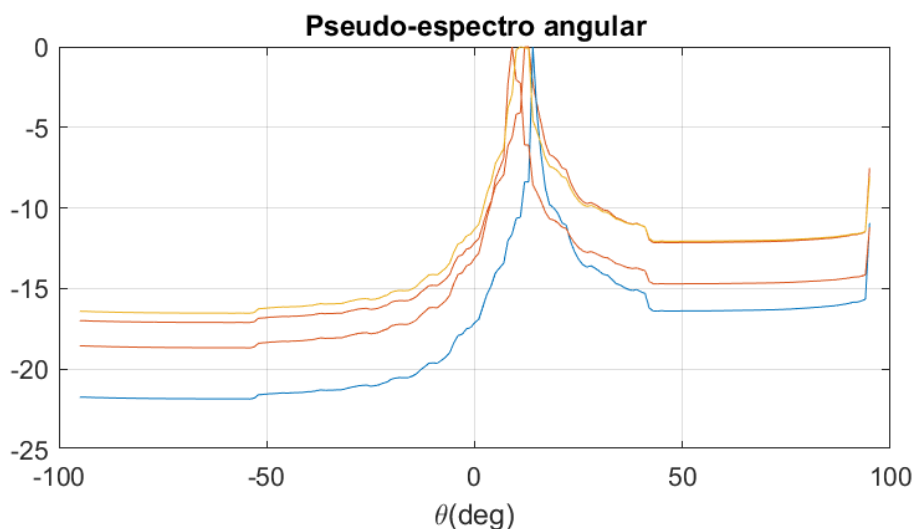


Figura 5.62. Pseudoespectros angulares para el tag 3 en el canal 1

Por último, queda mostrar cuáles han sido los ángulos estimados (DoA) en cada una de las muestras.

```
doa_estimados =  
  
    11    13    11    9    9    9    9    14    9    11
```

Figura 5.63. DoA estimadas para el tag 3 en el canal 1

Se puede concluir que la estimación coincide con lo esperado. No obstante, ha habido un cierto error cometido todas las muestras que podremos ver a continuación.

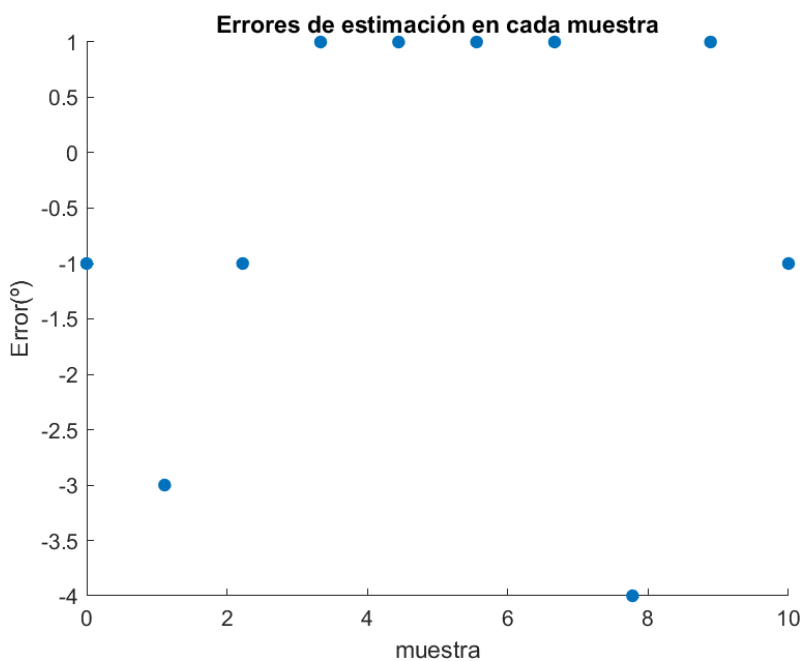


Figura 5.64. Errores cometidos para el tag 3 en el canal 1

5.3.2. Canal 2 (866,3 MHz)

En este apartado, se procede a mostrar los resultados obtenidos para el tag 3 en el canal de frecuencia número 2, correspondiente a los 866,3 MHz.

Comenzamos, como hasta ahora, con la evolución de las RSSI para cada muestra.

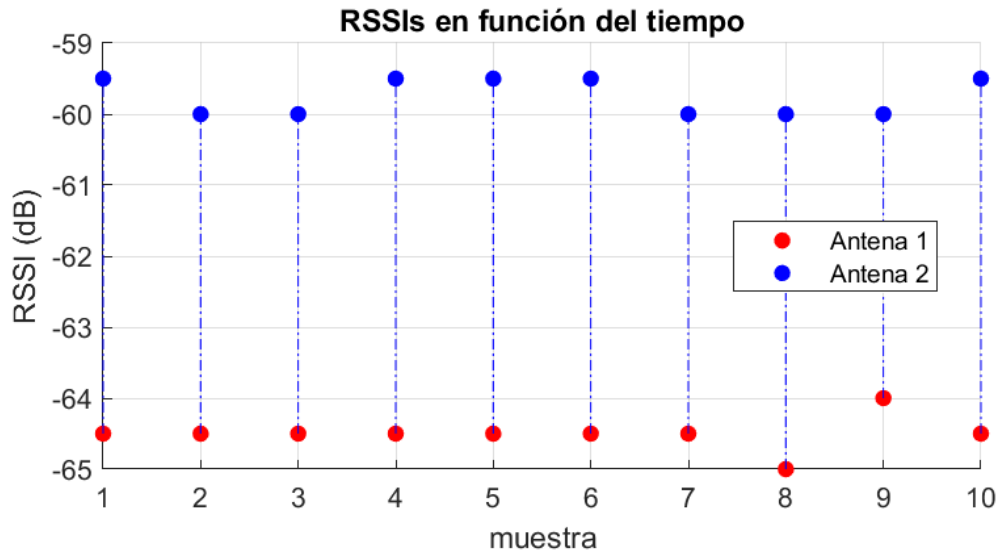


Figura 5.65. Evolución de las RSSI para el tag 3 en el canal 2

Como se espera, las distintas funciones de error monopulso no cambiarán mucho, ya que las RSSI son bastante parecidas en cada uno de los casos.

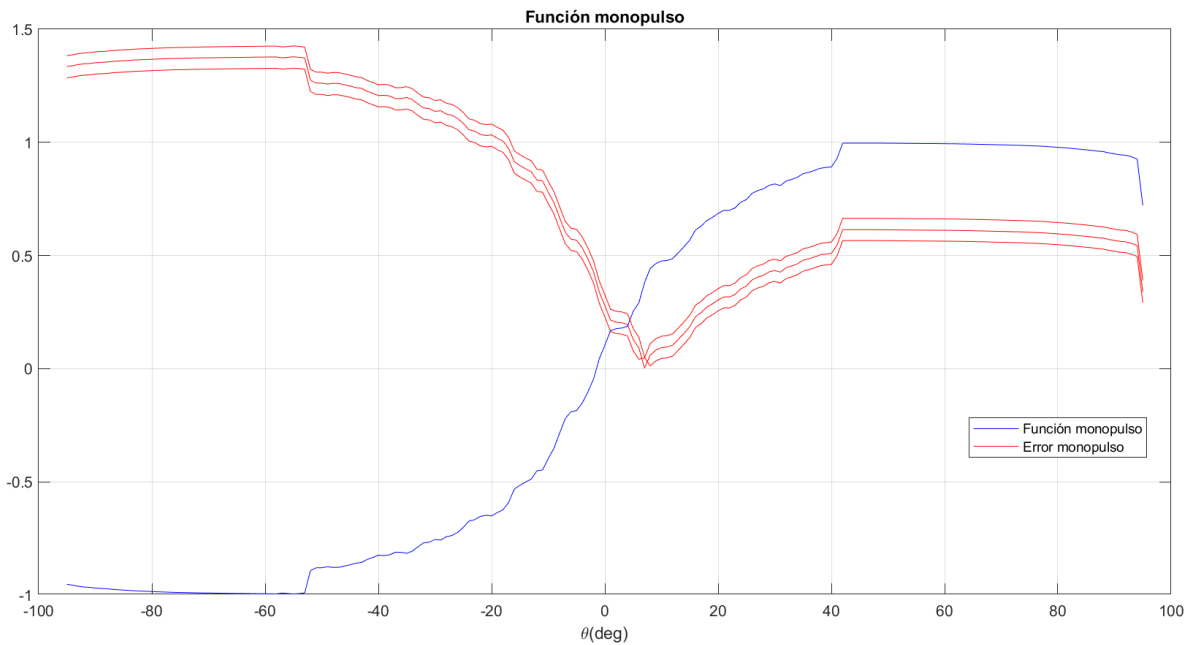


Figura 5.66. Errores monopulso para el tag 3 en el canal 2

Los valores monopulso obtenidos en esta ocasión son también positivos y tienen los siguientes valores:

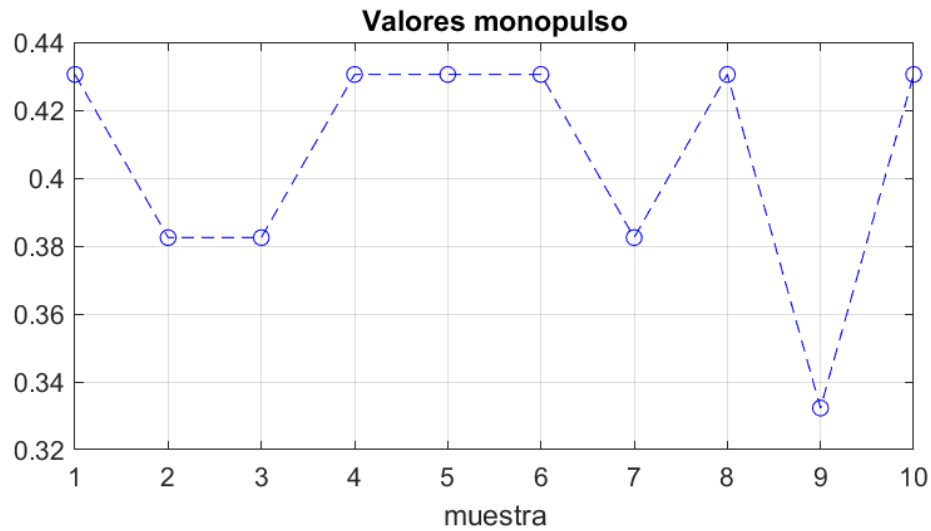


Figura 5.67. Valores monopolso para el tag 3 en el canal 2

```
valores_monopolso =
    0.4305    0.3825    0.3825    0.4305    0.4305    0.4305    0.3825    0.4305    0.3323    0.4305
    ...
```

Figura 5.68. Valores monopolso exactos para el tag 3 en el canal 2

Los pseudoespectros angulares obtenidos nos muestran que el tag se posicionó en una zona positiva cercana a los 10°.

