

C. Guía de análisis y verificación de espacios libres de la TVWS

Flujograma utilizado en GNU Radio, algoritmo utilizado en Google Colab y archivos obtenidos.

INTRODUCCIÓN

En este informe se abordó la verificación experimental de espacios libres en la banda de Televisión Digital Terrestre, TV White Spaces (TVWS), utilizando un sistema de Radio Definida por Software (SDR) y un analizador de espectro como patrón. El objetivo fue identificar, con la mayor trazabilidad posible, qué porciones del espectro comprendido entre 470 MHz y 698 MHz se encuentran ocupados por servicios de radiodifusión y cuáles pueden considerarse candidatos para el establecimiento de un enlace de comunicaciones de baja potencia orientado a la transmisión de datos médicos.

A diferencia de etapas previas, más centradas en simulación y en el modelo de propagación de Longley–Rice, en esta fase se diseñó y ejecutó un barrido sistemático del espectro, utilizando una antena sintonizada al rango UHF/TVWS y el flujograma de GNU Radio denominado "prueba_barrido" para entender más sobre este flujograma revisar otra vez el (Anexo A). Basándonos en eso se construyó un esquema de trabajo de dos niveles: 1. Adquisición de datos en paralelo con el SDR y con el analizador de espectro, 2. Procesamiento de todos los archivos mediante scripts en Google Colab para comparar curvas, estimar correlaciones y detectar automáticamente canales ocupados.

DESARROLLO

C.0.1 Análisis GNU Radio vs Analizador de espectro

Para la verificación de espacios libres de la TVWS se reutilizó el flujograma "prueba_barrido" en GNU Radio, el cual se encuentra su explicación en el (Anexo A), pero ahora con una antena para

el rango UHF conectada a la entrada del SDR. El rango completo de interés fue de 470 MHz a 698 MHz (canales 14 al 51), y se diseñó un esquema de 19 barridos consecutivos, contiguos, cada uno con un ancho de banda de 12,5 MHz.

El SDR se configuró con una frecuencia de muestreo de 12,5 MHz y una FFT de 1024 puntos. Con estos valores, la resolución teórica por bin en el dominio de la frecuencia se obtiene con la relación $\Delta f = F_s/N$, es decir: $\Delta f \approx 12,5 \text{ MHz} / 1024 \approx 12\,207 \text{ Hz}$. Este valor es coherente con el RBW de 10 kHz configurado en el analizador de espectro, de modo que la comparación entre ambos instrumentos no se ve afectada por ciertas diferencias en la resolución espectral.

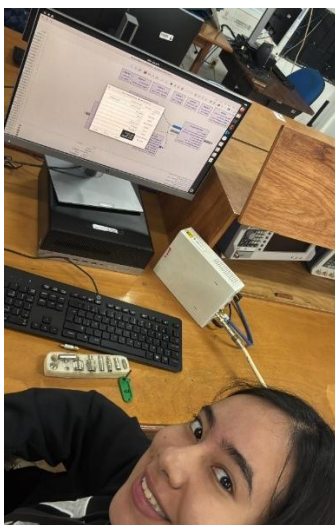


Figura 1. Adquisición espectral de cada barrido manualmente.

Cada barrido se definió como una ventana de $\pm 6,25 \text{ MHz}$ alrededor de una frecuencia central distinta. De esta forma, la sucesión de 19 frecuencias centrales cubre toda la banda objetivo sin dejar huecos entre ventanas de adquisición. En la práctica, esto significa que cualquier canal de 6 MHz de televisión que caiga entre 470 MHz y 698 MHz queda contenido completamente en al menos una de las ventanas. A continuación, se muestra los 19 barridos realizados, en la tabla 1 y la tabla 2.

Barrido	Frecuencia central [MHz]	Rango cubierto [MHz]
1	476	469,75 – 482,25
2	488.5	482,25 – 494,75
3	501	494,75 – 507,25
4	513.5	507,25 – 519,75
5	526	519,75 – 532,25
6	538.5	532,25 – 544,75
7	551	544,75 – 557,25
8	563.5	557,25 – 569,75
9	576	569,75 – 582,25
10	588.5	582,25 – 594,75
11	601	594,75 – 607,25
12	613.5	607,25 – 619,75
13	626	619,75 – 632,25
14	638.5	632,25 – 644,75
15	651	644,75 – 657,25
16	663.5	657,25 – 669,75
17	676	669,75 – 682,25
18	688.5	682,25 – 694,75
19	701	694,75 – 707,25

Tabla 1. 19 barridos para ver la ocupación espectral en la TVWS.

Barrido	Centro MHz	r, correlación	Piso de ruido dBm	Sigma dB	Umbral dBm	Canales detectados, nº	Frecuencias detectadas MHz
1	476	0.0757	-116.23	2.55	-109.86	0	—
2	488.5	0.1538	-113.88	2.51	-107.61	0	—
3	501	0.0531	-119.47	1.81	-114.94	0	—
4	513.5	0.0399	-119.33	1.94	-114.48	0	—
5	526	-0.0049	-119.27	1.61	-115.24	0	—
6	538.5	0.0025	-118.94	1.85	-114.31	0	—
7	551	0.2352	-118.06	2.1	-112.8	2	548.05 — 551.85, 551.93 — 554.11
8	563.5	0.0073	-119	1.84	-114.4	0	—
9	576	0.0159	-118.62	1.98	-113.67	0	—
10	588.5	0.0355	-119.79	2.14	-114.45	0	—
11	601	0.1	-118.31	1.71	-114.04	0	—
12	613.5	0.0727	-119.5	2.06	-114.35	0	—
13	626	0.0941	-118.89	2.27	-113.21	0	—

14	638.5	0.1895	-117.86	1.65	-113.73	1	634.97 638.21	—
15	651	0.3302	-117.53	2.37	-111.61	1	656.09 657.25	—
16	663.5	0.7143	-116.29	2.33	-110.47	1	657.25 661.94	—
17	676	0.5154	-117.72	2.03	-112.64	1	679.76 682.25	—
18	688.5	0.6972	-117.84	1.67	-113.66	1	682.25 685.99	—
19	701	0.11	-119.82	2.59	-113.35	0	—	

Tabla 2. Métricas de los 19 barridos y detección de canales.

Cada uno de estos 19 barridos se asoció a un par de archivos: un CSV proveniente del analizador de espectro y otro archivo CSV generado por el SDR a partir del flujograma. Esto permitió, más adelante, comparar ambas mediciones y evaluar la coherencia entre instrumentos.

Para que la adquisición experimental no dependiera de ejecutar manualmente cada flujograma de GNU Radio, se preparó un pequeño paquete de automatización en Python. A partir del archivo original “prueba_barrido.grc” se generaron scripts individuales en GNU Radio en formato Python (por ejemplo, prueba_barrido_476M.py, prueba_barrido_488_5M.py, hasta prueba_barrido_701M.py), de manera que cada script correspondía a un barrido con una frecuencia central específica. A partir de los anterior, se implementó un script maestro, denominado “maestro.py”, cuya función es actuar como “controlador”. Internamente, este script

define una lista ordenada con los nombres de los 19 scripts de barrido y los ejecuta de forma secuencial mediante llamadas a `subprocess.run`. En términos prácticos, el flujo del script maestro puede describirse así:

- Construye una lista de rutas, por ejemplo:
 - `["prueba_barrido_476M.py", "prueba_barrido_488_5M.py", ..., "prueba_barrido_701M.py"]`.
- Recorre la lista en un bucle `for`, y para cada elemento:
 - Llama a `subprocess.run(["python", nombre_script])` para ejecutar el barrido correspondiente.
 - Verifica el código de salida del proceso para confirmar que el script terminó correctamente.
 - Inserta una pausa controlada (`sleep`) si es necesario, para asegurar que el archivo de salida termine de escribirse antes de continuar con el siguiente barrido.
- Una vez finalizada la ejecución, mueve o guarda los archivos generados en carpetas con los respectivos nombres correspondientes a cada barrido.

Los archivos producidos por el SDR se organizaron en una carpeta llamada “gnuradioreal_frecuencias”, mientras que los archivos provenientes del analizador de espectro se almacenaron en `analizadorreal_frecuencias`. Este esquema de nombres permite identificar rápidamente, para cada frecuencia central, el par de archivos (“SDR” y “analizador”) que representan el mismo barrido. Esta automatización reduce errores humanos (por ejemplo, ejecutar un barrido equivocado o sobrescribir archivos) y deja un rastro claro para la trazabilidad de los datos.

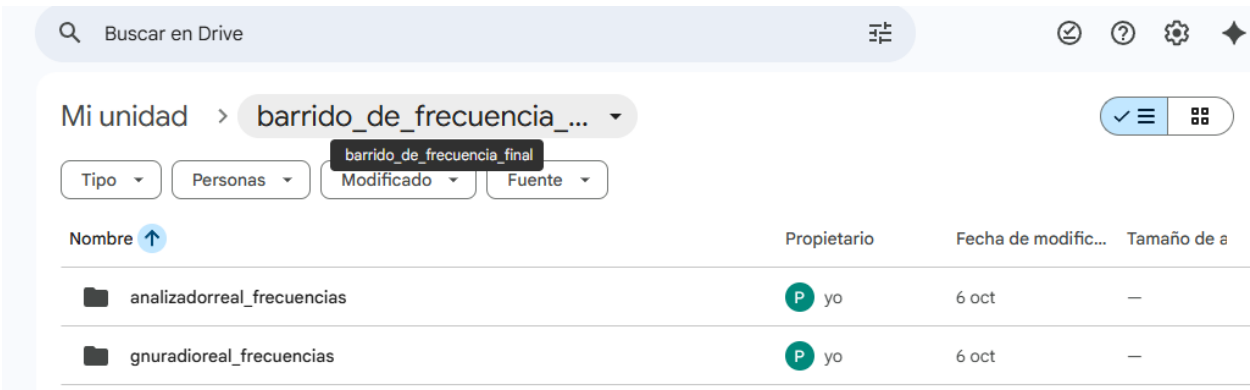


Figura 1. Archivos CSV del analizador y del flujograma de GNU radio.

Nombre	Fecha de modificación	Tipo	Tamaño
476barrido1.csv	28/11/2025 9:42 a. m.	Archivo de valores...	90 KB
488barrido2.csv	28/11/2025 9:42 a. m.	Archivo de valores...	90 KB
501barrido3.csv	28/11/2025 9:42 a. m.	Archivo de valores...	90 KB
513barrido4.csv	28/11/2025 9:42 a. m.	Archivo de valores...	90 KB
526barrido5.csv	28/11/2025 9:42 a. m.	Archivo de valores...	90 KB
538barrido6.csv	28/11/2025 9:42 a. m.	Archivo de valores...	90 KB
551barrido7.csv	28/11/2025 9:42 a. m.	Archivo de valores...	90 KB
563barrido8.csv	28/11/2025 9:42 a. m.	Archivo de valores...	90 KB
576barrido9.csv	28/11/2025 9:42 a. m.	Archivo de valores...	90 KB
588barrido010.csv	28/11/2025 9:42 a. m.	Archivo de valores...	90 KB
601barrido011.csv	28/11/2025 9:42 a. m.	Archivo de valores...	90 KB
613barrido012.csv	28/11/2025 9:42 a. m.	Archivo de valores...	90 KB
626barrido013.csv	28/11/2025 9:42 a. m.	Archivo de valores...	90 KB
638barrido014.csv	28/11/2025 9:42 a. m.	Archivo de valores...	90 KB
651barrido015.csv	28/11/2025 9:42 a. m.	Archivo de valores...	90 KB
663barrido016.csv	28/11/2025 9:42 a. m.	Archivo de valores...	90 KB
676barrido017.csv	28/11/2025 9:42 a. m.	Archivo de valores...	90 KB
688barrido018.csv	28/11/2025 9:42 a. m.	Archivo de valores...	90 KB
701barrido019.csv	28/11/2025 9:42 a. m.	Archivo de valores...	90 KB

Figura 2. Archivos CSV del analizador de espectro.




















 476M.csv	28/11/2025 9:42 a. m.	Archivo de valores...	25 KB
 488_5M.csv	28/11/2025 9:42 a. m.	Archivo de valores...	21 KB
 501M.csv	28/11/2025 9:42 a. m.	Archivo de valores...	25 KB
 513_5M.csv	28/11/2025 9:42 a. m.	Archivo de valores...	25 KB
 526M.csv	28/11/2025 9:42 a. m.	Archivo de valores...	25 KB
 538_5M.csv	28/11/2025 9:42 a. m.	Archivo de valores...	25 KB
 551M.csv	28/11/2025 9:42 a. m.	Archivo de valores...	25 KB
 563_5M.csv	28/11/2025 9:42 a. m.	Archivo de valores...	25 KB
 576M.csv	28/11/2025 9:42 a. m.	Archivo de valores...	25 KB
 588_5M.csv	28/11/2025 9:42 a. m.	Archivo de valores...	25 KB
 601M.csv	28/11/2025 9:42 a. m.	Archivo de valores...	25 KB
 613_5M.csv	28/11/2025 9:42 a. m.	Archivo de valores...	25 KB
 626M.csv	28/11/2025 9:42 a. m.	Archivo de valores...	25 KB
 638_5M.csv	28/11/2025 9:42 a. m.	Archivo de valores...	25 KB
 651M.csv	28/11/2025 9:42 a. m.	Archivo de valores...	25 KB
 663_5M.csv	28/11/2025 9:42 a. m.	Archivo de valores...	21 KB
 676M.csv	28/11/2025 9:42 a. m.	Archivo de valores...	21 KB
 688_5M.csv	28/11/2025 9:42 a. m.	Archivo de valores...	21 KB
 701M.csv	28/11/2025 9:42 a. m.	Archivo de valores...	25 KB

Figura 3. Archivos CSV del flujograma de GNU Radio “prueba_barrido”.