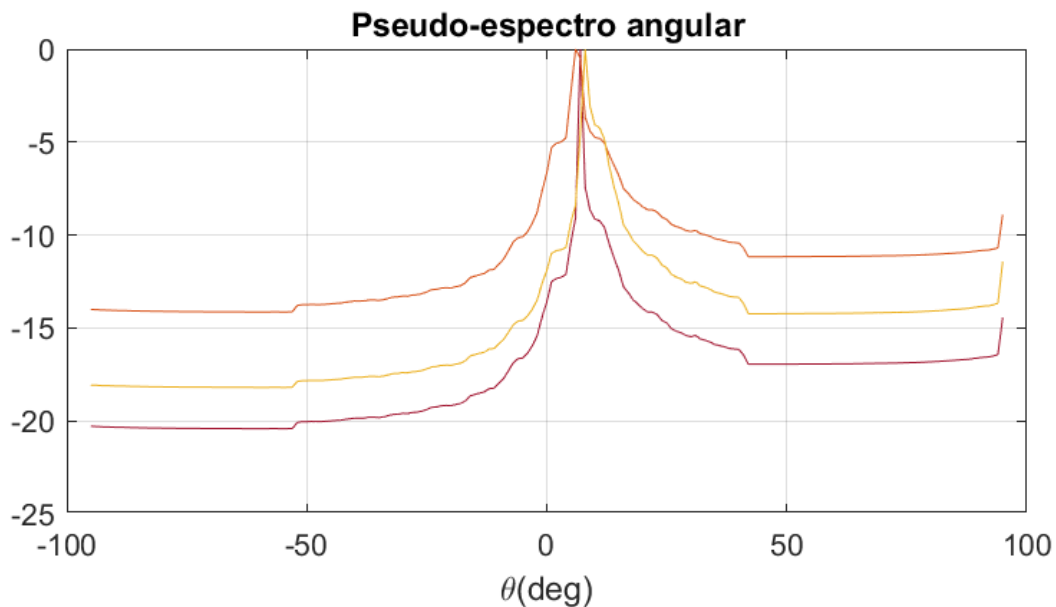


Figura 5.69. Pseudoespectros angulares para el tag 3 en el canal 2

Podremos ver más claramente cuáles son los DoA estimados gracias al Command Window de MATLAB.

doa_estimados =

8 7 7 8 8 8 7 8 6 8

Figura 5.70. DoA estimadas para el tag 3 en el canal 2

Como se puede observar, el error oscila entre los 2, 3 y 4° en el peor de los casos, pero la estimación es bastante precisa.

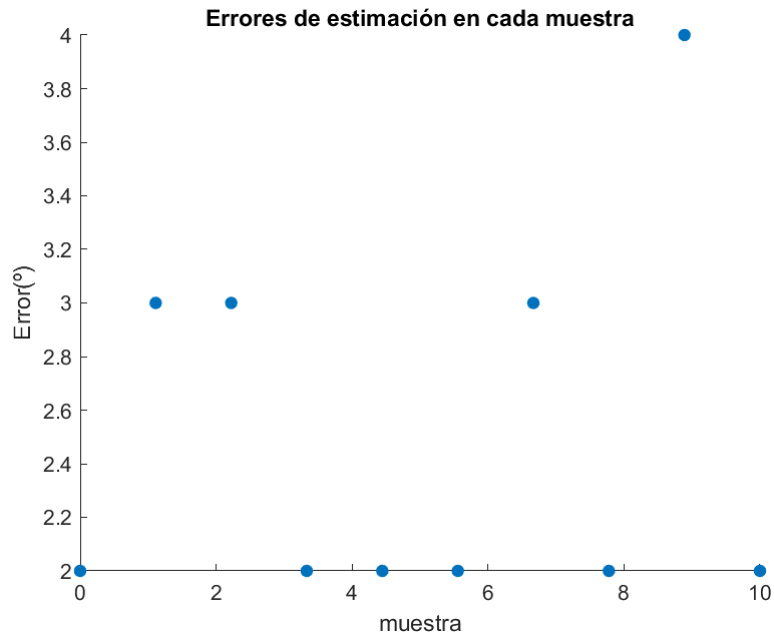


Figura 5.71. Errores de estimación para el tag 3 en el canal 2

5.3.3. Canal 3 (866,9 MHz)

El tercer canal, como ha venido sucediendo para los tags anteriores, suele tener mejores resultados que el resto. Para este tag veamos cuál ha sido la evolución de las RSSI.

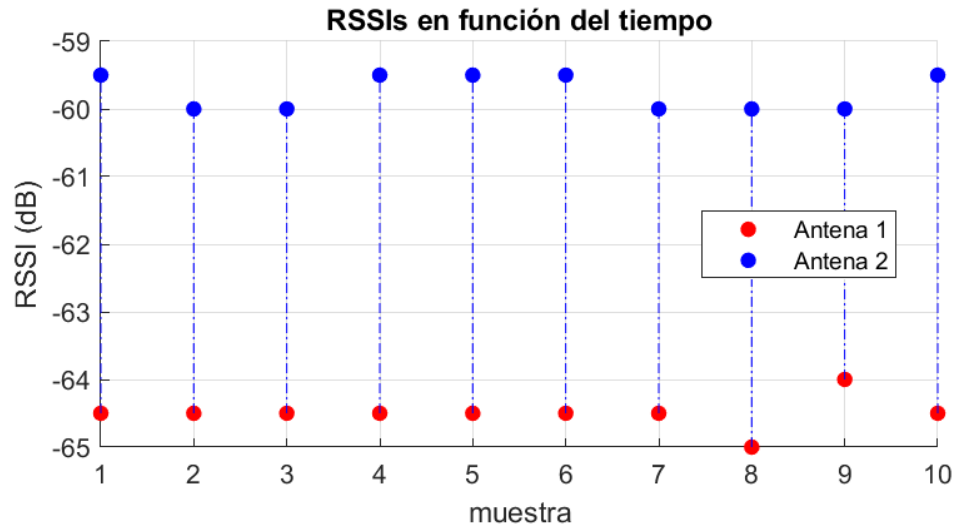


Figura 5.72. Evolución de las RSSI para el tag 3 en el canal 3

La función monopulso empleada para la estimación en este caso ha sido la correspondiente al canal 3, y se han obtenido las siguientes funciones de error monopulso.

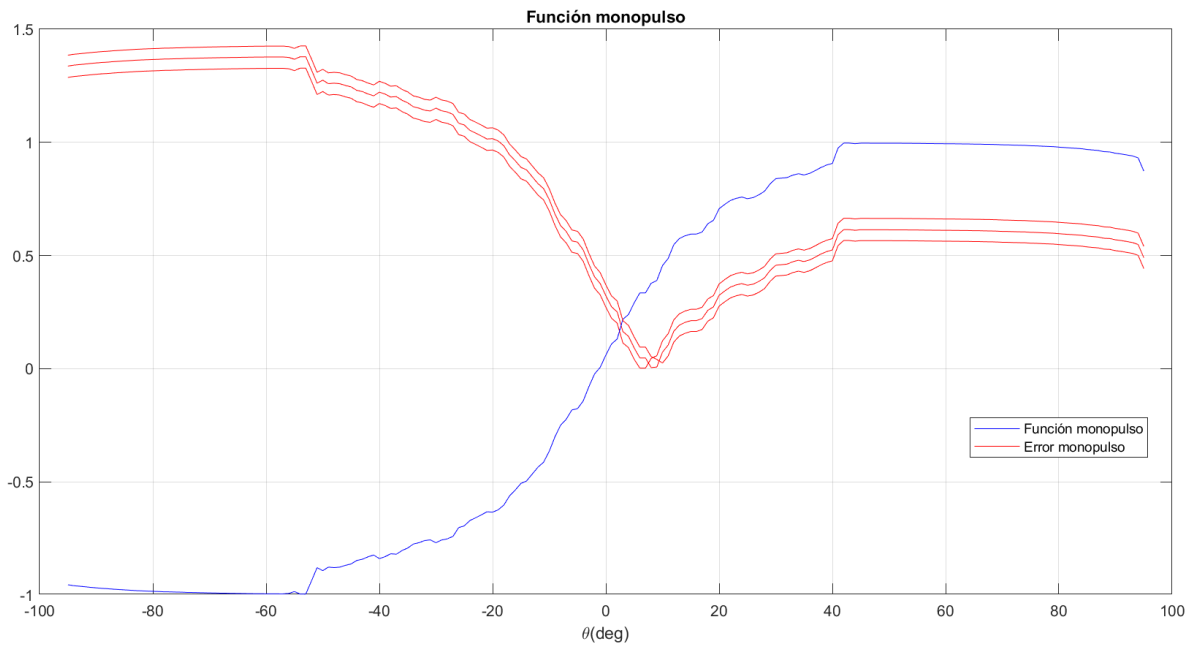


Figura 5.73. Errores monopulso para el tag 3 en el canal 3

Los valores monopulso obtenidos son los siguientes:

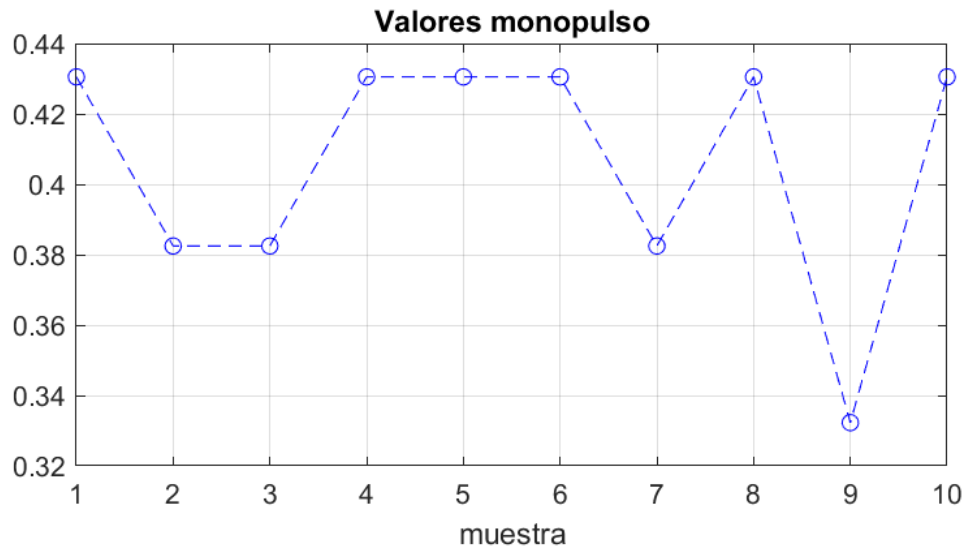


Figura 5.74. Valores monopulso para el tag 3 en el canal 3

Los valores exactos se muestran en la figura 5.75.

```
valores_monopulso =
    0.4305    0.3825    0.3825    0.4305    0.4305    0.4305    0.3825    0.4305    0.3323    0.4305
```

Figura 5.75. Valores monopulso exactos para el tag 3 en el canal 3

Finalmente, se procede a mostrar las distintas funciones de pseudoespectro angular, análogas a las funciones de error monopulso obtenidas, así como la DoA del tag 3.

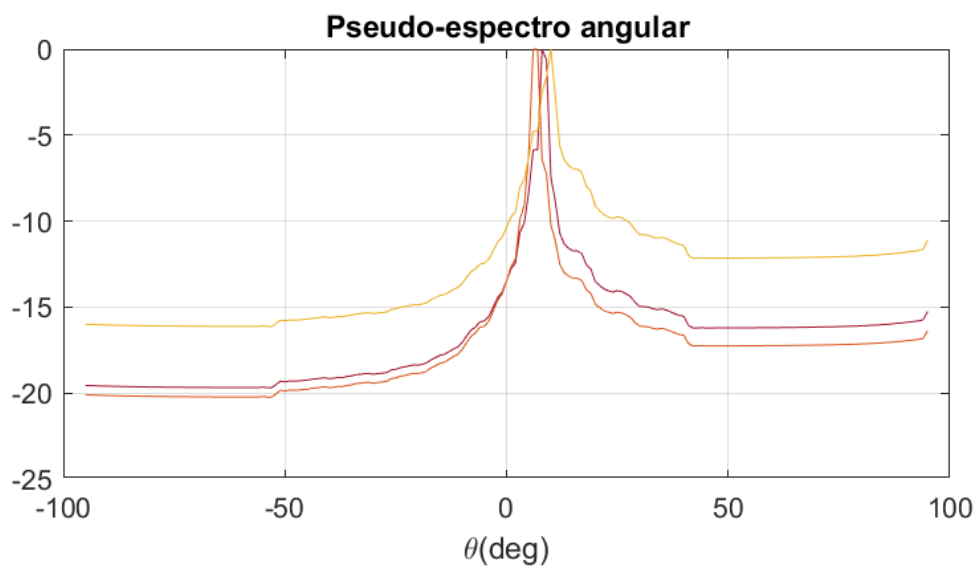


Figura 5.76. Pseudoespectros angulares para el tag 3 en el canal 3

Como se había previsto, en este canal la estimación ha sido más precisa que en cualquiera de los canales previos que se han analizado en esta sección.

doa_estimados =

10 8 8 10 10 10 8 10 6 10

Figura 5.77. DoA estimadas para el tag 3 en el canal 3

Esto queda reflejado en la representación del error cometido para cada uno de los pares de RSSI analizados, donde en la mayoría de las muestras resulta ser nulo.

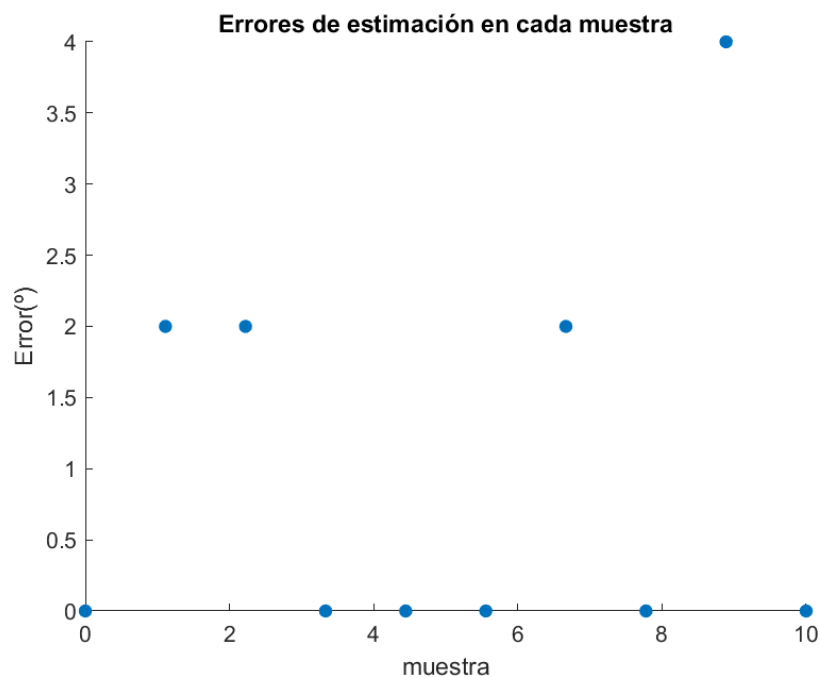


Figura 5.78. Errores de estimación para el tag 3 en el canal 3

5.3.4. Canal 4 (867,5 MHz)

Como ocurre para el caso del tag 2, el canal 4 es el más inestable de todos, presentando una evolución en la RSSI bastante variable en cada una de las distintas muestras.

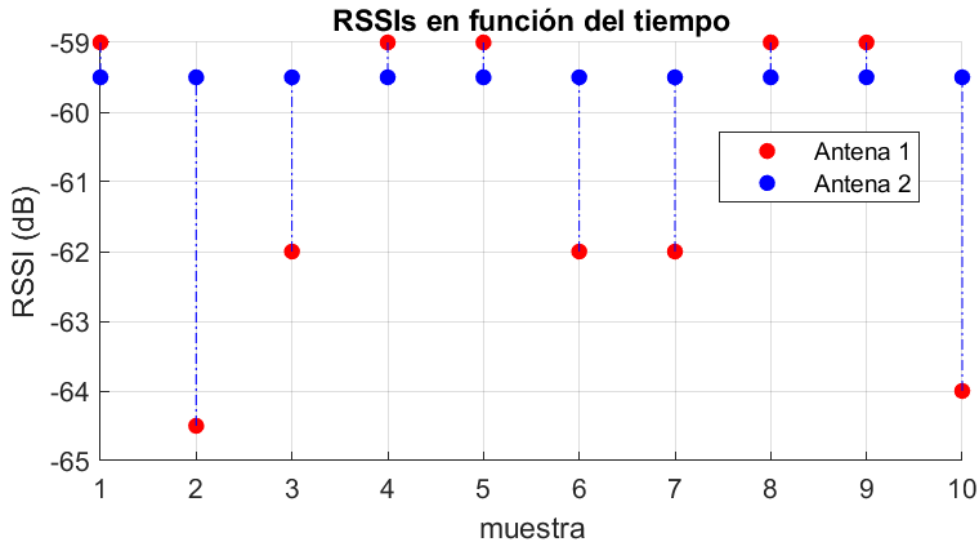


Figura 5.79. Evolución de las RSSI para el tag 3 en el canal 4

Se puede apreciar que la amplitud de la señal recibida por la antena 2 es menor en 4 de las muestras adquiridas, por tanto, al aplicar el factor de corrección digital a la antena 1, la RSSI de la antena 1 quedará muy por encima de la de la 2, provocando que la estimación resulte en ángulos negativos. Este hecho se aprecia mejor si vemos los valores monopulso obtenidos, ya que solamente serán positivos en 5 de las muestras, lo cual ya nos da un margen de error bastante alto.

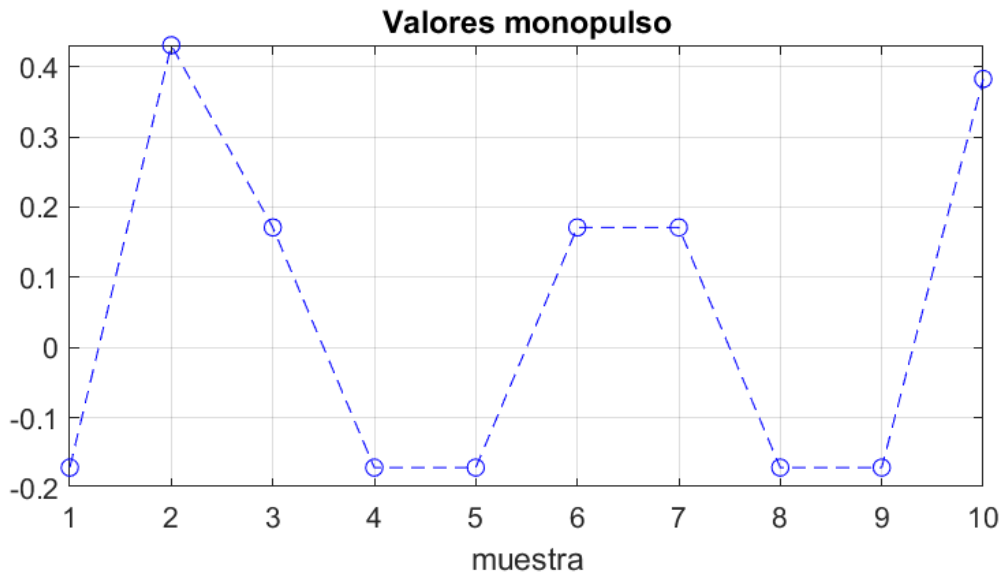


Figura 5.80. Valores monopulso para el tag 3 en el canal 4

```

valores_monopulso =
-0.1710  0.4305  0.1710  -0.1710  -0.1710  0.1710  0.1710  -0.1710  -0.1710  0.3825

```

Figura 5.81. Valores monopulso exactos para el tag 3 en el canal 4

Si ahora nos fijamos en la función monopulso empleada y las funciones de error obtenidas, vemos que, tal y como se previó el mínimo de error se encontrará en una zona angular positiva para 5 de las muestras y en una zona angular negativa para las otras 5 restantes.