C.0.2 Script del algoritmo implementado para la comparación de archivos

Procesamiento en Google Colab: comparación barrido por barrido (Rodríguez y Díaz, 2025)

Una vez obtenidos los archivos de todos los barridos, el siguiente paso consistió en analizar, para cada frecuencia central, la similitud entre lo medido por el analizador de espectro y lo registrado por el SDR. Para ello el script recibe dos rutas de archivo:

1. Un archivo CSV con frecuencia y magnitud en dBm, salida del analizador de espectro.
2. Un archivo CSV generado por el SDR, que incluye metadatos del flujo como frecuencia central, tasa de muestreo, longitud del vector y los bloques numéricos resultantes de las FFT.

El flujo lógico del script para un barrido individual se resume en los siguientes pasos, donde se explica cada bloque concreto del algoritmo:

1. **Lectura y limpieza del CSV del analizador**

Se utiliza `pandas.read_csv` para cargar el archivo. El script:

- Busca la fila que contiene la palabra "Frequency" en la primera columna.
- A partir de esa fila, extrae únicamente las dos columnas de interés: frecuencia y magnitud.
- Renombra las columnas como "Frecuencia [Hz]" y "Magnitud [dBm]".
- Convierte ambas columnas a tipo numérico (`to_numeric`) y elimina filas vacías o no válidas (`dropna`).

Esta limpieza asegura que el vector de magnitudes del analizador (`mag_ref`) esté libre de valores no numéricos y pueda usarse como referencia.

2. Normalización de la magnitud del analizador

El vector `mag_ref` se normaliza restando su media y dividiendo por su desviación estándar. Esto centra la curva alrededor de cero y le da varianza unitaria. La razón es que, para calcular correlaciones, interesa principalmente la **forma** del espectro (relieve, picos, valles) y no su nivel absoluto.

3. Lectura del archivo de texto del SDR y extracción de parámetros

El archivo CSV generado por GNU Radio recorre línea por línea. En las primeras líneas aparecen metadatos del flujo:

- Línea con "Frequency" → se extrae la **frecuencia central** (`freq_central`).
- Línea con "Sample Rate" → se obtiene la **tasa de muestreo** (`sample_rate`).
- Línea con "Vector Length" → se obtiene la **longitud del vector** (`vector_len`).

Estos valores permiten luego construir el eje de frecuencia del SDR y saber cuántos puntos tiene cada FFT.

4. Extracción de los bloques numéricos del SDR

Las líneas que contienen únicamente números, espacios, comas o puntos se interpretan como bloques de datos. Para cada línea se realizó lo siguiente:

- Se separan los valores con `re.split`.
- Se intenta convertir cada valor a `float`.
- Si una línea produce más de un número, se añade como parte de un bloque.

De esta manera se construye una lista `datos_txt`, donde cada elemento es un vector de magnitudes correspondiente a una ejecución de la FFT en el SDR.

5. Construcción del eje de frecuencias del SDR

Con `np.linspace` se genera un vector de frecuencias `freqs` que va desde $(\text{freq_central} - \text{sample_rate}/2)$ hasta $(\text{freq_central} + \text{sample_rate}/2)$ con `vector_len` puntos. Este eje representa, en Hz, la posición de cada bin de la FFT del SDR.

6. Selección del bloque más representativo mediante correlación de Pearson

Para comparar el analizador y el SDR:

- Si `mag_ref` y el bloque SDR tienen longitudes distintas, `mag_ref` se re-interpola (`np.interp`) a la longitud `vector_len`.
- Para cada bloque en `datos_txt`:
 - Se normaliza el bloque (restando su media y dividiendo por su desviación estándar).

- Se calcula la **correlación de Pearson** r entre el bloque normalizado y mag_ref (también normalizado).
- El script se queda con el bloque que produce el mayor valor de r . Ese bloque es el que mejor replica, en forma, el espectro medido por el analizador.

Esta lógica permite descartar bloques anómalos (por ejemplo, afectados por ruido transitorio) y elegir el que realmente describe el estado estacionario del canal durante el barrido.

7. Visualización conjunta de analizador y SDR

Finalmente, se genera una gráfica donde:

- Se representa la curva "Magnitud [dBm]" del analizador en función de la frecuencia (en MHz).
- Se superpone el bloque SDR seleccionado, usando el eje freqs, normalmente con una línea discontinua.
- Se marcan puntos de interés, como la magnitud mínima o picos relevantes, para facilitar la inspección visual.

Esta visualización permite corroborar que ambas curvas tienen la misma estructura de picos y valles, confirmando la coherencia entre instrumentos.

Detección de ocupación espectral

La idea es pasar de la señal continua de magnitudes a una lista de segmentos de frecuencia donde se detecta señal significativa, compatible con canales de televisión.

El procedimiento implementado en el script se puede describir así:

1. Suavizado de la curva de magnitud

Se aplica un promedio de longitud, utilizando un “padding” reflejado en los bordes para no introducir saltos artificiales. El objetivo es:

- Reducir ruido de alta frecuencia en la curva.
- Conservar la forma global de los picos reales.

2. Estimación del piso de ruido

Sobre la curva suavizada:

- Se selecciona un porcentaje de las muestras de menor magnitud (por ejemplo, el 20 % inferior).
- Con esas muestras se calcula la **media** (piso_ruido) y la **desviación estándar** (sigma_ruido).

Ese subconjunto representa, de forma aproximada, el “ruido de fondo” del canal cuando no hay señales fuertes activas.

• Cálculo de un umbral adaptativo

El umbral se define como:

$$\text{umbral } \lambda = \mu_{\text{ruido}} + k\sigma_{\text{ruido}} \quad (1)$$

donde μ_{ruido} es el promedio, σ_{ruido} desviación estándar y k es un factor ajustable, típicamente entre 2 y 3. Cuanto mayor es $k\sigma_{\text{ruido}}$, más conservador es el detector por lo que requiere señales más altas para considerar un canal como ocupado.

3. Construcción de una máscara de ocupación

Se genera una máscara booleana mask_ocupado:

- True para los puntos cuya magnitud suavizada está por encima del umbral.
- False para los puntos por debajo del umbral.

4. Agrupación de regiones contiguas y tolerancia a huecos

Se recorre la máscara de izquierda a derecha para agrupar puntos consecutivos con `mask_ocupado = True`. Para evitar que dividan un canal real en varios fragmentos:

- Se permite una pequeña cantidad de huecos internos (`gap_permitido`).
- Sólo cuando el número de puntos consecutivos por debajo del umbral supera esa tolerancia se considera que un canal ha terminado.

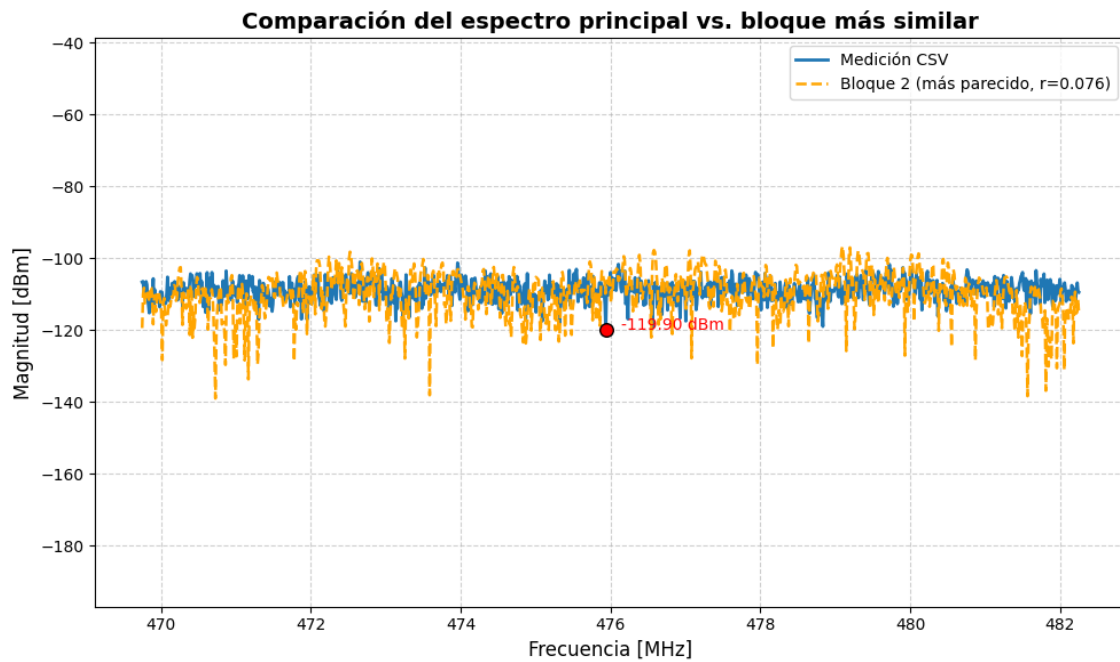
5. Filtrado por ancho mínimo de canal

Se calcula la diferencia entre la frecuencia de inicio y de fin. Si el ancho en frecuencia es mayor o igual a un límite (por ejemplo, 6 MHz), el tramo se conserva como **canal ocupado**. Segmentos más estrechos se descartan como picos aislados o ruido residual.

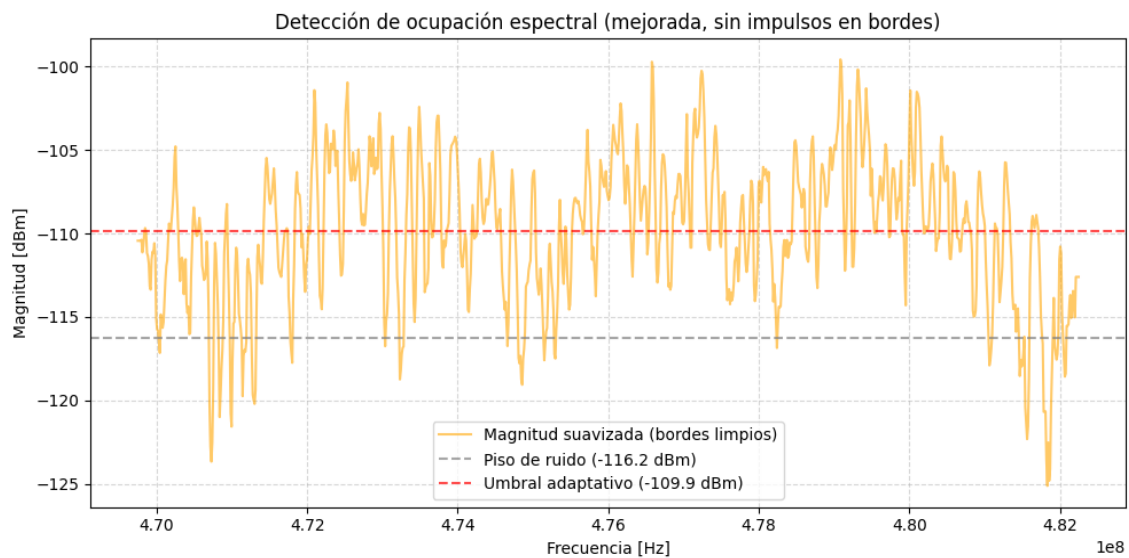
El resultado final es una lista de intervalos, que representan las porciones del barrido identificadas como **ocupadas**. Por complementación, las porciones sin actividad significativa son las que se consideran candidatas a TV White Spaces, siempre en coherencia con la asignación regulatoria de canales.

RESULTADOS Y HALLAZGOS

Figura 4. Primer barrido

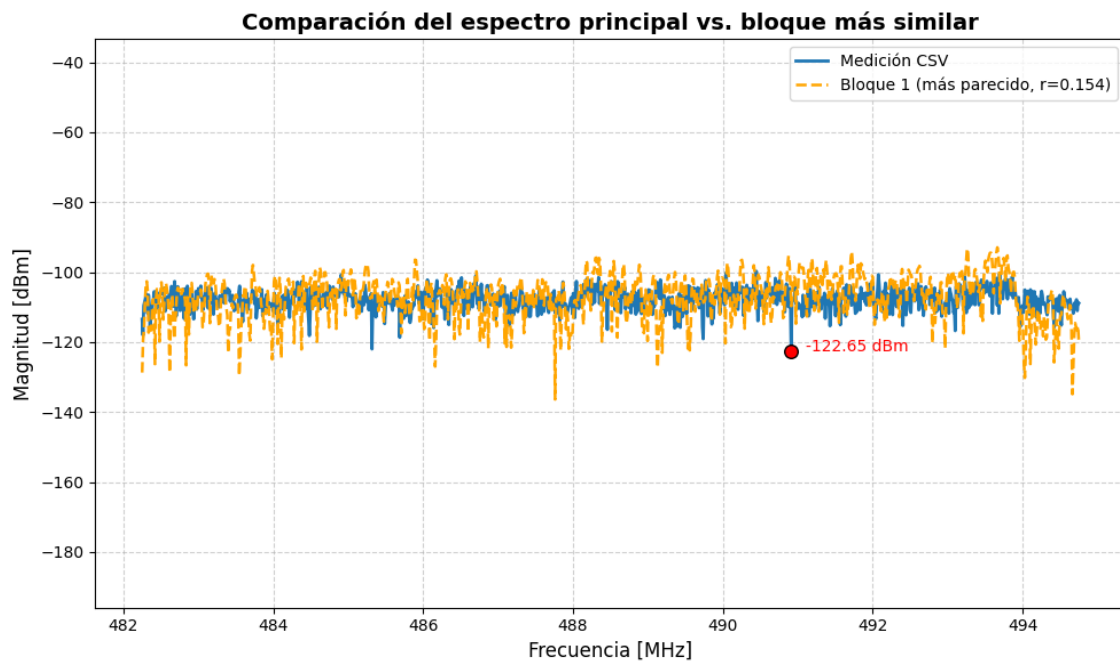


a) archivo CSV del GNU Radio vs el archivo CSV del analizador de espectro.

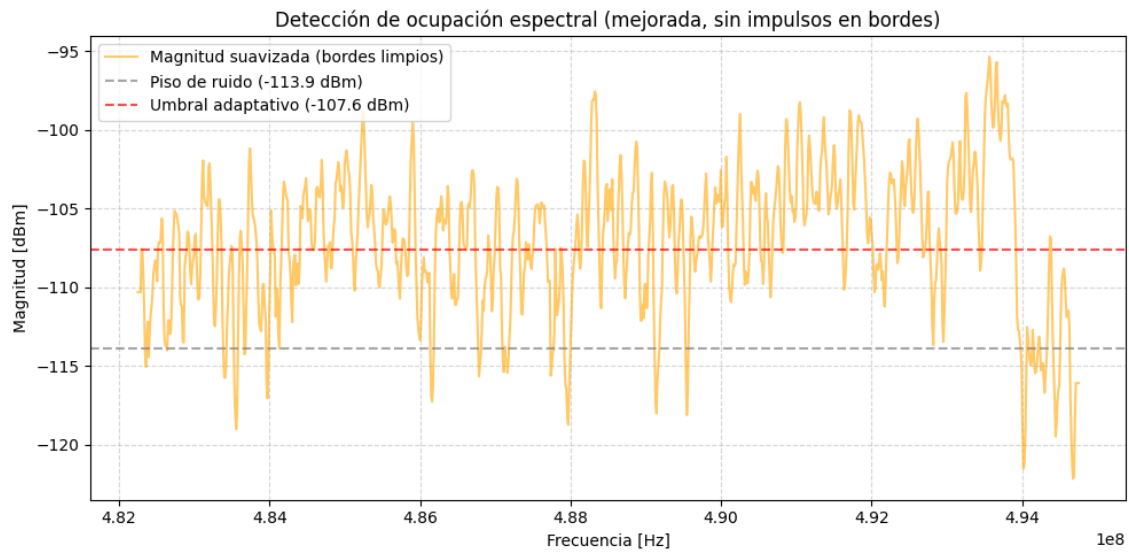


b) Detección de ocupación canal

Figura 5. Segundo barrido

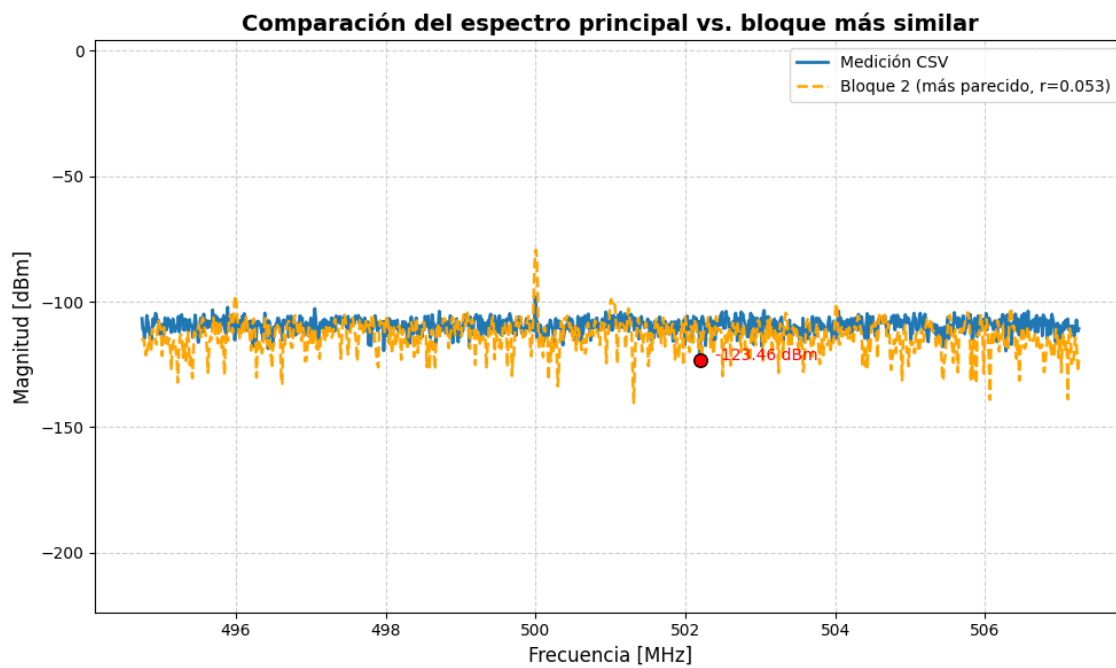


a) archivo CSV del GNU Radio vs el archivo CSV del analizador de espectro.

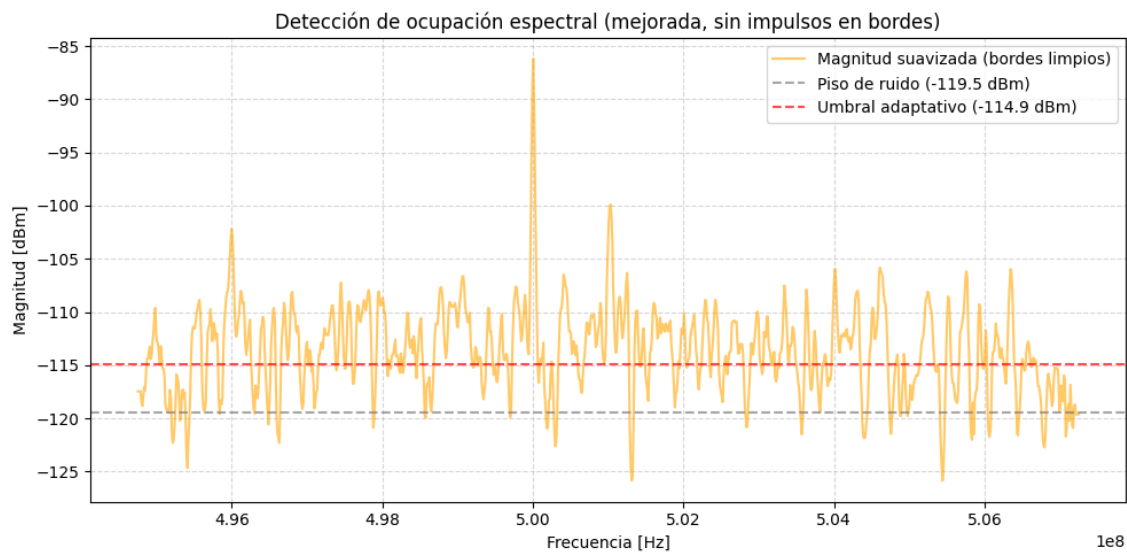


b) Detección de ocupación canal

Figura 6. Tercer barrido

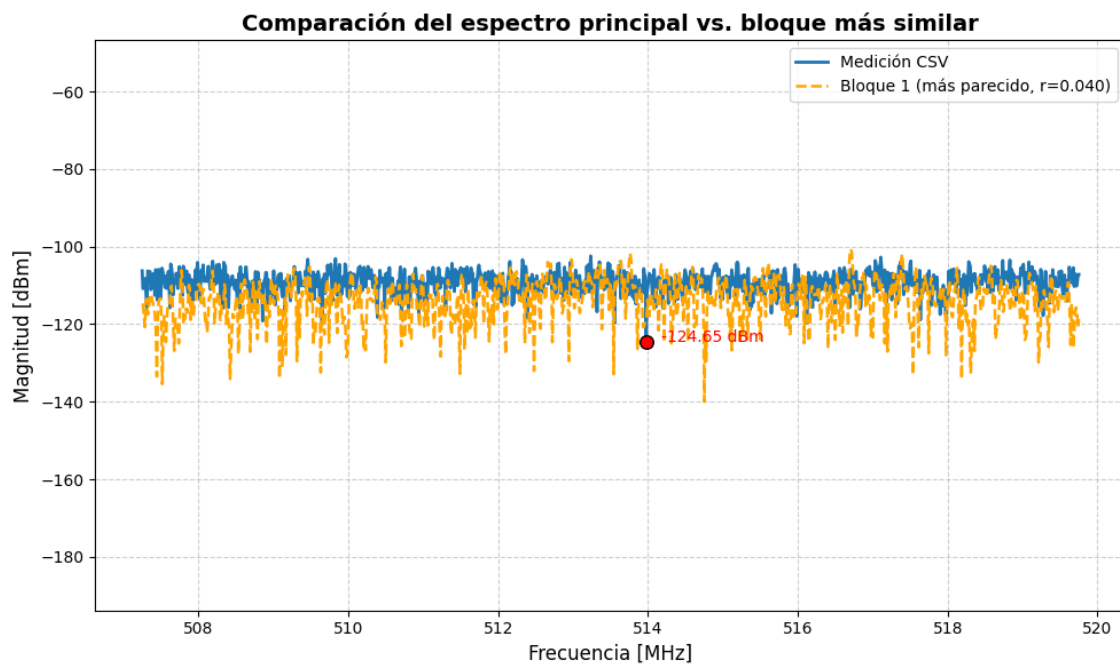


a) archivo CSV del GNU Radio vs el archivo CSV del analizador de espectro.

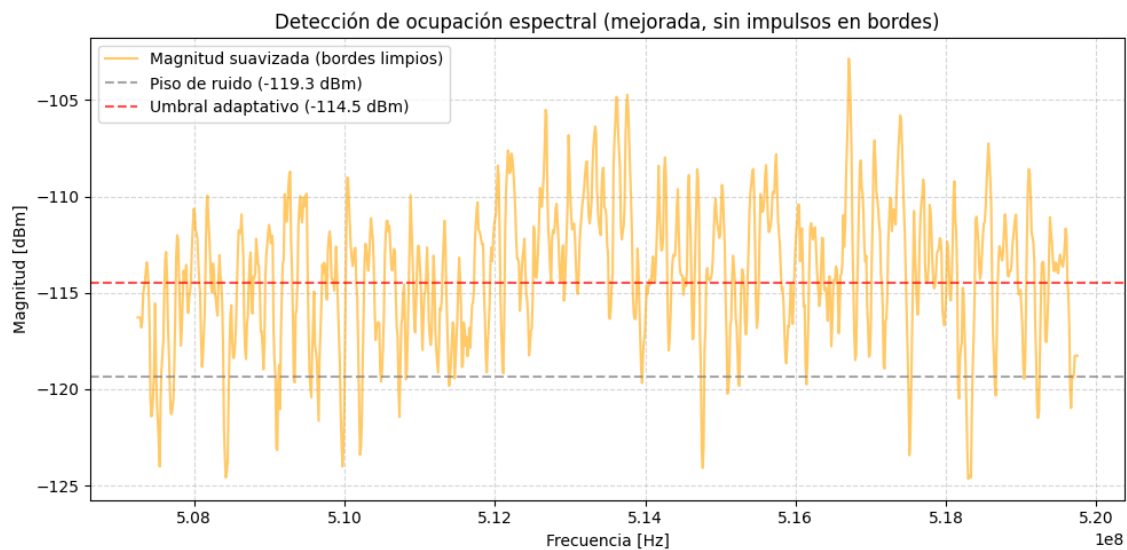


b) Detección de ocupación canal

Figura 7. Cuarto barrido

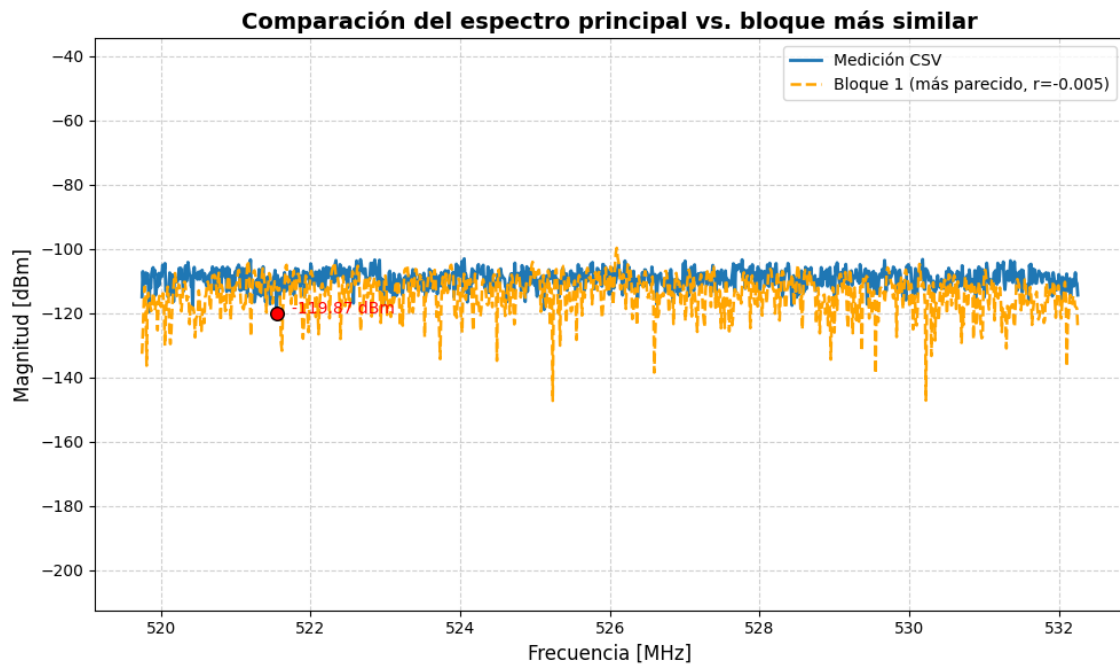


a) archivo CSV del GNU Radio vs el archivo CSV del analizador de espectro.