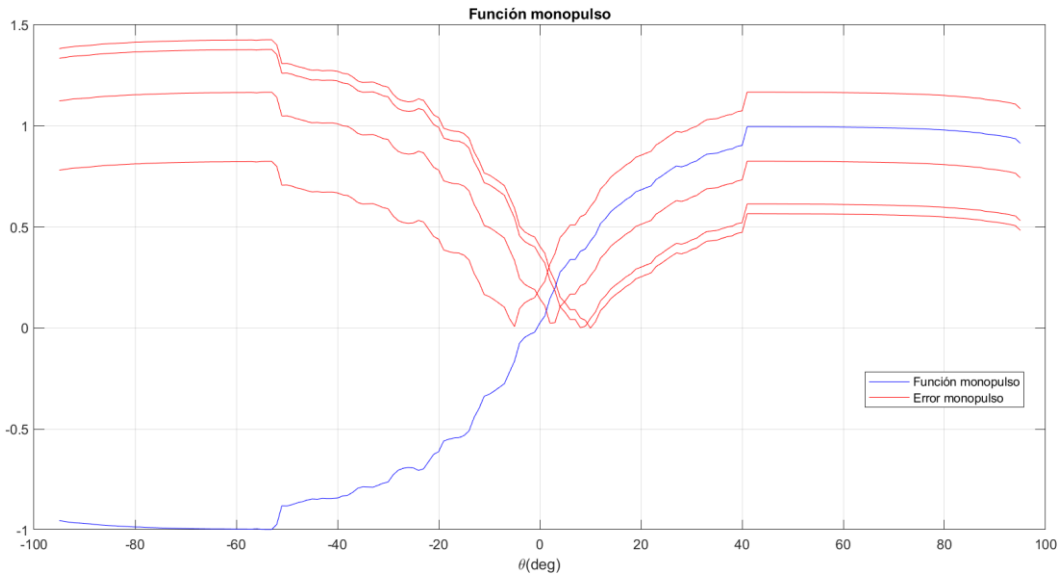


Figura 5.82. Errores monopulso para el tag 3 en el canal 4

Por último, se muestran los distintos pseudoespectros angulares, así como las DoA obtenidas para este canal.

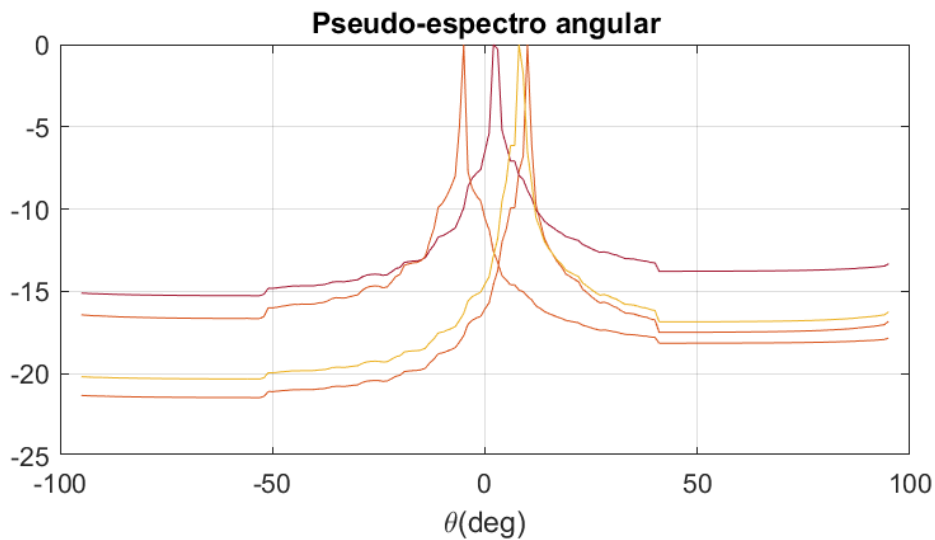


Figura 5.83. Pseudoespectros angulares para el tag 3 en el canal 4

```

doa_estimados =
    -5    10    2    -5    -5    2    2    -5    -5    8

```

Figura 5.84. DoA estimadas para el tag 3 en el canal 4

Como es de esperar, los errores cometidos en este canal son bastante elevados como podemos ver en la representación de los mismos.

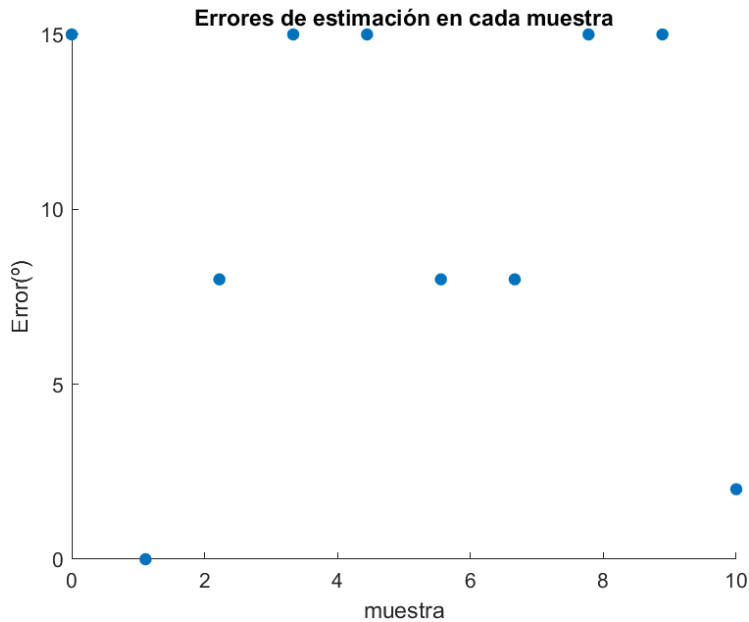


Figura 5.85. Errores de estimación para el tag 3 en el canal 4

5.4. Análisis general de los resultados

Tras haber realizado el procesado de datos y estimado la localización de los tags mediante el sistema monopulso se pueden extraer varias conclusiones.

En primer lugar, es notable que el sistema monopulso funciona correctamente para la localización de tags RFID pasivos, ya que en el capítulo anterior se comprobó sobre los valores que formaban la función monopulso, a modo de indicador. En esta ocasión se dispuso dentro de la cámara anecoica un conjunto de tags con ángulos conocidos mediante cálculos trigonométricos y se han obtenido los resultados esperados para cada uno de ellos.

Por otro lado, cabe mencionar que los resultados obtenidos han sido diferentes para cada uno de los canales y, en términos generales, la estimación realizada en los canales 2 y 3 han sido las que mejores resultados han dado, proporcionando un error bastante bajo.

Capítulo 6. Conclusiones y líneas futuras

Este proyecto ha pretendido poner de manifiesto la posibilidad de aplicar técnicas conocidas, como bien puede ser la monopulso, como solución a las tecnologías de RFID+L. Queda demostrado, en capítulos previos, que la localización de tags se realiza de forma exitosa empleando antenas de panel y readers convencionales siempre y cuando sepamos procesar correctamente los datos obtenidos y trabajemos bajo unas ciertas condiciones, concretamente, dentro del Field of View. Por otro lado, el sistema implementado tiene otras limitaciones, no solo el Field Of View, y una de las más importantes es que la estimación solo puede realizarse cuando el tag y las antenas empleadas se encuentran a la misma distancia a la que estaban cuando se adquirieron los diagramas digitales que dieron lugar a las funciones monopulso. En cualquier otra situación, los datos obtenidos pueden no corresponderse con lo esperado.

También cabe mencionar que, con estos métodos de estimación y adquisición de datos, la velocidad a la que se pueden localizar los tags es bastante rápida. El software de adquisición de datos permite modificar a gusto del usuario la velocidad a la que se envían los pulsos por ambas antenas y la obtención de datos en el entorno de trabajo es casi inmediata. La gran flexibilidad del lenguaje escogido para programar el Reader permite que podamos adaptarlo a cualquier tipo de antenas y a cualquier frecuencia de trabajo que el lector RFID permita. No obstante, debe tenerse en cuenta lo variantes que son los resultados dependiendo del canal de frecuencia en el que se trabaje, por ejemplo, teniendo en cuenta la banda europea de frecuencias, se ha concluido que los mejores canales de estimación son los canales 2 y 3.

Por esas razones, se pueden plantear una serie de líneas futuras que pueden seguir con el hilo de investigación de este trabajo. Algunas de estas líneas pueden ser el desarrollo de un software que sea capaz de adquirir los datos y procesarlos en tiempo real de tal forma que se estime dónde está el tag. Esto podría implementarse siguiendo la misma filosofía de comunicación entre herramientas de trabajo que se ha planteado en el capítulo 2. Por otro lado, análogo a lo que se ha realizado con antenas de panel y el procesado monopulso, se podrían aplicar técnicas como MUSIC, basadas en la transformada de Fourier, que permiten la estimación de la dirección de llegada, empleando para ello Leaky Wave Antennas (LWA).

Bibliografía

M. Cremer, A. Pervez, U. Dettmar, C. Hudasch, R. Kronberger and R. Lerche, "Multichannel digital transmit beamforming experiments for RFID tag localization," 2015 International EURASIP Workshop on RFID Technology (EURFID), Rosenheim, 2015, pp. 126-131, doi: 10.1109/EURFID.2015.7332397.

C. Alcaraz, J. V. Balbastre and F. Vega, "Monopulse RFID reader for enhanced intelligent transportation systems applications," 2015 IEEE Brasil RFID, Sao Paulo, 2015, pp. 1-4, doi: 10.1109/BrasilRFID.2015.7523834.

J. Lechner, A. Ascher, S. Nosovic and W. A. Guenther, "Concept for an intelligent UHF RFID reader according to the Ideas of Industry 4.0," Smart SysTech 2016; European Conference on Smart Objects, Systems and Technologies, Duisburg, Germany, 2016, pp. 15.

Qiu, L., Liang, X. & Huang, Z. PATL: A RFID Tag Localization based on Phased Array Antenna. Sci Rep 7, 44183 (2017). ([enlace de acceso, julio 2021](#))

A. Ascher, J. Lechner, S. Nosovic, P. Eschlwech and E. Biebl, "3D localization of passive UHF RFID transponders using direction of arrival and distance estimation techniques," 2017 IEEE 2nd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC), Chongqing, 2017, pp. 1373-1379, doi: 10.1109/IAEAC.2017.8054239.

T. Sakogawa, K. Aoki and F. Kuroki, "A technique to narrow down radiation patterns of broad beam antenna operationally and its application to security gate to prevent shoplifting based on monopulse system," 2017 11th European Conference on Antennas and Propagation (EUCAP), Paris, 2017, pp. 1945-1948, doi: 10.23919/EuCAP.2017.7928235.

Laker, Darrell, Galal Hassan, Russel Dobson and Ruiheng Wu. "RFID localisation using special antenna technique," International Journal of Radio Frequency Identification Technology and Applications, 5(1):775, pp. 77-92, January 2018.

Laker, Darrell and Hassan, Galal and Dobson, Russel and Wu, Ruiheng (2018) "RFID localization using special antenna technique". International Journal of Radio Frequency Identification Technology and Applications, 5 (1). ISSN 1745-3216

Ma, L.; Liu, M.; Wang, H.; Yang, Y.; Wang, N.; Zhang, Y. WallSense: "Device-Free Indoor Localization Using Wall-Mounted UHF RFID Tags". Sensors 2019, 19, 68.

Mohamedatni, Y. et al. "New methodology for the short range localization of UHF RFID tags using a linear uniform array." I. J. RF Technol.: Res. and Appl. 10 (2019): 9-26.

J. L. Gómez-Tornero, D. Cañete-Rebenaque, J. A. López-Pastor and A. S. Martínez-

Sala, "Hybrid Analog-Digital Processing System for Amplitude-Monopulse RSSI-Based MiMo WiFi Direction-of-Arrival Estimation," in IEEE Journal of Selected Topics in Signal Processing, vol. 12, no. 3, pp. 529-540, June 2018, doi: 10.1109/JSTSP.2018.2827701.

J. A. López-Pastor, A. Gómez-Alcaraz, D. Cañete-Rebenaque, A. S. Martínez-Sala and J. L. Gómez-Tornero, "Near-Field Monopulse DoA Estimation for Angle-Sensitive Proximity WiFi Readers," in IEEE Access, vol. 7, pp. 88450-88460, 2019, doi: 10.1109/ACCESS.2019.2925739.

Anexo I. Clases Java

I.I. MainServidor

```
package sockets.servidor;

import java.io.IOException;

import com.example.sdksamples.SampleProperties;
import com.mathworks.engine.EngineException;
import com.mathworks.engine.MatlabEngine;

import sockets.servidor.Servidor;

//Clase principal que hará uso del servidor
public class MainServidor
{
    public static void main(String[] args) throws IOException, EngineException,
    InterruptedException
    {
        String hostname = System.getProperty(SampleProperties.hostname);
        String[] engines = MatlabEngine.findMatlab();
        MatlabEngine matEng = MatlabEngine.connectMatlab(engines[0]);
        Servidor serv = new Servidor(); //Se crea el servidor

        System.out.println("Iniciando servidor\n");
        serv.startServer(matEng, hostname); //Se inicia el servidor
    }
}
```

I.II. Conexion

```
package sockets.conexion;

import java.io.DataOutputStream;
import java.io.IOException;
import java.net.ServerSocket;
import java.net.Socket;

public class Conexion
{
    private final int PUERTO = 30000; //Puerto para la conexión
    private final String HOST = "localhost"; //Host para la conexión
    protected String mensajeServidor; //Mensajes entrantes (recibidos) en el
servidor
    protected ServerSocket ss; //Socket del servidor
    protected Socket cs; //Socket del cliente
    protected DataOutputStream salidaServidor, salidaCliente; //Flujo de datos
de salida
}
```

```

public Conexion(String tipo) throws IOException //Constructor
{
    if(tipo.equalsIgnoreCase("servidor"))
    {
        ss = new ServerSocket(PUERTO); //Se crea el socket para el servidor
en puerto
        cs = new Socket(); //Socket para el cliente
    }
    else
    {
        cs = new Socket(HOST, PUERTO); //Socket para el cliente en localhost
en puerto 1234
    }
}
}

```

I.III. Servidor

```

package sockets.servidor;

import java.io.BufferedReader;

import com.example.sdksamples.Config;
import com.mathworks.engine.MatlabEngine;
import java.io.DataOutputStream;
import java.io.IOException;
import java.io.InputStreamReader;
import sockets.conexion.Conexion;

public class Servidor extends Conexion //Se hereda de conexión para hacer uso de
los sockets y demás
{
    public Servidor() throws IOException{super("servidor");} //Se usa el
constructor para servidor de Conexion

    public void startServer(MatlabEngine matEng, String hostname) //Método para
iniciar el servidor
    {
        try
        {
            int contador = 1;

            System.out.println("Esperando..."); //Esperando conexión

            cs = ss.accept(); //Accept comienza el socket y espera una conexión
desde un cliente

            System.out.println("Cliente en línea");

            //Se obtiene el flujo de salida del cliente para enviarle mensajes
salidaCliente = new DataOutputStream(cs.getOutputStream());

```

```

//Se le envía un mensaje al cliente usando su flujo de salida
salidaCliente.writeUTF("Petición recibida y aceptada");

//Se obtiene el flujo entrante desde el cliente
BufferedReader entrada = new BufferedReader(new
InputStreamReader(cs.getInputStream()));

while((mensajeServidor = entrada.readLine()) != null) //Mientras
haya mensajes desde el cliente
{
    //Se muestra por pantalla el mensaje recibido
    matEng.eval("pause('on')");
    String angulo = String.valueOf(contador);
    if (angulo.equals(mensajeServidor)) {
        System.out.println("Estamos en el ángulo " + contador);
        Config toma_datos = new Config();
        toma_datos.main(contador, matEng, hostname);
        matEng.eval("contador =" + angulo);
        matEng.eval("pause('off')");
        contador ++;
    }
}

System.out.println("Fin de la conexión");

ss.close();//Se finaliza la conexión con el cliente
}
catch (Exception e)
{
    System.out.println(e.getMessage());
}
}
}

```

I.IV. Config

```

package com.example.sdksamples;
import com.mathworks.engine.MatlabEngine;

public class Config {

    public static void main(int angle, MatlabEngine matEng, String hostname)
throws Exception {
        // TODO Auto-generated method stub

        if (hostname == null) {
            throw new Exception("Must specify the '"
                + SampleProperties.hostname + "' property");
        }
    }
}

```

```

    int [] puertos = new int [2];
    puertos[0] = 1;
    puertos[1]=1;
    double [] canales = new double [4];
    canales[0]=865.7;
    canales[1]=866.3;
    canales[2]=866.9;
    canales[3]=867.5;
    double tiempo_ms= 400;

    Start start = new Start();
    start.angle = angle;
    start.puertos=puertos;
    start.canales=canales;
    start.tiempo_ms= tiempo_ms;
    start.matEng=matEng;
    start.main(hostname);
}
}

```

I.V. Start

```

package com.example.sdksamples;
import com.impinj.octane.TagReportListener;
import com.mathworks.engine.MatlabEngine;

public class Start {
    public static int angle;
    public static double [] canales;
    public static int [] puertos;
    public static double tiempo_ms;
    public static double [][] datos;
    public static String hostname;
    public static MatlabEngine matEng;

    public static void main(String args) throws Exception {
        // TODO Auto-generated method stub

        // String hostname = getHostname();
        String hostname = args;

        HelloOctaneSdk hello = new HelloOctaneSdk();

        TagReportListenerImplementation t = new
TagReportListenerImplementation();
        int cuenta = t.cuenta;
    }
}

```

```

    if (puertos[0]==1) {
        if (canales[0]==865.7) {
            hello.main(hostname,angle, tiempo_ms, puertos[0], canales[0],t);
        }

        if(canales[1]==866.3) {
            hello.main(hostname,angle, tiempo_ms, puertos[0], canales[1],t);
        }
        if(canales[2]==866.9) {
            hello.main(hostname,angle, tiempo_ms, puertos[0], canales[2],t);
        }
        if(canales[3]==867.5) {
            hello.main(hostname,angle, tiempo_ms, puertos[0], canales[3],t);
        }
    }

    if (puertos[1]==1) {
        if (canales[0]==865.7) {
            hello.main(hostname,angle, tiempo_ms, puertos[1]+1,
canales[0],t);
        }
        if(canales[1]==866.3) {
            hello.main(hostname,angle, tiempo_ms, puertos[1]+1,
canales[1],t);
        }
        if(canales[2]==866.9) {
            hello.main(hostname,angle, tiempo_ms, puertos[1]+1,
canales[2],t);
        }
        if(canales[3]==867.5) {
            hello.main(hostname,angle, tiempo_ms, puertos[1]+1,
canales[3],t);
        }
    }

    int size = 0;
    for (int i = 1; i<=t.puerto.length; i++) {if(t.puerto[i-1]==1 ||
t.puerto[i-1]==2) {size++;}}

    short resultado_puertos[] = new short[size];
    String resultado_IDS[] = new String[size];
    double resultado_RSSI[] = new double[size];
    double resultado_f[] = new double [size];

    for(int i =1;i<=size;i++) {resultado_puertos[i-1]=t.puerto[i-1];}
    for(int i=1;i<=size;i++) {resultado_IDS[i-1]=t.IDS[i-1];}
    for(int i=1;i<=size;i++) {resultado_RSSI[i-1]=t.rssi[i-1];}
    for(int i=1;i<=size;i++) {resultado_f[i-1]=t.f[i-1];}

    short resultado_puertos_canal1[] = new short[cuenta];
    String resultado_IDS_canal1[] = new String[cuenta];
    double resultado_RSSI_canal1[] = new double[cuenta];
    double resultado_f_canal1[] = new double [cuenta];