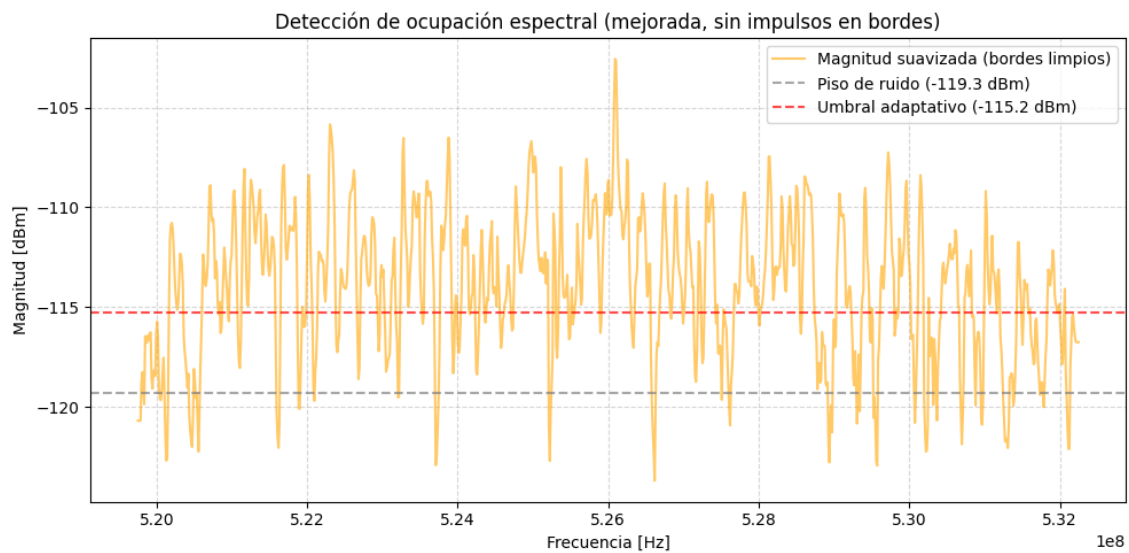


b) Detección de ocupación canal

Figura 8. Quinto barrido

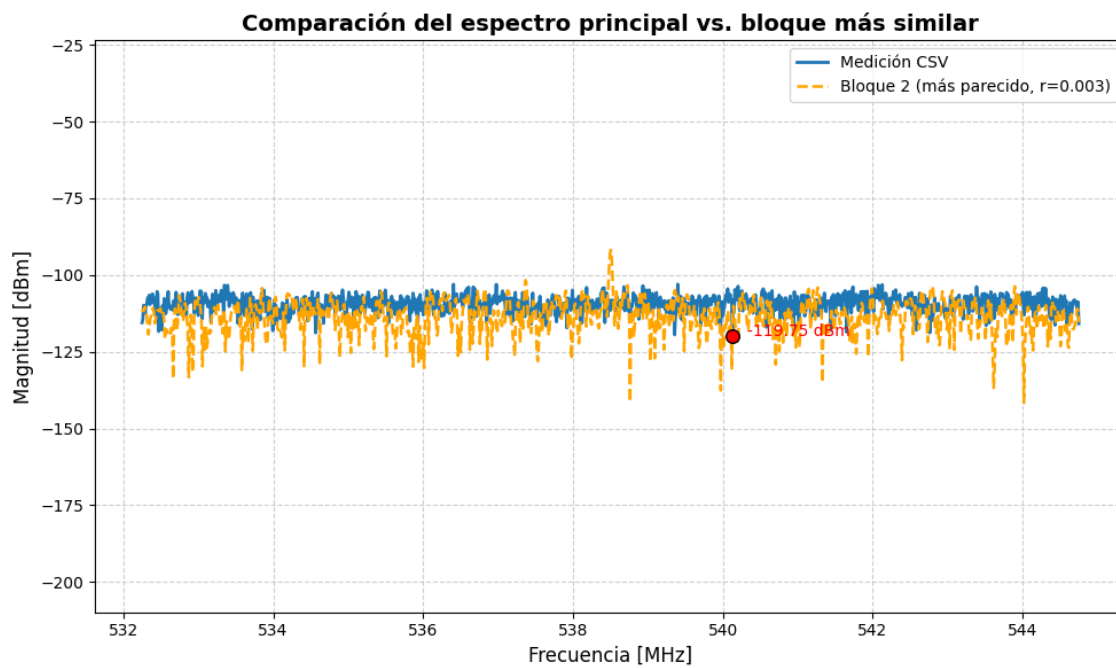


a) archivo CSV del GNU Radio vs el archivo CSV del analizador de espectro.

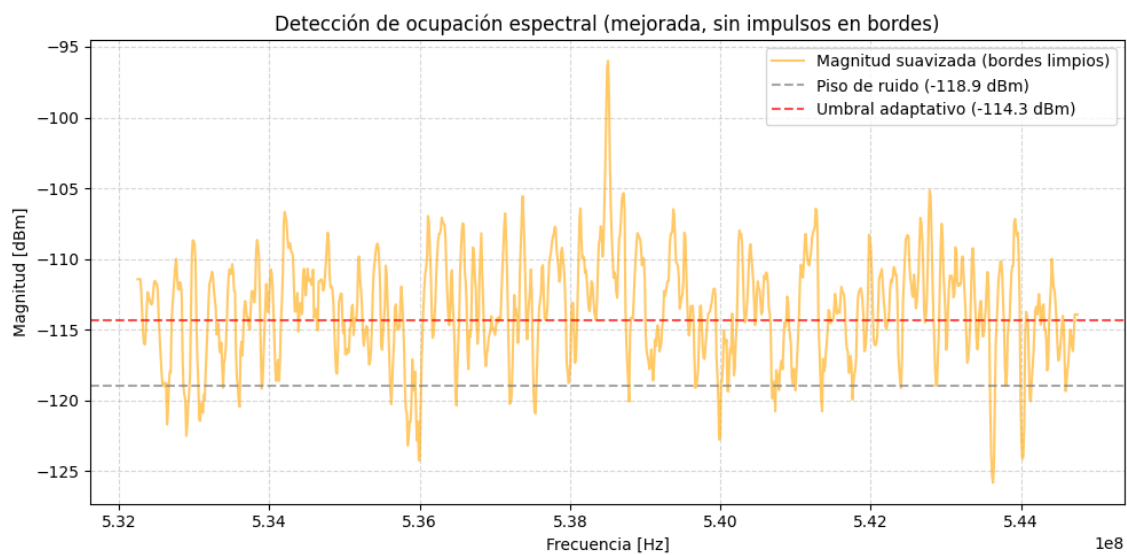


b) Detección de ocupación canal

Figura 9. Sexto barrido

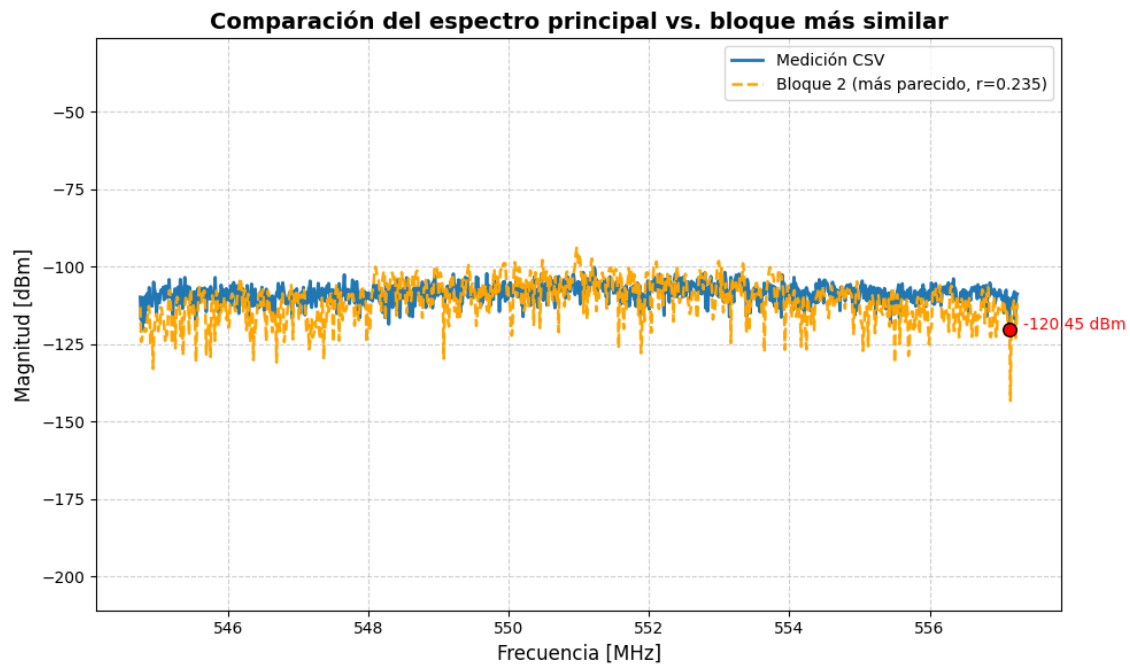


a) archivo CSV del GNU Radio vs el archivo CSV del analizador de espectro.

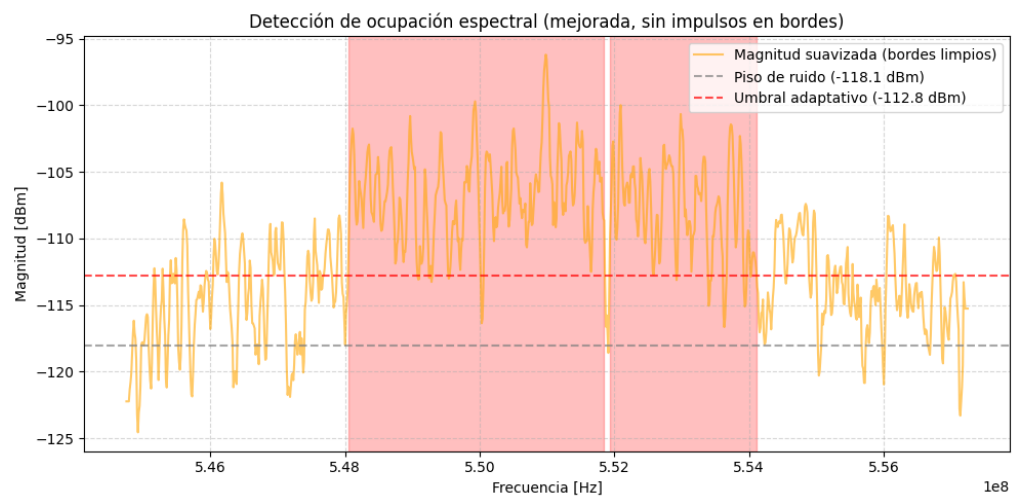


a) Detección de ocupación canal

Figura 10. Séptimo barrido



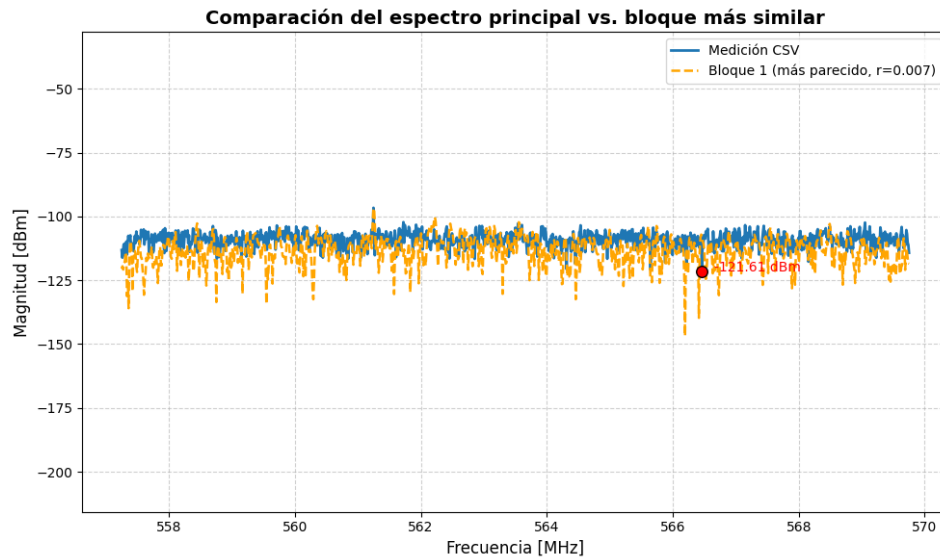
a) archivo CSV del GNU Radio vs el archivo CSV del analizador de espectro.



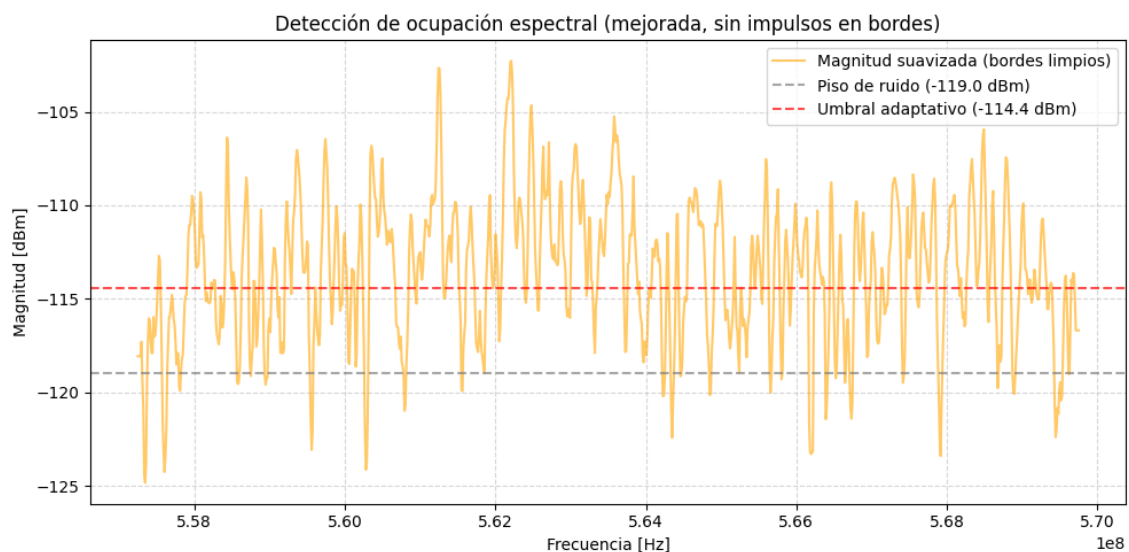
b) Detección de ocupación canal

En el barrido número 7, centro 551 MHz, se detectaron dos bloques ocupados, 548,05 a 551,85 MHz y 551,93 a 554,11 MHz, con un piso de ruido de -118,06 dBm y umbral -112,80 dBm, lo que sugiere transmisiones activas y persistentes en esa sub-banda.