```

```

short resultado_puertos_canal12[] = new short[cuenta];
String resultado_IDS_canal12[] = new String[cuenta];
double resultado_RSSI_canal12[] = new double[cuenta];
double resultado_f_canal12[] = new double [cuenta];

short resultado_puertos_canal2[] = new short[cuenta];
String resultado_IDS_canal2[] = new String[cuenta];
double resultado_RSSI_canal2[] = new double[cuenta];
double resultado_f_canal2[] = new double [cuenta];

short resultado_puertos_canal22[] = new short[cuenta];
String resultado_IDS_canal22[] = new String[cuenta];
double resultado_RSSI_canal22[] = new double[cuenta];
double resultado_f_canal22[] = new double [cuenta];

short resultado_puertos_canal3[] = new short[cuenta];
String resultado_IDS_canal3[] = new String[cuenta];
double resultado_RSSI_canal3[] = new double[cuenta];
double resultado_f_canal3[] = new double [cuenta];

short resultado_puertos_canal32[] = new short[cuenta];
String resultado_IDS_canal32[] = new String[cuenta];
double resultado_RSSI_canal32[] = new double[cuenta];
double resultado_f_canal32[] = new double [cuenta];

short resultado_puertos_canal4[] = new short[cuenta];
String resultado_IDS_canal4[] = new String[cuenta];
double resultado_RSSI_canal4[] = new double[cuenta];
double resultado_f_canal4[] = new double [cuenta];

short resultado_puertos_canal42[] = new short[cuenta];
String resultado_IDS_canal42[] = new String[cuenta];
double resultado_RSSI_canal42[] = new double[cuenta];
double resultado_f_canal42[] = new double [cuenta];

int cont1=0;
int cont12=0;
int cont2=0;
int cont22=0;
int cont3=0;
int cont32=0;
int cont4=0;
int cont42=0;

for(int i = 1; i<=size;i++) {
    if(resultado_f[i-1]==865.7 && resultado_puertos[i-1]==1) {
        if(cont1<cuenta) {
            resultado_puertos_canal1[cont1] = resultado_puertos[i-1];
            resultado_IDS_canal1[cont1]=resultado_IDS[i-1];
            resultado_RSSI_canal1[cont1]=resultado_RSSI[i-1];
            resultado_f_canal1[cont1]=resultado_f[i-1];
            cont1++;}}

```



```

    if(resultado_f[i-1]==865.7 && resultado_puertos[i-1]==2 ) {
        if(cont12<cuenta) {
            resultado_puertos_canal12[cont12] = resultado_puertos[i-1];
            resultado_IDS_canal12[cont12]=resultado_IDS[i-1];
            resultado_RSSI_canal12[cont12]=resultado_RSSI[i-1];
            resultado_f_canal12[cont12]=resultado_f[i-1];
            cont12++;
        }
    }
}

for(int i=1;i<=size;i++) {
    if(resultado_f[i-1]==866.3 && resultado_puertos[i-1]==1) {
        if(cont2<cuenta) {
            resultado_puertos_canal2[cont2] = resultado_puertos[i-1];
            resultado_IDS_canal2[cont2]=resultado_IDS[i-1];
            resultado_RSSI_canal2[cont2]=resultado_RSSI[i-1];
            resultado_f_canal2[cont2]=resultado_f[i-1];
            cont2++;
        }
    }

    if(resultado_f[i-1]==866.3 && resultado_puertos[i-1]==2 ) {
        if(cont22<cuenta) {
            resultado_puertos_canal22[cont22] = resultado_puertos[i-1];
            resultado_IDS_canal22[cont22]=resultado_IDS[i-1];
            resultado_RSSI_canal22[cont22]=resultado_RSSI[i-1];
            resultado_f_canal22[cont22]=resultado_f[i-1];
            cont22++;
        }
    }
}

for (int i=1;i<=size;i++) {
    if(resultado_f[i-1]==866.9 && resultado_puertos[i-1]==1) {
        if(cont3<cuenta) {
            resultado_puertos_canal3[cont3] = resultado_puertos[i-1];
            resultado_IDS_canal3[cont3]=resultado_IDS[i-1];
            resultado_RSSI_canal3[cont3]=resultado_RSSI[i-1];
            resultado_f_canal3[cont3]=resultado_f[i-1];
            cont3++;
        }
    }

    if(resultado_f[i-1]==866.9 && resultado_puertos[i-1]==2 ) {
        if(cont32<cuenta) {
            resultado_puertos_canal32[cont32] = resultado_puertos[i-1];
            resultado_IDS_canal32[cont32]=resultado_IDS[i-1];
            resultado_RSSI_canal32[cont32]=resultado_RSSI[i-1];
            resultado_f_canal32[cont32]=resultado_f[i-1];
            cont32++;
        }
    }
}

```

```

    }
}

for (int i=1;i<=size;i++) {
    if(resultado_f[i-1]==867.5 && resultado_puertos[i-1]==1) {
        if(cont4<cuenta) {
            resultado_puertos_canal4[cont4] = resultado_puertos[i-1];
            resultado_IDS_canal4[cont4]=resultado_IDS[i-1];
            resultado_RSSI_canal4[cont4]=resultado_RSSI[i-1];
            resultado_f_canal4[cont4]=resultado_f[i-1];
            cont4++;
        }
    }

    if(resultado_f[i-1]==867.5 && resultado_puertos[i-1]==2 ) {
        if(cont42<cuenta) {
            resultado_puertos_canal42[cont42] = resultado_puertos[i-1];
            resultado_IDS_canal42[cont42]=resultado_IDS[i-1];
            resultado_RSSI_canal42[cont42]=resultado_RSSI[i-1];
            resultado_f_canal42[cont42]=resultado_f[i-1];
            cont42++;
        }
    }
}
}

```

```

double [][] datosmedida = new double [2*cuenta][15];

```

```

for (int j=0;j<15;j++) {
    for(int i = 0; i < cuenta; i++) {
        if (j==0) {
            datosmedida[i][j]=resultado_puertos_canal1[i];
        }
        if (j==1) {
            datosmedida[i][j]=resultado_f_canal1[i];
        }
        if (j==2) {
            datosmedida[i][j]=resultado_RSSI_canal1[i];
        }
        if(j==4) {
            datosmedida[i][j]=resultado_puertos_canal2[i];
        }
        if(j==5) {
            datosmedida[i][j]=resultado_f_canal2[i];
        }
        if(j==6) {
            datosmedida[i][j]=resultado_RSSI_canal2[i];
        }
        if(j==8) {
            datosmedida[i][j]=resultado_puertos_canal3[i];
        }
        if(j==9) {
            datosmedida[i][j]=resultado_f_canal3[i];
        }
    }
}

```

```

        if(j==10) {
            datosmedida[i][j]=resultado_RSSI_canal3[i];
        }
        if(j==12) {
            datosmedida[i][j]=resultado_puertos_canal4[i];
        }
        if(j==13) {
            datosmedida[i][j]=resultado_f_canal4[i];
        }

        if(j==14) {
            datosmedida[i][j]=resultado_RSSI_canal4[i];
        }
    }
}

//Estoy llenando todo lo referente a la antena 2
for (int j=0;j<15;j++) {
    for(int i = cuenta; i < 2*cuenta; i++) {
        if (j==0) {
            datosmedida[i][j]=resultado_puertos_canal12[i-cuenta];
        }
        if (j==1) {
            datosmedida[i][j]=resultado_f_canal12[i-cuenta];
        }
        if (j==2) {
            datosmedida[i][j]=resultado_RSSI_canal12[i-cuenta];
        }
        if(j==4) {
            datosmedida[i][j]=resultado_puertos_canal22[i-cuenta];
        }
        if(j==5) {
            datosmedida[i][j]=resultado_f_canal22[i-cuenta];
        }
        if(j==6) {
            datosmedida[i][j]=resultado_RSSI_canal22[i-cuenta];
        }
        if(j==8) {
            datosmedida[i][j]=resultado_puertos_canal32[i-cuenta];
        }
        if(j==9) {
            datosmedida[i][j]=resultado_f_canal32[i-cuenta];
        }
        if(j==10) {
            datosmedida[i][j]=resultado_RSSI_canal32[i-cuenta];
        }
        if(j==12) {
            datosmedida[i][j]=resultado_puertos_canal42[i-cuenta];
        }
        if(j==13) {
            datosmedida[i][j]=resultado_f_canal42[i-cuenta];
        }
        if(j==14) {

```

```

        datosmedida[i][j]=resultado_RSSI_canal42[i-cuenta];
    }
}

//A continuación, se llama a Matlab y se le pasan los datos pertinentes

matEng.putVariable("datos_medida"+String.valueOf(angle), datosmedida);
matEng.eval("theta=-95:1:95;");

matEng.eval("[Pot_antena1_canal1,Pot_antena1_canal2,Pot_antena1_canal3,Pot_antena1_canal4,Pot_antena2_canal1,Pot_antena2_canal2,Pot_antena2_canal3,Pot_antena2_canal4]=mediaRSSI_dBm(datos_medida"+String.valueOf(angle)+"",theta("+angle+""),Pot_antena1_canal1,Pot_antena1_canal2,Pot_antena1_canal3,Pot_antena1_canal4,Pot_antena2_canal1,Pot_antena2_canal2,Pot_antena2_canal3,Pot_antena2_canal4);");

matEng.eval("plot(theta,Pot_antena1_canal1,theta,Pot_antena2_canal1,theta,Pot_antena1_canal2,theta,Pot_antena2_canal2,theta,Pot_antena1_canal3,theta,Pot_antena2_canal3,theta,Pot_antena1_canal4,theta,Pot_antena2_canal4,'linewidth',0.7);");
matEng.eval("xlabel('Theta(deg)')");
matEng.eval("ylabel('Potencia en dB')");
matEng.eval("legend('P_A_n_t_e_n_a_1 (865,7 MHz)','P_A_n_t_e_n_a_2 (865,7 MHz)','P_A_n_t_e_n_a_1 (866,3 MHz)','P_A_n_t_e_n_a_2 (866,3 MHz)','P_A_n_t_e_n_a_1 (866,9 MHz)','P_A_n_t_e_n_a_2 (866,9 MHz)','P_A_n_t_e_n_a_1 (867,5 MHz)','P_A_n_t_e_n_a_2 (867,5 MHz)','Location','Best')");
matEng.eval("title('Diagramas de radiación digitales')");
matEng.eval("axis([-95 95 -87 -40])");
//función que mueve la mesa
matEng.eval("gira_mesa("+angle-95+",h)");
}
}

```

I.VI. HelloOctaneSdk

```

package com.example.sdksamples;

import com.impinj.octane.AntennaConfigGroup;
import com.impinj.octane.AutoStartMode;
import com.impinj.octane.AutoStopMode;
import com.impinj.octane.ImpinjReader;
import com.impinj.octane.OctaneSdkException;
import com.impinj.octane.ReportConfig;
import com.impinj.octane.Settings;
import com.impinj.octane.TagReportListener;

import java.util.ArrayList;
import java.util.Scanner;

public class HelloOctaneSdk {

```

```

    public static void main(String hostname, int angle, double tiempo_ms, int
puerto, double canal, TagReportListenerImplementation t) throws
OctaneSdkException, InterruptedException {

    ImpinjReader reader = new ImpinjReader();

    // Conectamos con el reader
    System.out.println("Connecting to " + hostname);
    reader.connect(hostname);

    // Aplicamos la configuración deseada
    Settings settings = reader.queryDefaultSettings();
    ArrayList<Double> list = new ArrayList<>();
    list.add(canal);
    settings.setTxFrequenciesInMhz(list);
    ReportConfig report = settings.getReport();
    report.setIncludeAntennaPortNumber(true);
    report.setIncludeChannel(true);
    report.setIncludePeakRssi(true);

    // Antenas
    AntennaConfigGroup ac = settings.getAntennas();
    ac.disableAll();
    ac.getAntenna((short) puerto).setEnabled(true);

    // Aplicamos la nueva configuración

    reader.applySettings(settings);
    reader.setTagReportListener(t);

    // Comenzamos a transmitir, en este momento se realiza la llamada a
TagReportListenerImplementation para que éste quede a la escucha de los datos
devueltos por el tag
    reader.start();
    Thread.sleep((long) tiempo_ms);

    //Paramos el reader y lo desconectamos:
    reader.stop();
    reader.disconnect();
    System.out.println("Done");
}
}

```

I.VI. TagReportListenerImplementation

```

package com.example.sdksamples;

import com.impinj.octane.ImpinjReader;
import com.impinj.octane.Tag;
import com.impinj.octane.TagReport;

```

```

import com.impinj.octane.TagReportListener;

import java.util.List;

public class TagReportListenerImplementation implements TagReportListener {

    public double [] rssi = new double [2000];
    public double [] f = new double [2000];
    public short puerto[] = new short[2000];
    public String IDS[] = new String[2000];
    int count = 0;
    int count2=0;
    int cuenta = 8;

    @Override
    public void onTagReported(ImpinjReader reader, TagReport report) {
        List<Tag> tags = report.getTags();

        for (Tag t : tags) {
            System.out.print(" EPC: " + t.getEpc().toString());

            if (reader.getName() != null) {
                System.out.print(" Reader_name: " + reader.getName());
            } else {
                System.out.print(" Reader_ip: " + reader.getAddress());
            }

            if (t.isAntennaPortNumberPresent()) {
                System.out.print(" antenna: " + t.getAntennaPortNumber());
            }

            if (t.isFirstSeenTimePresent()) {
                System.out.print(" first: " + t.getFirstSeenTime().ToString());
            }

            if (t.isLastSeenTimePresent()) {
                System.out.print(" last: " + t.getLastSeenTime().ToString());
            }

            if (t.isSeenCountPresent()) {
                System.out.print(" count: " + t.getTagSeenCount());
            }

            if (t.isRfDopplerFrequencyPresent()) {
                System.out.print(" doppler: " + t.getRfDopplerFrequency());
            }

            if (t.isPeakRssiInDbmPresent()) {
                System.out.print(" peak_rssi: " + t.getPeakRssiInDbm());
            }

            if (t.isChannelInMhzPresent()) {
                System.out.print(" chan_MHz: " + t.getChannelInMhz());
            }
        }
    }
}

```

```

    }

    if (t.isRfPhaseAnglePresent()) {
        System.out.print(" phase angle: " + t.getPhaseAngleInRadians());
    }

    if (t.isFastIdPresent()) {
        System.out.print("\n      fast_id: " + t.getTid().toHexString());

        System.out.print(" model: " +
            t.getModelDetails().getModelName());

        System.out.print(" epcsize: " +
            t.getModelDetails().getEpcSizeBits());

        System.out.print(" usermemsize: " +
            t.getModelDetails().getUserMemorySizeBits());
    }

    System.out.println("");
}
// En este punto leemos los datos que han venido de vuelta del tag y se
han captado por las antenas

    rssi[count] = report.getTags().get(0).getPeakRssiInDbm();
    puerto[count]= report.getTags().get(0).getAntennaPortNumber();
    IDS[count] = report.getTags().get(0).getEpc().toString();
    f[count] = report.getTags().get(0).getChannelInMhz();
    count++;
}
}
}

```

Anexo II. MATLAB

II.I. Cliente.m

```
% Controlador de ángulos de mesa giratoria y adquisición de datos
close all;
clear all;
clc;
matlab.engine.shareEngine;

contador=0;
t=tcip('localhost',30000,'NetworkRole','client');
fopen(t);

%Establecemos la comunicación con la mesa giratoria e %inicializamos
el ángulo
%-----
h = visa ('keysight', 'GPIB0::7::INSTR' );
fopen(h)
fprintf(h,'LD 1 DV');           % seleccionamos la mesa giratoria
angulo_inicio=-95;
fprintf(h,['LD' num2str(angulo_inicio) ' DG NP GO'])
pause(10);
%-----

% Llenar los vectores de potencias a 0 desde -95° a 95°
theta=-95:1:95;
Pot_antena1_canal1=zeros(1,length(theta));
Pot_antena1_canal2=zeros(1,length(theta));
Pot_antena1_canal3=zeros(1,length(theta));
Pot_antena1_canal4=zeros(1,length(theta));
Pot_antena2_canal1=zeros(1,length(theta));
Pot_antena2_canal2=zeros(1,length(theta));
Pot_antena2_canal3=zeros(1,length(theta));
Pot_antena2_canal4=zeros(1,length(theta));

% Bucle que donde i representa el ángulo de medida, realiza la
%conexión con Java para la adquisición de datos mediante la función
%"conexion.m"

for i=1:191
    conexion(i,t);
end

%Finaliza la conexión con la mesa giratoria y con eclipse
fclose(h);
fclose(t);
```


II.II. Conexion.m

```
function conexion(dato,t)

data=string(dato);
fprintf(t, data);

end
```

II.III. gira_mesa.m

```
% Función que hace girar la mesa a un determinado ángulo que se le
pasa como argumento

function gira_mesa(angulo, h)

    disp(['Cambio de grados ', num2str(angulo)]);
    fprintf(h, 'LD 1 DV')    % Seleccionamos la mesa giratoria
    fprintf(h, ['LD ' num2str(angulo) ' DG NP GO'])

    pause(5);

end
```

II.IV. mediaRSSI_dBm.m

```
function
[Pot_antena1_canal1,Pot_antena1_canal2,Pot_antena1_canal3,Pot_ante
na1_canal4,Pot_antena2_canal1,Pot_antena2_canal2,Pot_antena2_canal
3,Pot_antena2_canal4] =
mediaRSSI_dBm(datos_medida,angulo,Pot_antena1_canal1,Pot_antena1_c
anal2,Pot_antena1_canal3,Pot_antena1_canal4,Pot_antena2_canal1,Pot
_antena2_canal2,Pot_antena2_canal3,Pot_antena2_canal4)

Num_RSSIs=20;
% RSSI media
theta=-90:1:90;
position = find(theta==angulo);
%Toma de datos del archivo
RSSIs_canal1=datos_medida(:,3);
RSSIs_canal2=datos_medida(:,7);
RSSIs_canal3=datos_medida(:,11);
RSSIs_canal4=datos_medida(:,15);
RSSIs_canal1=RSSIs_canal1';
RSSIs_canal2=RSSIs_canal2';
RSSIs_canal3=RSSIs_canal3';
RSSIs_canal4=RSSIs_canal4';
```

```

% Contemplamos la posibilidad de que se reciba un 0 como RSSI. En
ese caso, cambiamos dicho valor por el de la sensibilidad mínima
que el reader es capaz de captar.
for i=1:16
    if RSSIs_canal1(i) == 0
        RSSIs_canal1(i)=-85;
    end
    if RSSIs_canal2(i) == 0
        RSSIs_canal2(i) = -85;
    end
    if RSSIs_canal3(i) == 0
        RSSIs_canal3(i) = -85;
    end
    if RSSIs_canal4(i) == 0
        RSSIs_canal4(i) ==-85;
    end
end

RSSIs_canal1=RSSIs_canal1';
RSSIs_canal2=RSSIs_canal2';
RSSIs_canal3=RSSIs_canal3';
RSSIs_canal4=RSSIs_canal4';
RSSIs_canal1_puerto1=RSSIs_canal1(1:Num_RSSIs);
RSSIs_canal2_puerto1=RSSIs_canal2(1:Num_RSSIs);
RSSIs_canal3_puerto1=RSSIs_canal3(1:Num_RSSIs);
RSSIs_canal4_puerto1=RSSIs_canal4(1:Num_RSSIs);

RSSIs_canal1_puerto2=RSSIs_canal1(Num_RSSIs+1:length(RSSIs_canal1)
);
RSSIs_canal2_puerto2=RSSIs_canal2(Num_RSSIs+1:length(RSSIs_canal1)
);
RSSIs_canal3_puerto2=RSSIs_canal3(Num_RSSIs+1:length(RSSIs_canal1)
);
RSSIs_canal4_puerto2=RSSIs_canal4(Num_RSSIs+1:length(RSSIs_canal1)
);

%RSSI media

RSSIs_canal1_puerto1_lineal=10.^(RSSIs_canal1_puerto1/10);
RSSIs_canal1_puerto1_Media_dBm=10*log10(sum(RSSIs_canal1_puerto1_l
ineal)/length(RSSIs_canal1_puerto1_lineal));
Pot_antena1_canal1(position)=RSSIs_canal1_puerto1_Media_dBm;

RSSIs_canal2_puerto1_lineal=10.^(RSSIs_canal2_puerto1/10);
RSSIs_canal2_puerto1_Media_dBm=10*log10(sum(RSSIs_canal2_puerto1_l
ineal)/length(RSSIs_canal2_puerto1_lineal));
Pot_antena1_canal2(position)=RSSIs_canal2_puerto1_Media_dBm;

RSSIs_canal3_puerto1_lineal=10.^(RSSIs_canal3_puerto1/10);
RSSIs_canal3_puerto1_Media_dBm=10*log10(sum(RSSIs_canal3_puerto1_l
ineal)/length(RSSIs_canal3_puerto1_lineal));
Pot_antena1_canal3(position)=RSSIs_canal3_puerto1_Media_dBm;