Figura 11. Octavo barrido

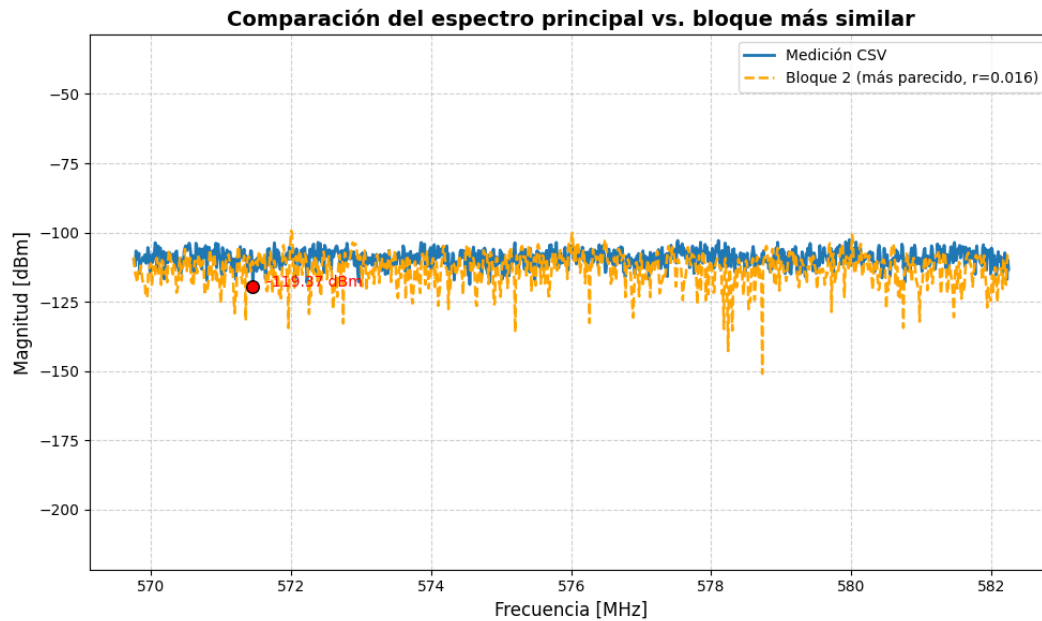


a) archivo CSV del GNU Radio vs el archivo CSV del analizador de espectro.

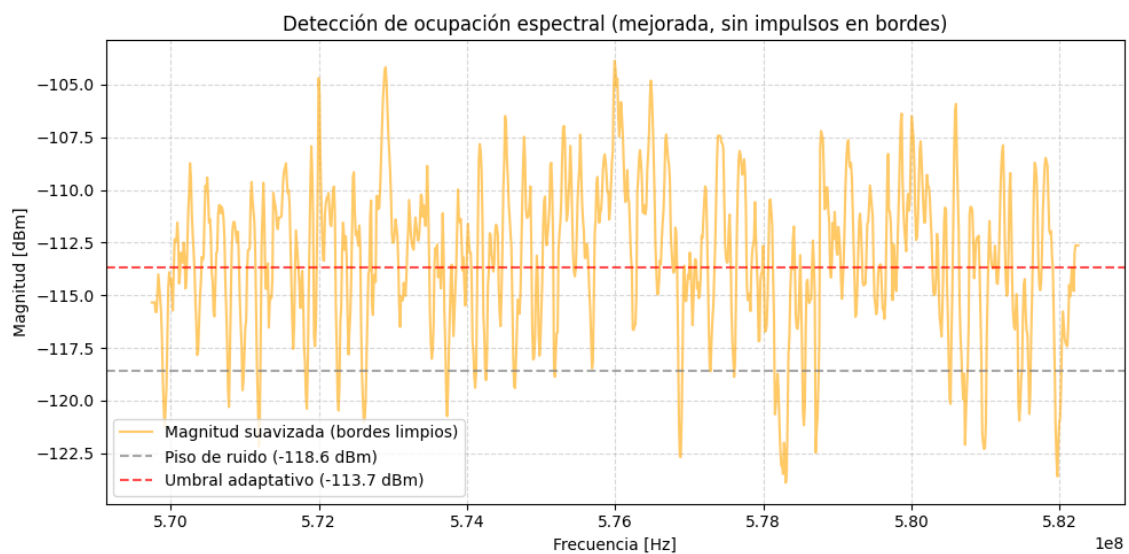


b) Detección de ocupación canal

Figura 12. Noveno barrido

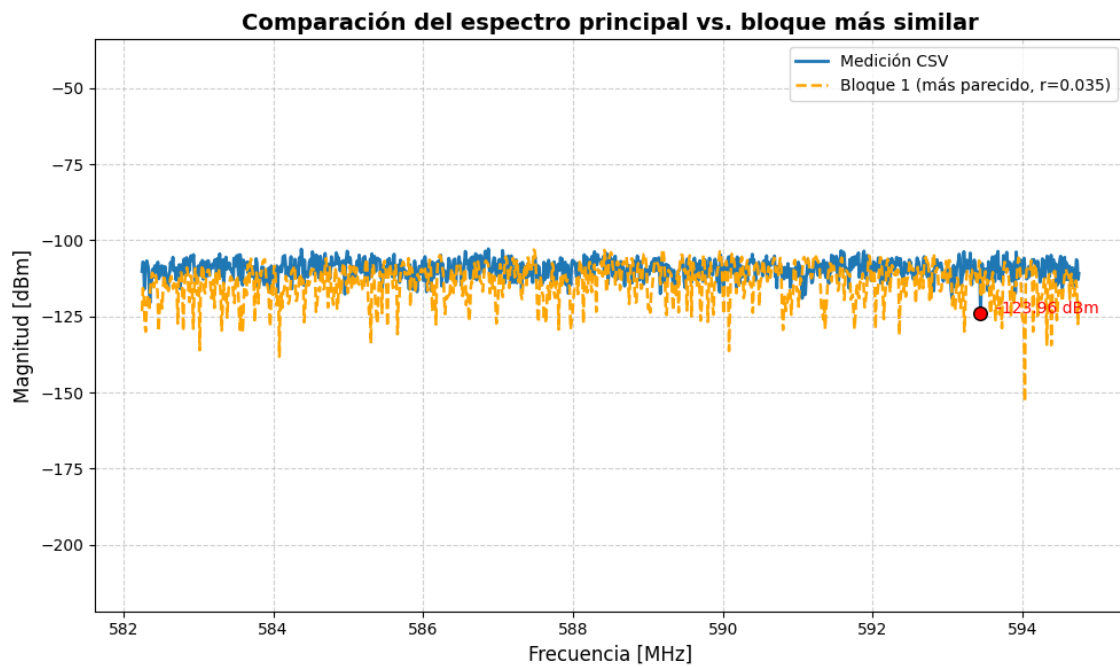


a) archivo CSV del GNU Radio vs el archivo CSV del analizador de espectro.

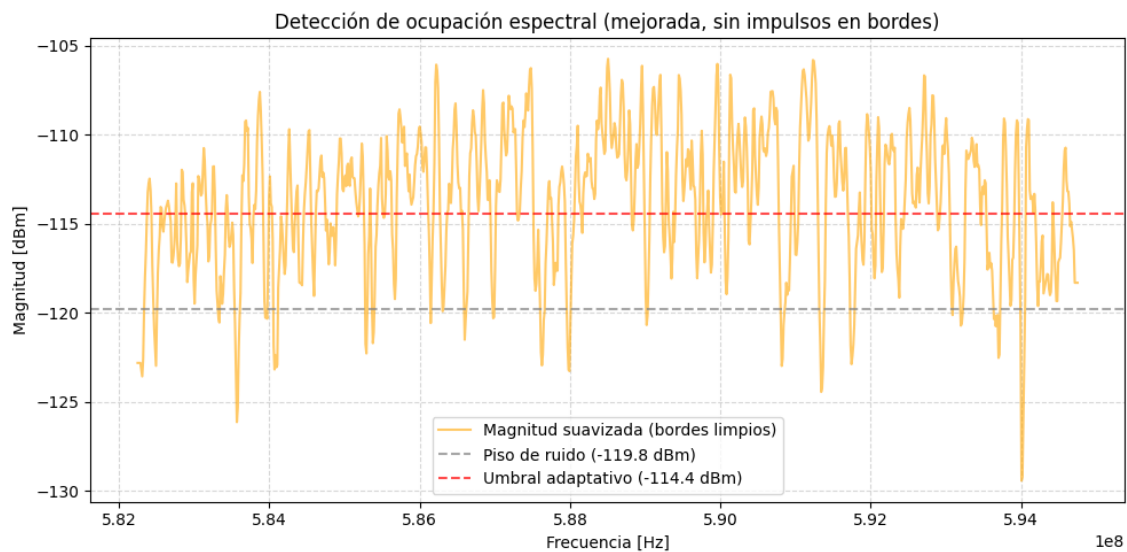


b) Detección de ocupación canal

Figura 13. Décimo barrido

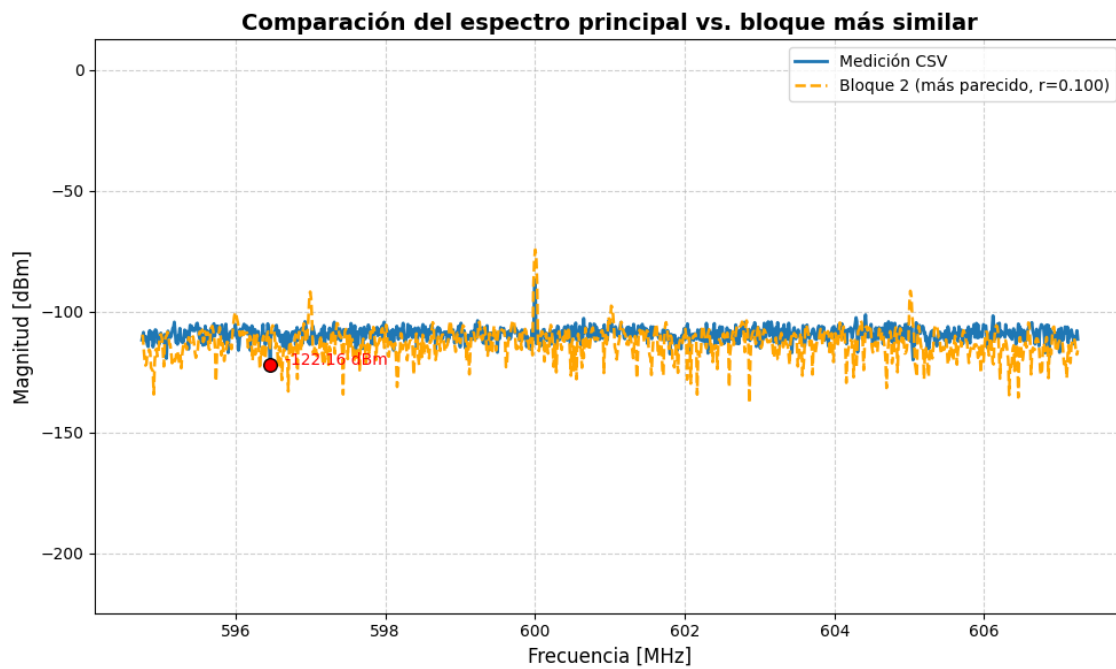


a) archivo CSV del GNU Radio vs el archivo CSV del analizador de espectro.

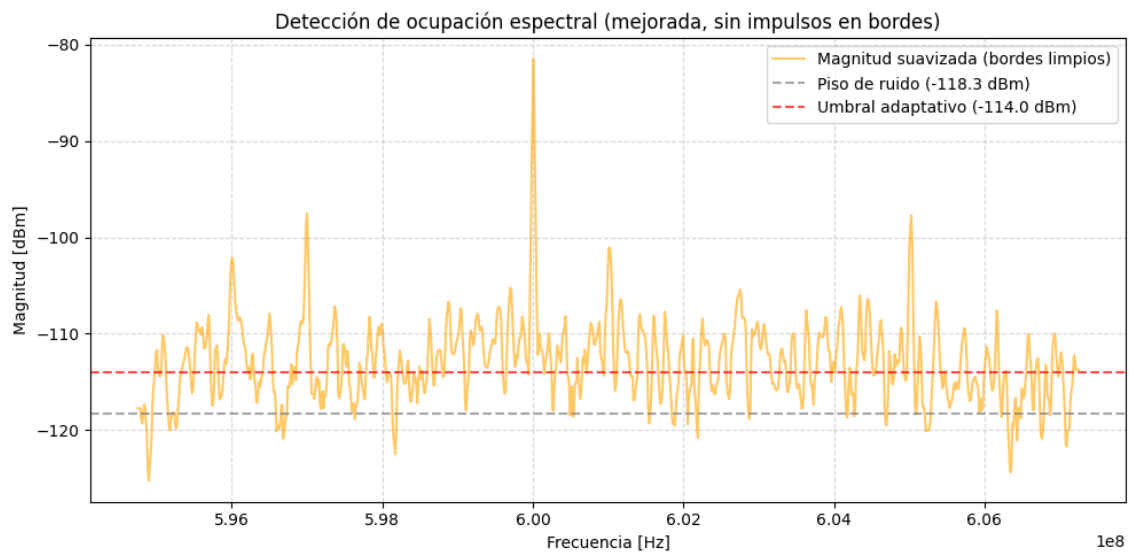


b) Detección de ocupación canal

Figura 14. Undécimo barrido

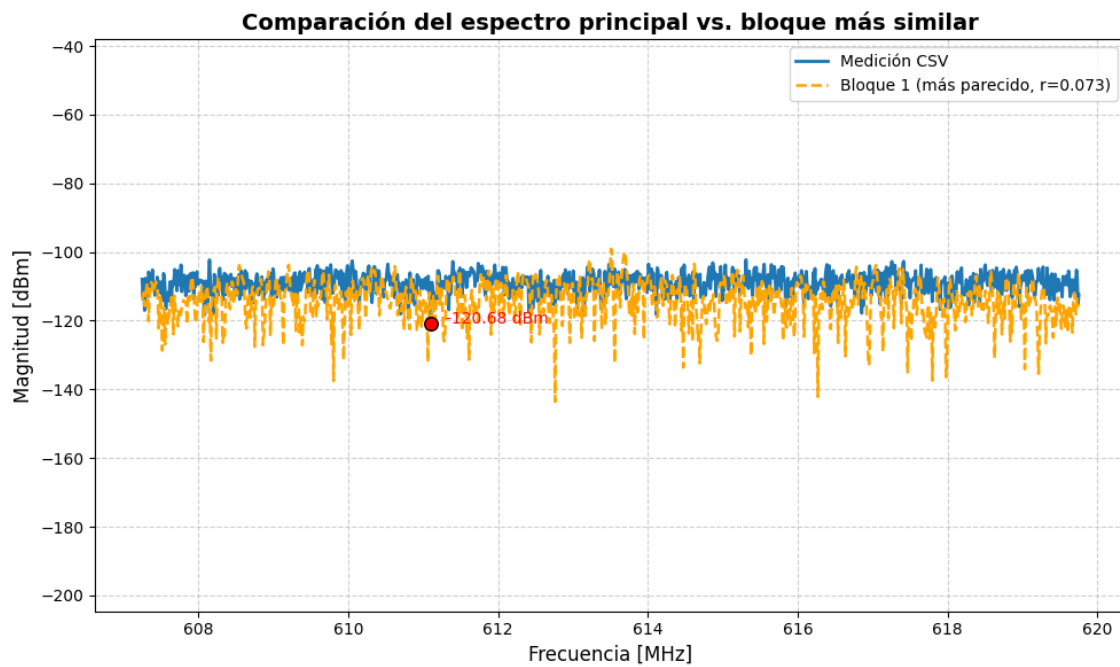


a) archivo CSV del GNU Radio vs el archivo CSV del analizador de espectro.

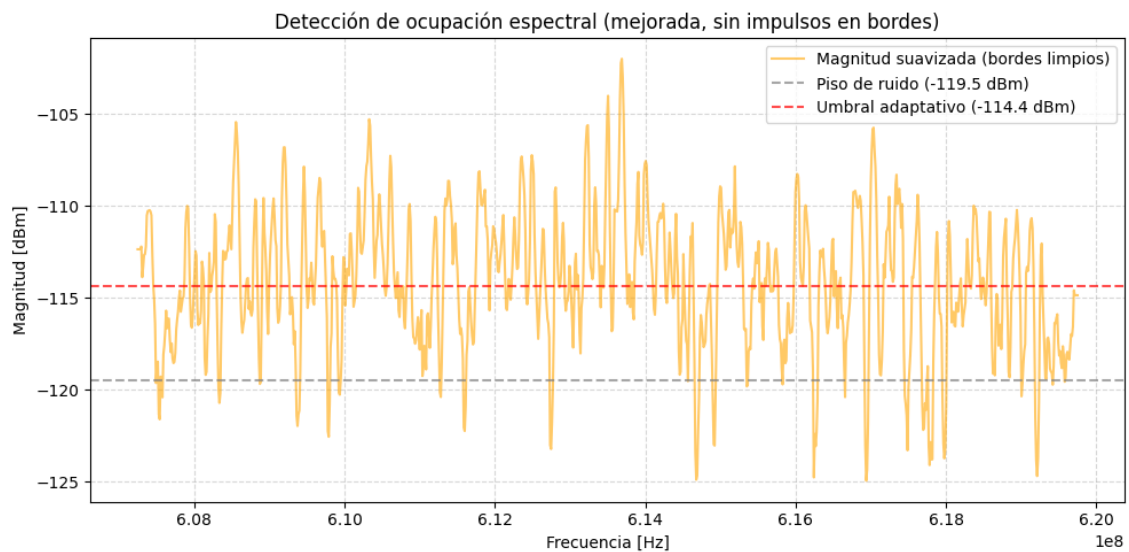


b) Detección de ocupación canal

Figura 15. Décimo segundo barrido

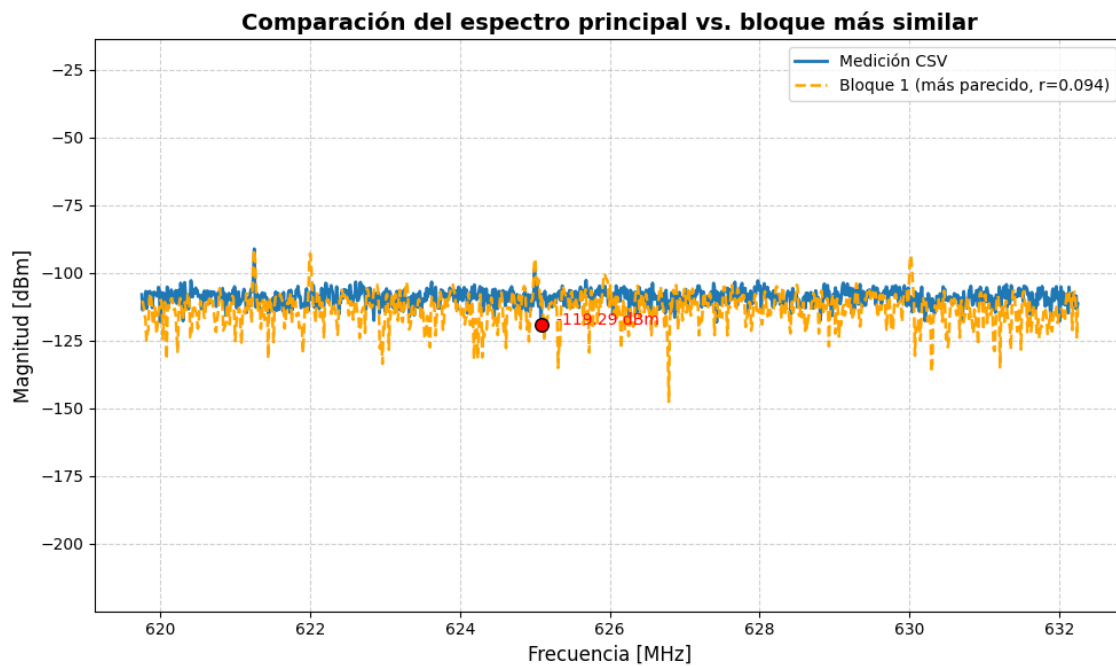


a) archivo CSV del GNU Radio vs el archivo CSV del analizador de espectro.

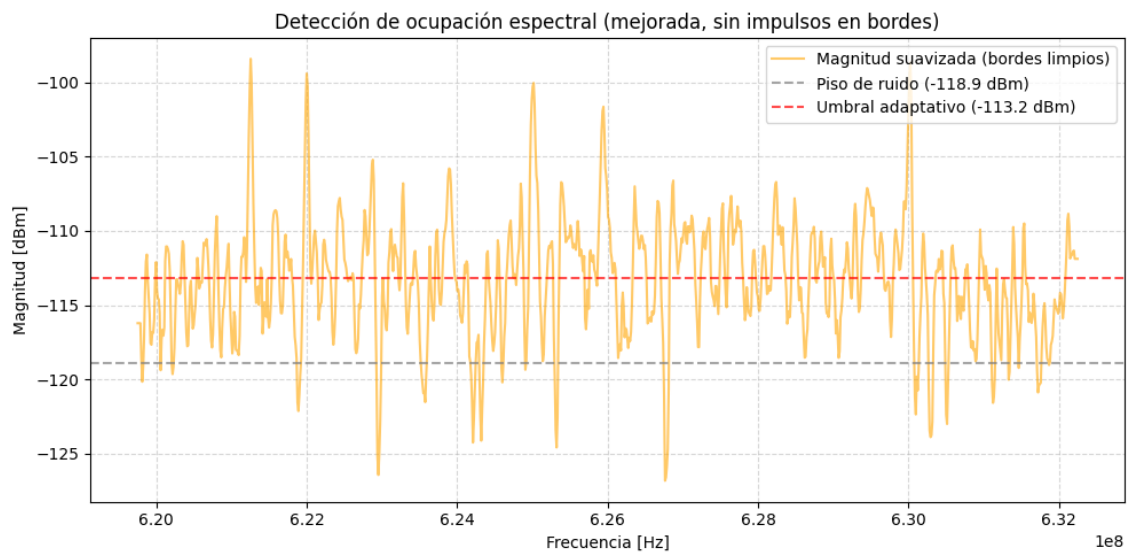


b) Detección de ocupación canal

Figura 16. Décimo tercero barrido

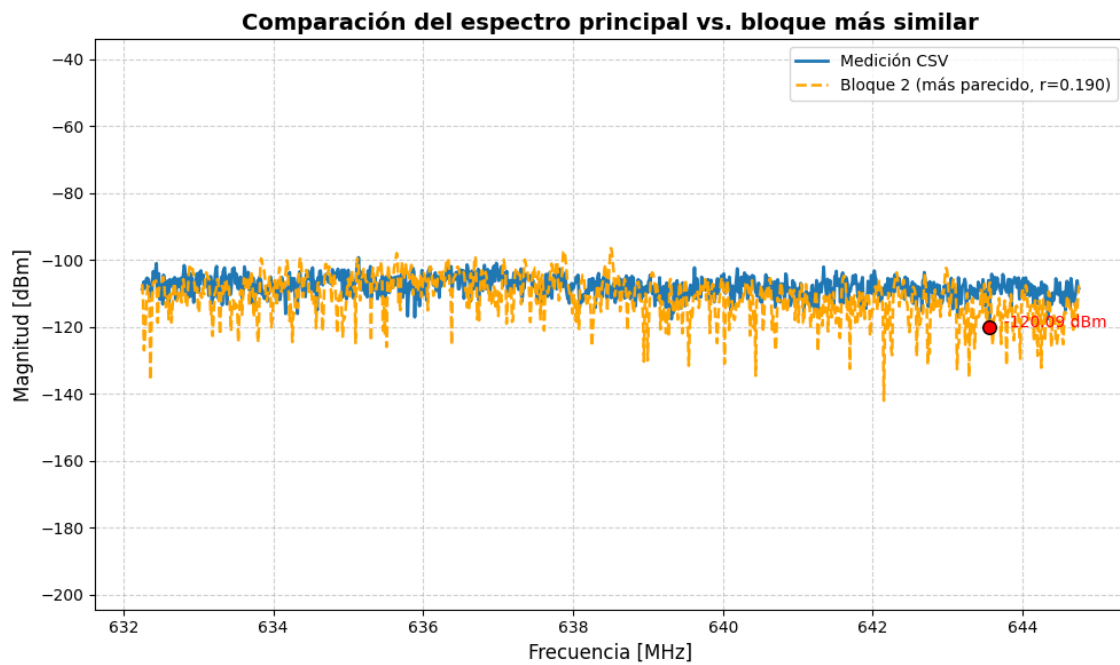


a) archivo CSV del GNU Radio vs el archivo CSV del analizador de espectro.

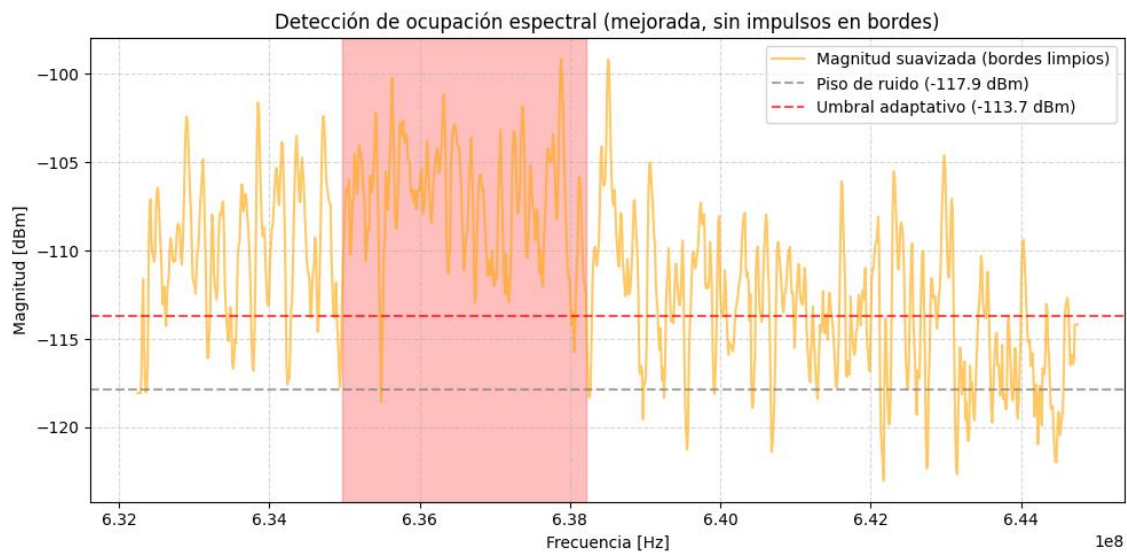


b) Detección de ocupación canal

Figura 17. Décimo cuarto barrido

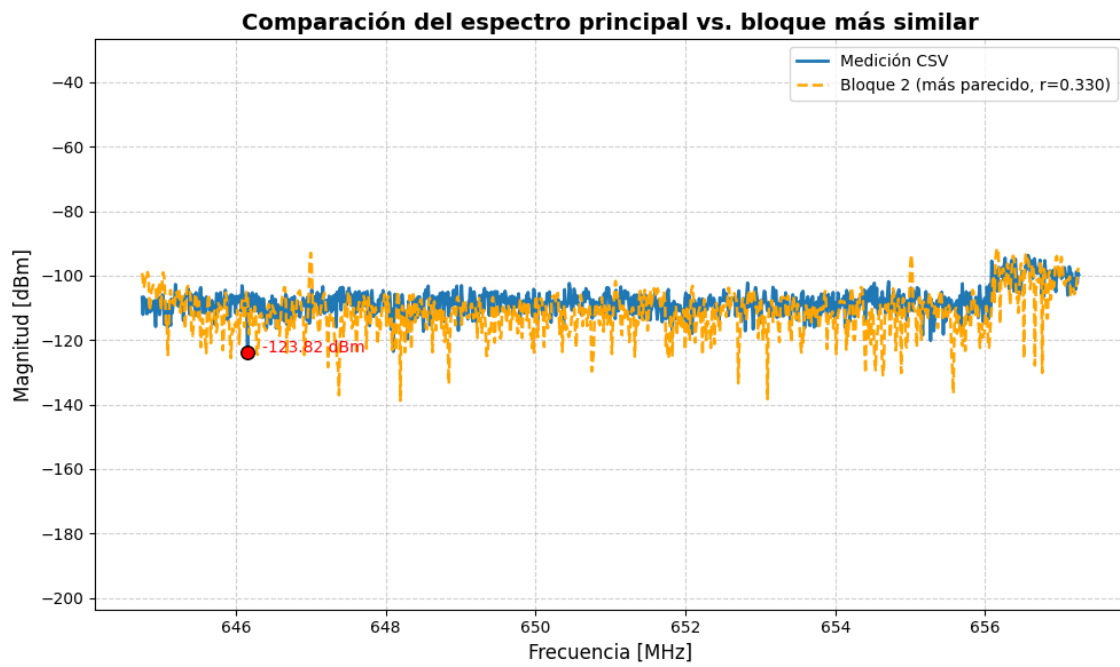


a) archivo CSV del GNU Radio vs el archivo CSV del analizador de espectro.

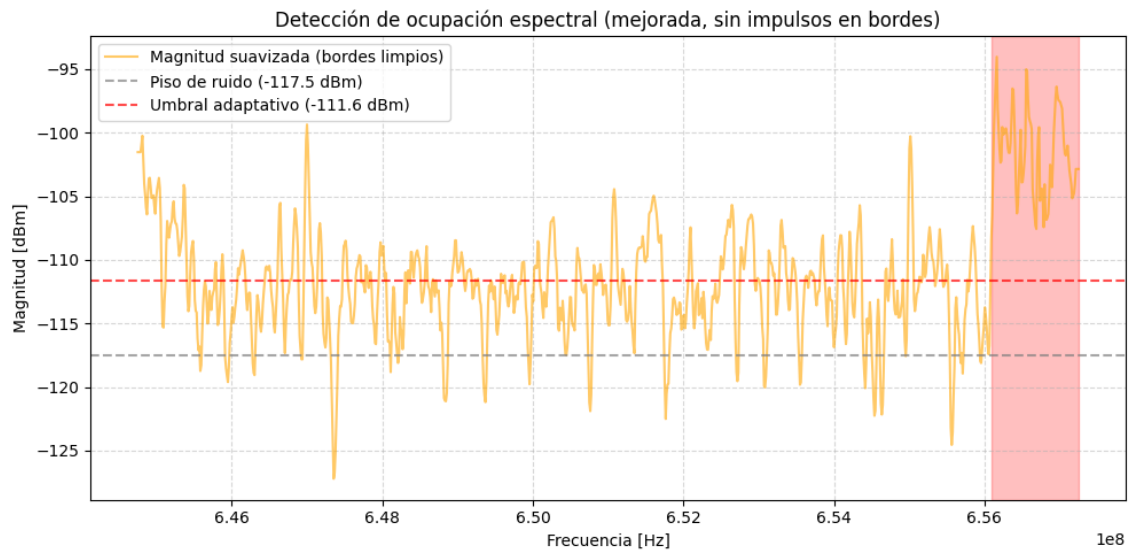


b) Detección de ocupación canal

Figura 18. Décimo quinto barrido

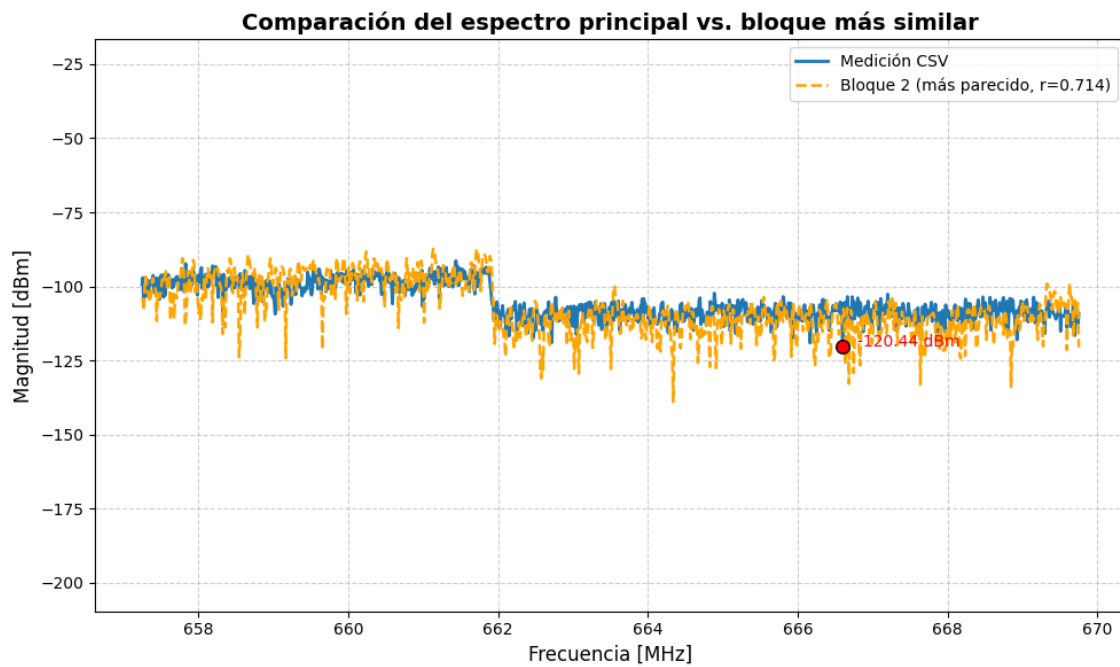


a) archivo CSV del GNU Radio vs el archivo CSV del analizador de espectro.

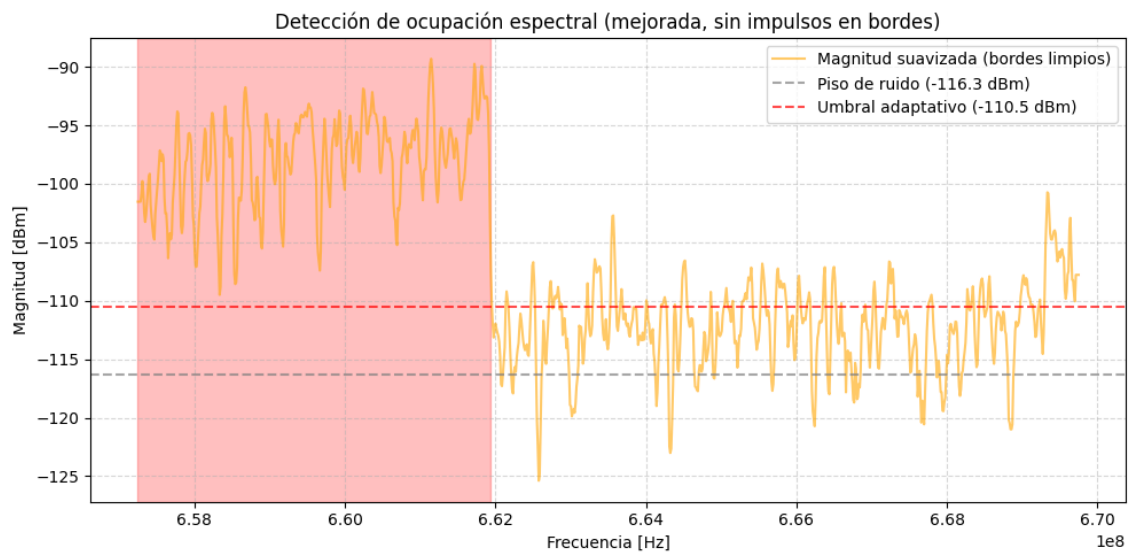


b) Detección de ocupación canal

Figura 19. Décimo sexto barrido



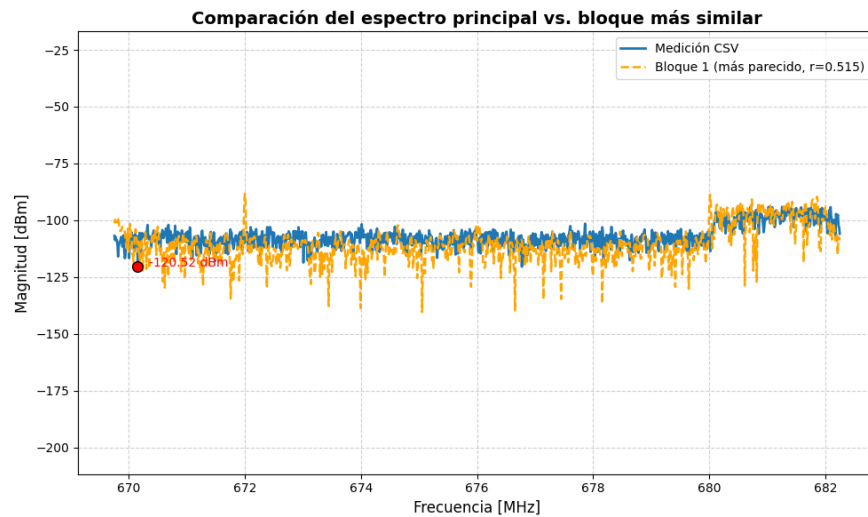
a) archivo CSV del GNU Radio vs el archivo CSV del analizador de espectro.



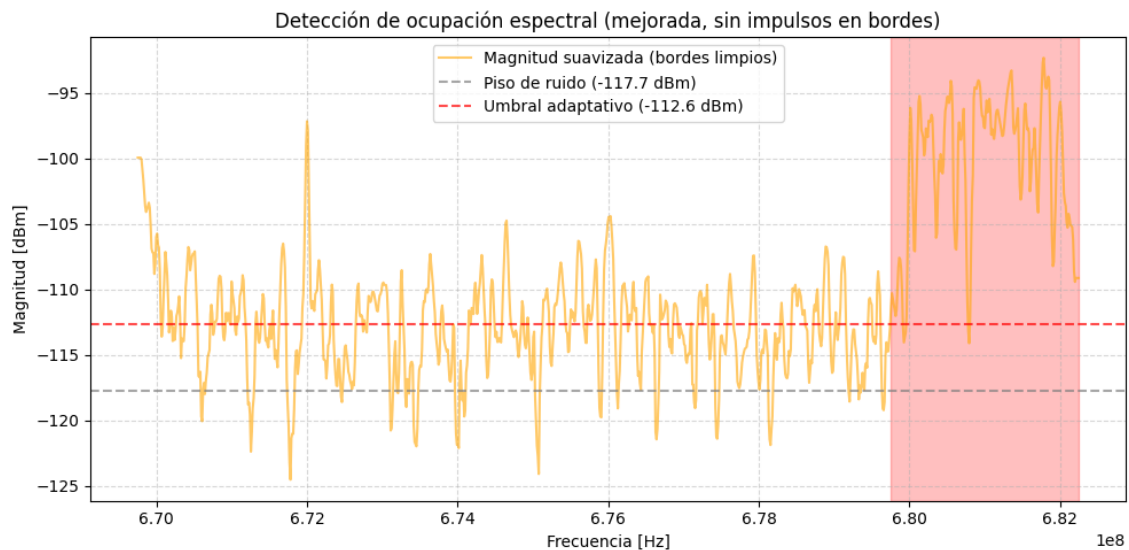
b) Detección de ocupación canal

En el barrido número 16, centro 663,5 MHz, la correlación r fue de 0,7143 y se detectó un canal ocupado entre 657,25 y 661,94 MHz, con un umbral de -110,47 dBm, la alta correlación indica estabilidad de la señal en la ventana.

Figura 20. Décimo séptimo barrido

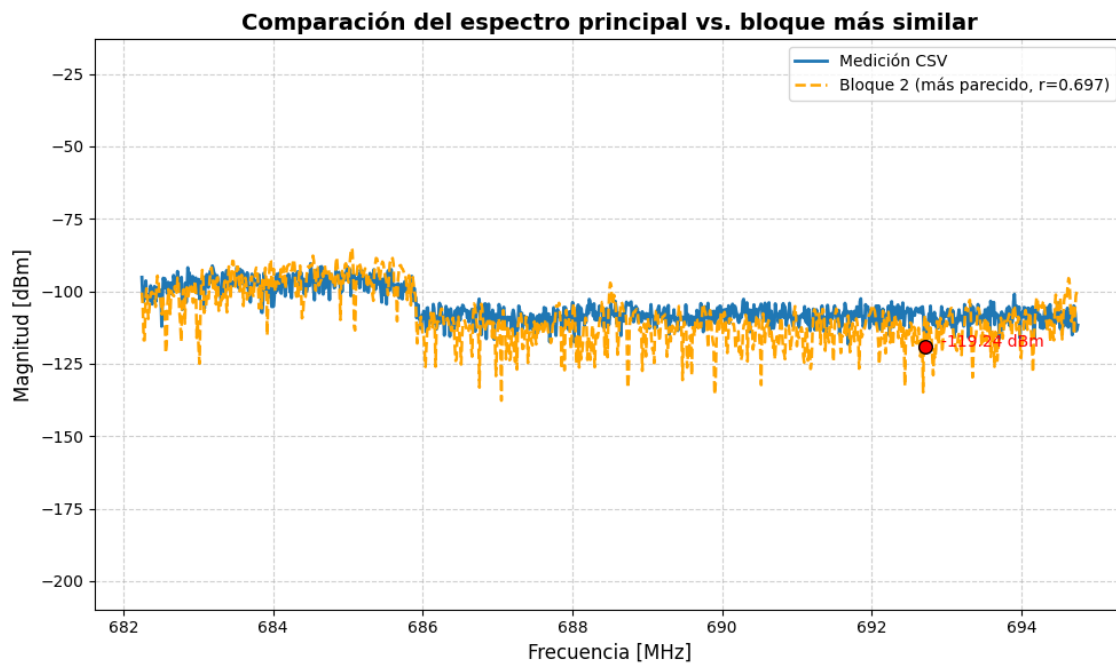


a) archivo CSV del GNU Radio vs el archivo CSV del analizador de espectro.

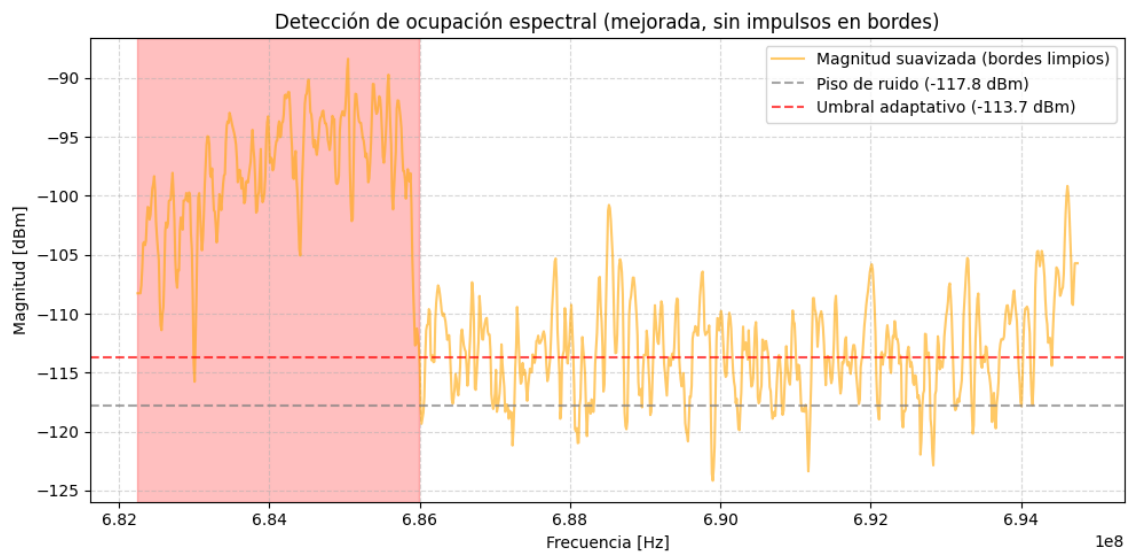


b) Detección de ocupación canal

Figura 214. Décimo octavo barrido

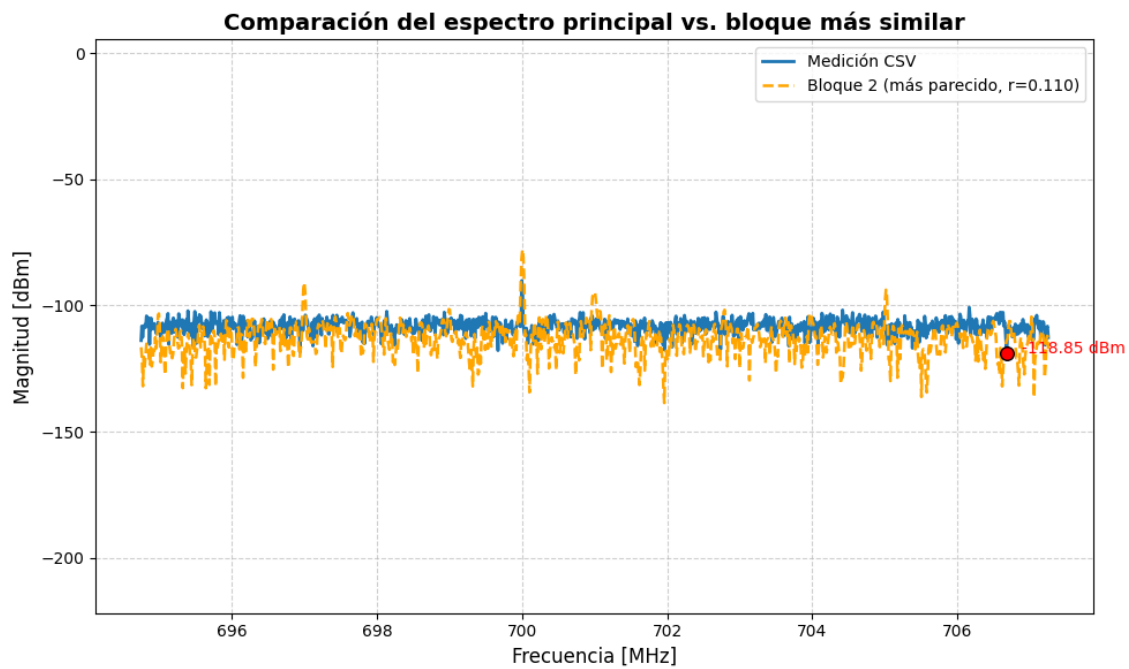


a) archivo CSV del GNU Radio vs el archivo CSV del analizador de espectro.

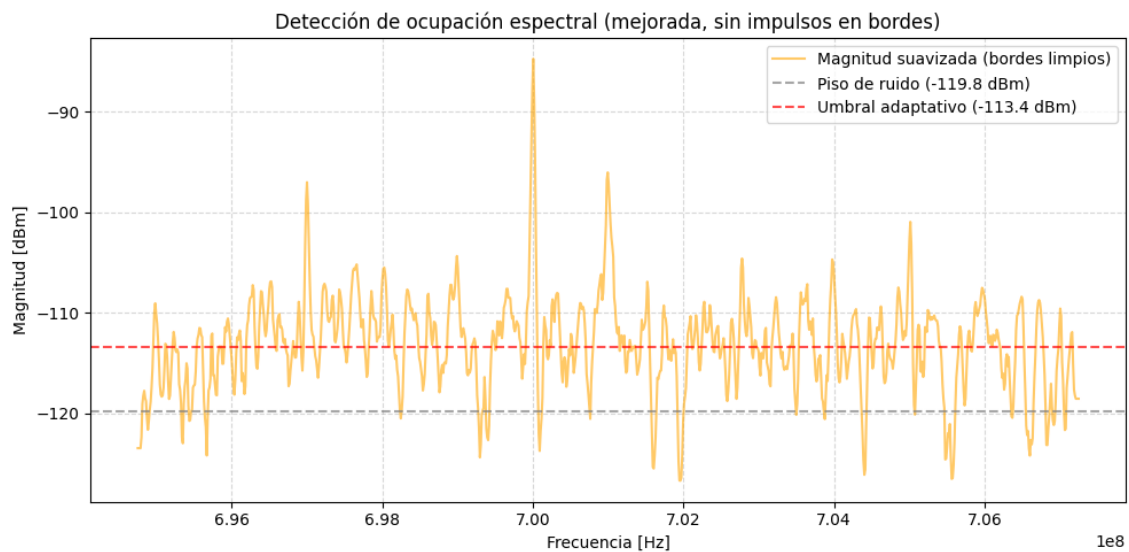


b) Detección de ocupación canal

Figura 22. Décimo noveno barrido



a) archivo CSV del GNU Radio vs el archivo CSV del analizador de espectro.



b) Detección de ocupación canal

Barrido total y unión de los 19 segmentos

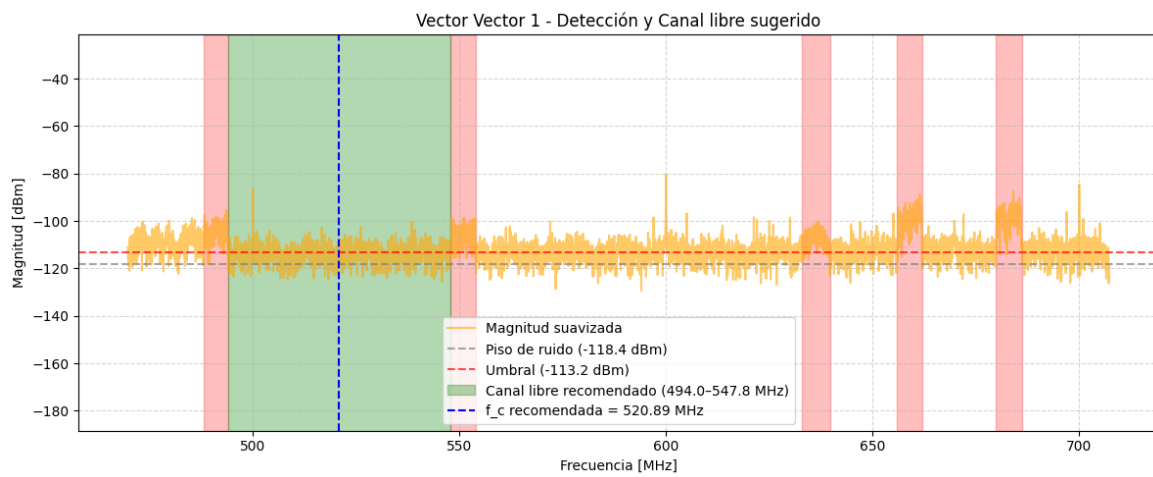


Figura 23. Unión de los 19 barridos utilizando el primer vector de datos.

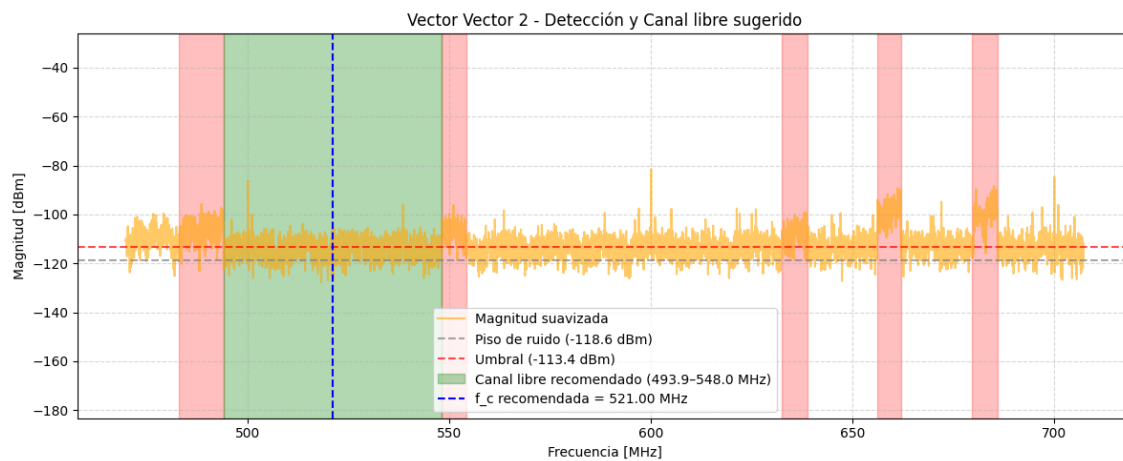


Figura 24. Unión de los 19 barridos utilizando el segundo vector de datos.

Finalmente, en esta parte, se consolida los 19 barridos consecutivos adquiridos con el SDR sobre la banda TVWS, 470–698 MHz para construir una traza espectral continua y, a partir de ella, detectar regiones ocupadas que pueden ser posibles canales de TV o servicios presentes, delimitar

huecos libres suficientemente anchos y proponer una frecuencia central candidata para pruebas de transmisión y recepción.

El procedimiento implementado en Google Colab trabaja únicamente con los archivos CSV exportados por GNU Radio, uno por cada barrido, se extrae los primeros dos vectores válidos por archivo (Vector 1 y Vector 2) y los concatena en orden de frecuencia central. Posteriormente aplica un detector robusto (piso de ruido + umbral adaptativo + agrupación con tolerancia a huecos) para obtener bandas ocupadas y bandas libres.

Entradas: 19 archivos CSV en formato texto con metadatos del flujo y bloques numéricos. Cada archivo corresponde a una frecuencia central (f_c) distinta y contiene, entre otras, las líneas: Frequency, Sample Rate y Vector Length, seguidas de múltiples vectores numéricos de magnitud que corresponde a la salida de FFT con longitud igual a Vector Length.

CONCLUSIONES Y RECOMENDACIONES

1. La estrategia de barrido espectral mediante ventanas solapadas permitió cubrir de forma continua y sin pérdidas el rango completo de TV White Spaces (470–698 MHz), garantizando una representación fiel del entorno radioeléctrico real en la zona de estudio.
2. La unión de los 19 barridos evidenció una alta consistencia entre segmentos adyacentes, lo que confirma la estabilidad del sistema SDR durante el proceso de adquisición y valida la configuración empleada en términos de frecuencia de muestreo, tamaño de FFT y ancho de banda efectivo.
3. El análisis independiente de los vectores consecutivos (Vector 1 y Vector 2) mostró resultados altamente coherentes en cuanto a piso de ruido, umbral de detección y

localización de canales ocupados, lo cual demuestra una buena repetibilidad temporal de las mediciones y reduce la probabilidad de decisiones basadas en eventos transitorios.

4. La estimación adaptativa del piso de ruido, combinada con un umbral estadístico basado en la desviación estándar, permitió diferenciar de manera robusta entre regiones ocupadas y no ocupadas del espectro, incluso en presencia de fluctuaciones de ruido y picos aislados.
5. Los resultados obtenidos confirman que, dentro de la banda TVWS, coexisten múltiples canales claramente ocupados por servicios de radiodifusión y amplios segmentos espectrales con niveles de potencia cercanos al piso de ruido, los cuales pueden ser considerados candidatos viables para comunicaciones alternativas.
6. La identificación de canales libres con anchos superiores a 6 MHz y baja potencia promedio constituye una evidencia experimental sólida de la disponibilidad efectiva de espacios blancos en el entorno analizado, respaldando el uso de esta banda para aplicaciones de comunicación en zonas rurales.
7. Esta fase de verificación espectral establece un vínculo directo entre la caracterización teórica del entorno y la implementación práctica del sistema SDR, proporcionando criterios objetivos para la selección de frecuencias y reduciendo la incertidumbre en las pruebas de transmisión posteriores.