```

```
RSSIs_canal4_puerto1_lineal=10.^(RSSIs_canal4_puerto1/10);
RSSIs_canal4_puerto1_Media_dBm=10*log10(sum(RSSIs_canal4_puerto1_lineal)/length(RSSIs_canal4_puerto1_lineal));
Pot_antena1_canal4(position)=RSSIs_canal4_puerto1_Media_dBm;

RSSIs_canal1_puerto2_lineal=10.^(RSSIs_canal1_puerto2/10);
RSSIs_canal1_puerto2_Media_dBm=10*log10(sum(RSSIs_canal1_puerto2_lineal)/length(RSSIs_canal1_puerto2_lineal));
Pot_antena2_canal1(position)=RSSIs_canal1_puerto2_Media_dBm;

RSSIs_canal2_puerto2_lineal=10.^(RSSIs_canal2_puerto2/10);
RSSIs_canal2_puerto2_Media_dBm=10*log10(sum(RSSIs_canal2_puerto2_lineal)/length(RSSIs_canal2_puerto2_lineal));
Pot_antena2_canal2(position)=RSSIs_canal2_puerto2_Media_dBm;

RSSIs_canal3_puerto2_lineal=10.^(RSSIs_canal3_puerto2/10);
RSSIs_canal3_puerto2_Media_dBm=10*log10(sum(RSSIs_canal3_puerto2_lineal)/length(RSSIs_canal3_puerto2_lineal));
Pot_antena2_canal3(position)=RSSIs_canal3_puerto2_Media_dBm;

RSSIs_canal4_puerto2_lineal=10.^(RSSIs_canal4_puerto2/10);
RSSIs_canal4_puerto2_Media_dBm=10*log10(sum(RSSIs_canal4_puerto2_lineal)/length(RSSIs_canal4_puerto2_lineal));
Pot_antena2_canal4(position)=RSSIs_canal4_puerto2_Media_dBm;
```

end

Anexo III. Hojas de Características

Far-Field RAIN RFID Antenna **Laird**

865-868 MHz



The Far-Field panel antennas by Laird are circularly polarized providing reception and transmission of signals in the 865-868 MHz frequency band. Laird's industry renowned design methodology achieves maximum efficiency and performance across the entire frequency band.

Both VSWR and axial ratios are excellent and allow the user to achieve the maximum performance for an antenna of this type. The antennas are housed in a heavy duty radome enclosure that can be directly wall mounted. An optional articulating mount allows either wall or mast mounting. The antennas are offered with an integrated coax pigtail and a variety of connector types are available.

FEATURES AND BENEFITS

- Low profile
- Extremely low VSWR and axial ratio
- Weather and UV resistant radome
- Wide range of connector and cable options
- Left-hand and right-hand CP versions

ELECTRICAL SPECIFICATIONS	
Impinj Part Numbers	IPJ-A1000-EU1/IPJ-A1001-EU1
Laird Part Numbers	S8658PLJ96RTN/S8658PRJ96RTN
Operating Frequency (MHz)	865-868 MHz
VSWR – Max	1.5:1
Gain (dBic)	8.5
Nominal Impedance (Ohms)	50
Max Power - Ambient 25°C (W)	10
Polarization	Circular right-hand or circular left-hand
Horizontal Plane 3 dB Beamwidth	70°
Front-to-Back Ratio (dB)	16
Axial Ratio (dB)	2

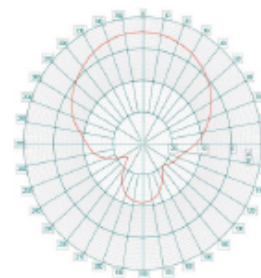
MECHANICAL SPECIFICATIONS	
Dimensions – length x width x height – mm (Inches)	259 x 259 x 34 (10.2 x 10.2 x 1.32)
Weight – kg (lbs.)	0.91 (2.0)
Radome Material	High strength PC
Mount Style	Threaded stud
Lightning Protection	DC grounded

ENVIRONMENTAL SPECIFICATIONS	
Operating Temperature – °C (°F)	-25 to +70°C (-13 to +158°F)
Environmental Rating	IP 54
Material Substance Compliance	RoHS compliant

CONFIGURATION

IMPINJ PART NUMBERS	LAIRD PART NUMBERS	CONNECTOR
IPJ-A1000-EU1	S8658PLJ-IP1	Coax Pigtail, Rev TNC Male
IPJ-A1001-EU1	S8658PRJ-IP1	Coax Pigtail, Rev TNC Male

RADIATION PATTERN 867 MHz



Laird warrants to the original and user customer of its products that its products are free from defects in materials and workmanship. Subject to conditions and limitations Laird will, at its option, either repair or replace any part of its products that prove defective because of improper workmanship or materials. This limited warranty is in force for the useful lifetime of the original and product into which the Laird product is installed. Useful lifetime of the original and product may vary but is not to exceed five (5) years from the original date of the end product purchase.

Any information furnished by Laird Inc. and its agents is believed to be accurate and reliable. All specifications are subject to change without notice. Responsibility for the use and application of Laird materials rests with the end user; since Laird and its agents cannot be aware of all potential uses. Laird makes no warranties as to the fitness, merchantability or suitability of any Laird materials or products for any specific or general uses. Laird shall not be liable for incidental or consequential damages of any kind. All Laird products are sold pursuant to the Laird Terms and Conditions of sale in effect from time to time, a copy of which will be furnished upon request.

© Copyright 2018 Laird Inc. All Rights Reserved. Laird, Laird Technologies, the Laird Logo, and other marks are trademarks or registered trademarks of Laird Inc. or an affiliate company thereof. Other product or service names may be the property of third parties. Nothing herein provides a license under any Laird or any third party intellectual property rights.

Impinj is a registered trademark of Impinj, Inc.

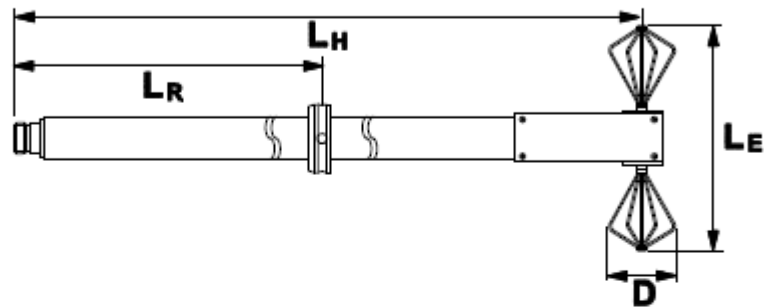


ANT-DS-S8658Pxx_impinj 1218

SCHWARZBECK MESS - ELEKTRONIK

An der Klinge 29 D-69250 Schönau Tel.: 06228/1001 Fax.: (49)6228/1003

Mikrowellen Bikonusantenne SBA 9113 Microwave Biconical Antenna SBA 9113



Technische Daten:

Linear polarisierte Mikrowellen-
Bikonus-Breitbandantenne

Frequenzbereich: 500 MHz - 3 GHz
Anschluß: Buchse 50 Ω N
Befestigungsrohr: $L_H=560\text{mm}$, $d=22\text{mm}$
Rastring: $L_R=190\text{mm}$
Elementlänge gesamt: $L_E=140\text{mm}$
Elementdurchmesser: $D=49\text{mm}$
Elementaufnahme: M 4
Isotropgewinn: typ. -1.5...+1.5 dBi
(600 MHz - 3 GHz)

Antennenwandlungsmaß: 25 ... 41 dB/m
SWR typ.: 1-4
Balun (verlustarm): 1:1
Inversionssymmetrie: typ. < 0.2 dB
(600 MHz - 3 GHz)

Max. Eingangsleistung: 20 W
Halbwertsbreite E-Ebene: 80° - 50°
Gewicht: 750 g

Specifications:

Linear polarised microwave
biconical broadband antenna

Frequency range:
Connector: female

Mounting tube:
Index Ring:

Element length total:
Element diameter:

Element fixture:
Isotropic gain:

Antenna Factor:
SWR typ.:

Balun (low loss):
Inversion Symmetry:

Max. Input Power:
Half-Power Beamwidth (E-plane):
Weight: