



ETSETB

Escola Tècnica Superior  
d'Enginyeria de Telecomunicació de Barcelona

# ESTUDIO DE LAS INTERFERENCIAS EN DETECCIÓN DE BLANCOS PARA UN SISTEMA LIDAR FMCW

Autor: Sergio Ramírez Castillo

Director: Óscar Batet Torrell

Director Ponente: Víctor Federico Dios Otín



A **mis padres, Antonio y Dolores**, que me han regalado mi educación, tanto académica como personal, la posesión más preciada que nunca tendré.

A **Óscar Batet**, que ha sido mi guía durante la elaboración de este PFC.

A los **Waltrapas**, los mejores amigos que se pueden desear.

Y por supuesto a **Ana**, ya que sin ella todo lo demás no tiene sentido.



<b>Capítulo 1: Introducción.....</b>	<b>7</b>
<b>1.1 Proyecto Completo: detección de gases en medios continuos.....</b>	<b>7</b>
1.1.1 WMS: Wavelength Modulation Spectroscopy .....	8
1.1.2 FMCW: Frequency Modulated Continuous-Wave.....	9
1.1.3 Sistema completo.....	10
1.1.4 Aplicaciones .....	11
<b>1.2 Objetivos .....</b>	<b>12</b>
<b>Capítulo 2: Análisis Teórico .....</b>	<b>13</b>
<b>2.1 Lidar.....</b>	<b>13</b>
2.1.1 Principio de funcionamiento: Ecuación Lidar .....	14
2.1.2 Ejemplos de Lidar.....	14
<b>2.2 FMCW .....</b>	<b>17</b>
2.2.1 Principio de funcionamiento.....	17
2.2.2 Análisis teórico .....	19
2.2.3 Diagrama de bloques .....	23
2.2.4 Montaje real: “diente de sierra” frecuencial y resolución .....	23
2.2.5 Fase inicial y ruido aditivo .....	25
2.2.6 FMCW en medios continuos .....	26
2.2.7 Caso particular: Dos blancos .....	29
<b>Capítulo 3: Simulación .....</b>	<b>33</b>
<b>3.1 Generación de un periodo de la señal sweep .....</b>	<b>33</b>
3.1.1 Distancia máxima y frecuencia inicial ficticia .....	34
3.1.2 Generación del vector que contiene la señal seep .....	35
<b>3.2 Procedimientos de simulación .....</b>	<b>35</b>
3.2.1 Vector recibido: Retardo del vector enviado.....	35
3.2.2 Multiplicación de las señales enviada y recibida.....	36
3.2.3 Suma de n periodos de la señal sweep .....	36
3.2.4 Cálculo de la FFT y presentación .....	37
<b>3.3 Simulaciones de un blanco .....</b>	<b>38</b>
<b>3.4 Interacción de dos blancos .....</b>	<b>41</b>
3.4.1 Errores en amplitud y distancia .....	42
<b>3.5 Interacción de tres blancos .....</b>	<b>47</b>
3.5.1 Dos blancos fijos .....	48
3.5.2 Tres blancos equidistantes .....	54
<b>Capítulo 4: Montaje y técnica experimental.....</b>	<b>59</b>
<b>4.1 Montaje general .....</b>	<b>59</b>
4.1.1 Programa en LabView del laboratorio.....	61
4.1.2 Programa de visualización de los datos.....	61
<b>4.2 Procedimientos de Laboratorio .....</b>	<b>62</b>
4.2.1 Alineación del sistema.....	62
4.2.2 Línea de retardo .....	64
4.2.3 Enventanado .....	65
4.2.4 Igualación de potencias .....	67

4.2.5	Normalización .....	70
<b>Capítulo 5: Resultados .....</b>		<b>71</b>
<b>5.1</b>	<b>Medidas de un blanco .....</b>	<b>71</b>
<b>5.2</b>	<b>Medidas de dos blancos .....</b>	<b>73</b>
<b>5.3</b>	<b>Medidas de tres blancos .....</b>	<b>79</b>
<b>5.4</b>	<b>Medidas de blanco disperso .....</b>	<b>80</b>
5.4.1	Estudio y optimización del Background.....	81
5.4.2	Resta de background.....	84
5.4.3	Montaje alternativo.....	85
5.4.4	Diagrama de reflexión del cristal.....	87
<b>5.5</b>	<b>Medidas de coherencia del láser .....</b>	<b>88</b>
5.5.1	Medidas de dos blancos muy separados .....	88
5.5.2	Medidas de dos blancos desplazados.....	91
5.5.3	Medidas de separaciones consecutivas.....	92
<b>Capítulo 6: Conclusiones .....</b>		<b>95</b>
<b>Capítulo 7: Bibliografía .....</b>		<b>97</b>
<b>Anexo I: Procedimientos.....</b>		<b>99</b>
<b>Anexo II: Archivos de MatLab .....</b>		<b>103</b>
<b>Anexo III: Aparatos del laboratorio.....</b>		<b>149</b>
<b>Anexo IV: Características de los componentes.....</b>		<b>159</b>

# **CAPÍTULO 1: INTRODUCCIÓN**

Multitud de aplicaciones actuales necesitan de técnicas de detección de gases o aerosoles. El uso de sistemas de detección con acciones automáticas adecuadas antes de alcanzar niveles peligrosos de los gases detectados, puede ayudar a evitar explosiones o daños por exposición a gases tóxicos en personal. Además de en seguridad, la detección de gases o aerosoles atmosféricos puede ser útil para hacer previsiones meteorológicas más exactas.

Antes de que las tecnologías de detección de gases mediante técnicas lidar estuviesen suficientemente maduras, se usaban detectores químicos que identificaban el tipo de gas y su concentración mediante el estudio de ciertas reacciones químicas. Este sistema de detección tiene dos problemas fundamentales: en primer lugar las concentraciones del gas a detectar deben ser suficientemente grandes como para poder causar la reacción química que lo detecte; en segundo lugar, la medida de estos gases debe hacerse *in situ*, por lo que no es viable medir gases a distancia, como por ejemplo gases atmosféricos.

Gracias a las tecnologías de detección de gases mediante ondas electromagnéticas (ya sea mediante radar o lidar) es posible detectar gases a concentraciones bajas a distancia. La mayoría de los sistemas de detección de gases mediante lidar usados hasta la actualidad están basados en láser pulsado, ya que se necesitan pulsos de alta potencia para detectar pequeñas concentraciones de gas. Precisamente, la necesidad del envío de estos pulsos de alta potencia hace que estos sistemas sean ineficientes energéticamente. Idealmente, esta detección de gases debería hacerse mediante un sistema lidar de onda continua, ya que tendría las mismas ventajas de la detección mediante ondas electromagnéticas (medida de concentraciones bajas de gas a distancia) y además sería un sistema energéticamente eficiente. Estos sistemas lidar de onda continua requieren en general técnicas de procesado de señal más complicadas que las del caso de lidar pulsado.

Este Proyecto Final de Carrera se enmarca dentro de un Proyecto más global dedicado a la detección de gases mediante un sistema lidar de onda continua utilizando las técnicas WMS y FMCW.

El objetivo principal de este PFC es estudiar la interacción de dos o más blancos a distancias comparables a la distancia de resolución, tratando de establecer las bases para el estudio de blancos continuos.

## **1.1 Proyecto Completo: detección de gases en medios continuos**

El objetivo principal del Proyecto en el que se enmarca este PFC es el estudio e implementación de un sistema lidar de onda continua con resolución en distancia, capaz de detectar y medir concentraciones de cierto gas en una distribución continua de este. Para conseguir este objetivo, se estudian dos técnicas que, cada una por separado, resuelven parte de los objetivos del proyecto. Las técnicas utilizadas son

WMS (Wavelength Modulation Spectroscopy) y FMCW (Frequency Modulated Continuous-Wave). WMS se utiliza para medir concentraciones de gas en una zona concreta. FMCW se utiliza para detectar blancos, midiendo la distancia a la que se encuentran. La combinación de estas técnicas da lugar a otra más compleja capaz de detectar gases en distribuciones continuas en el espacio y obtener los valores de concentración con resolución en distancia.

### 1.1.1 WMS: Wavelength Modulation Spectroscopy

Cuando la luz (por ejemplo de un láser) atraviesa un gas, parte de ella es absorbida antes de llegar al receptor. La potencia absorbida por el gas depende de varios factores, como su volumen y concentración. Además, la potencia absorbida no es igual para todas las longitudes de onda de emisión, presentando líneas de absorción que son características de cada gas, como se observa en la Figura 1.

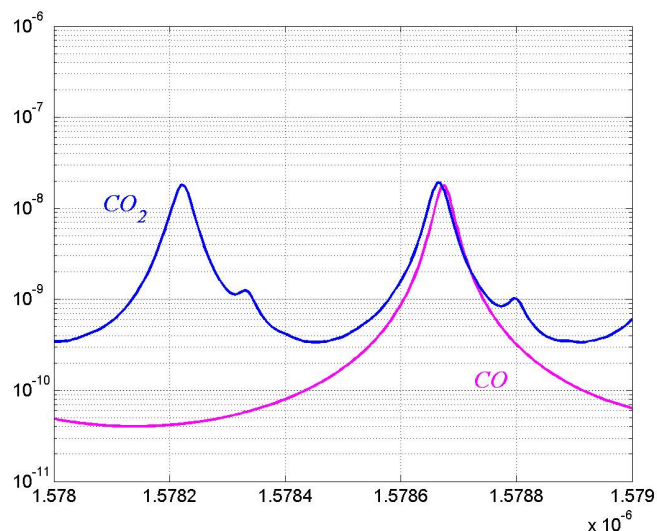


Figura 1: Absorción de los gases CO y CO<sub>2</sub> para diferentes frecuencias.

Estas líneas de absorción son conocidas para cada gas y, por lo tanto, sintonizando la longitud de onda del láser en una línea de absorción del gas bajo estudio, es posible medir su amplitud y determinar sus características, como la concentración de ese gas en la zona que atraviesa la señal luminosa.

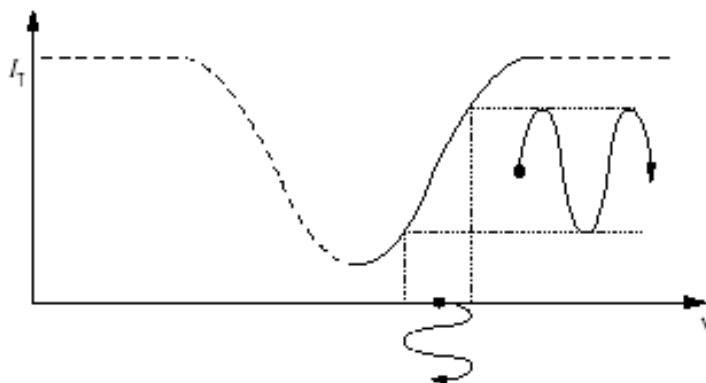


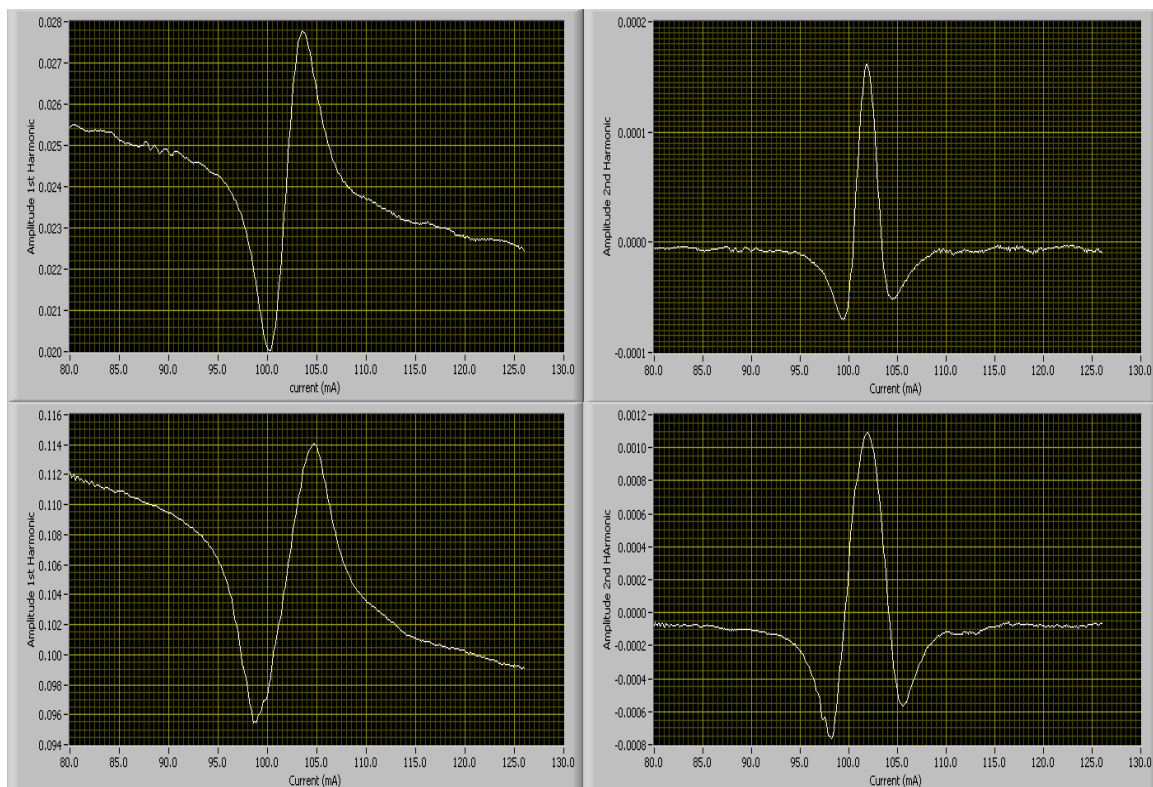
Figura 2: Amplitud de respuesta a una sinusoide centrada a una longitud de onda próxima a una línea de absorción.



La técnica WMS se basa precisamente en la medición de la amplitud de las líneas de absorción. El láser es modulado por una corriente senoidal de amplitud constante. Esta corriente modula potencia y longitud de onda simultáneamente de la señal luminosa emitida. Como se observa en la Figura 2, la amplitud de la fluctuación de la potencia absorbida por el gas depende de la longitud de onda central de la señal senoidal.

Representando la amplitud del primer y segundo armónico de la señal resultante respecto de la longitud de onda central de emisión, obtenemos gráficas que se asemejan a la primera y la segunda derivada respectivamente de la Lorentziana que representa el punto de absorción. Para el caso del primer armónico, además se obtiene esta gráfica superpuesta a una recta, debido a que la potencia de la señal luminosa emitida no es constante, sino que también está modulada por la corriente:

Figura 3: Representación de la amplitud del primer y segundo armónico en un sistema WMS.



En la Figura 3 están representadas las amplitudes del primer y segundo armónico en función de la corriente un sistema WMS. Se observa que las curvas del primer armónico están superpuestas a sendas rectas debidas a que la corriente de entrada del láser modula la potencia de salida además de su longitud de onda.

Para el caso de un gas en espacio libre, y en pequeñas concentraciones, la amplitud del primer y segundo armónico dependen directamente de la amplitud de la línea de absorción, y por lo tanto de la concentración del gas de estudio.

### 1.1.2 FMCW: Frequency Modulated Continuous-Wave

FMCW es una técnica comúnmente utilizada en radares. Sirve para la detección de uno (o más) blancos sólidos y medir la distancia a la que se encuentra. Esta técnica

también puede usarse en lidar (con ciertas ventajas, como veremos en capítulos posteriores) para la detección de gases en regiones concretas.

Esta técnica se basa en el envío de una senoide cuya frecuencia aumenta linealmente con el tiempo. Aunque idealmente la frecuencia instantánea debe aumentar indefinidamente, para que el sistema sea implementable esta frecuencia sólo aumenta hasta cierta frecuencia máxima, volviendo a la frecuencia inicial para empezar de nuevo. De esta manera, la frecuencia instantánea de la señal es un diente de sierra frecuencial, tal y como se observa en la Figura 4.

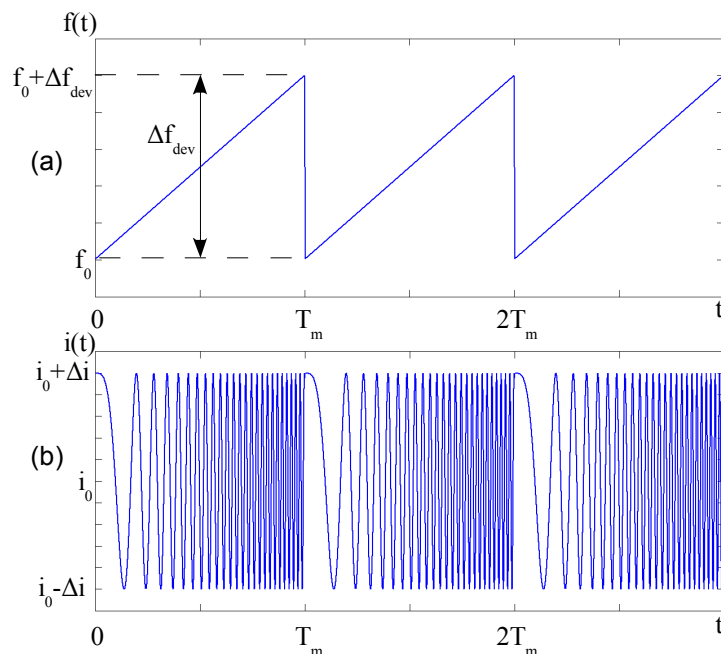


Figura 4: Diagrama temporal de la variación de frecuencia en la señal enviada (a); forma de onda de la señal enviada (b).

La señal reflejada por los blancos será esta misma, pero retardada proporcionalmente a la distancia a la que se encuentra cada blanco. Mezclando la señal recibida con la enviada, podemos obtener mediante filtrado la componente resta frecuencial. Conociendo la velocidad a la que cambia esta frecuencia (pendiente de la rampa frecuencial), se puede calcular el retardo sufrido por la señal al rebotar en el blanco, y por lo tanto la distancia a la que se encuentra.

### 1.1.3 Sistema completo

La combinación de las técnicas WMS y FMCW dan como resultado la representación de la concentración del gas de estudio respecto de la distancia. Estas dos técnicas son compatibles entre sí, ya que mientras que WMS utiliza la modulación de la longitud de onda del láser para transmitir la información (aunque también module la envolvente, no es la parte útil de la señal), la técnica FMCW modula la envolvente de la señal luminosa, dando lugar a una señal que combina el efecto de ambas, tal y como se observa en la Figura 5.

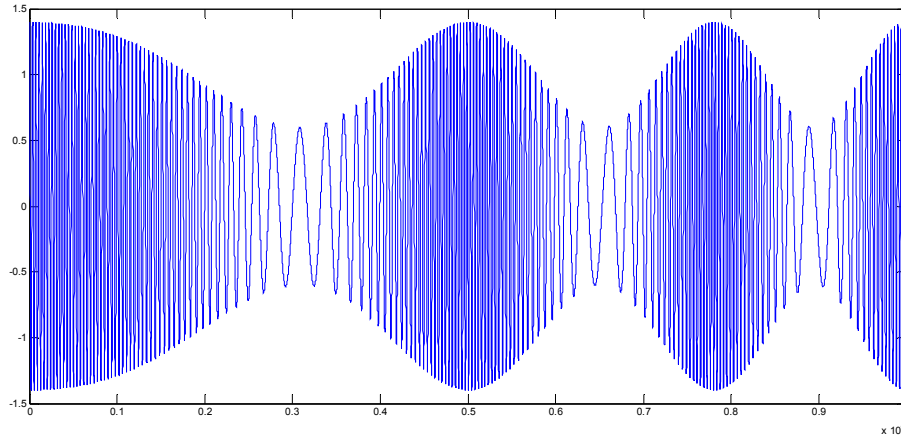


Figura 5: Representación de la señal luminosa a la salida del láser. Eje de abscisas: tiempo en segundos. Eje de ordenadas: amplitud normalizada.

Aunque en recepción la potencia recibida no es la misma que en el caso WMS debido a que la señal FMCW viaja en la envoltura de la señal luminosa, mediante procesado es posible separar la contribución de ambas técnicas. En la Figura 6 se observa el efecto de la fluctuación de potencia descrito para la Figura 2, pero para el caso en que la senoide emitida aumenta en frecuencia.

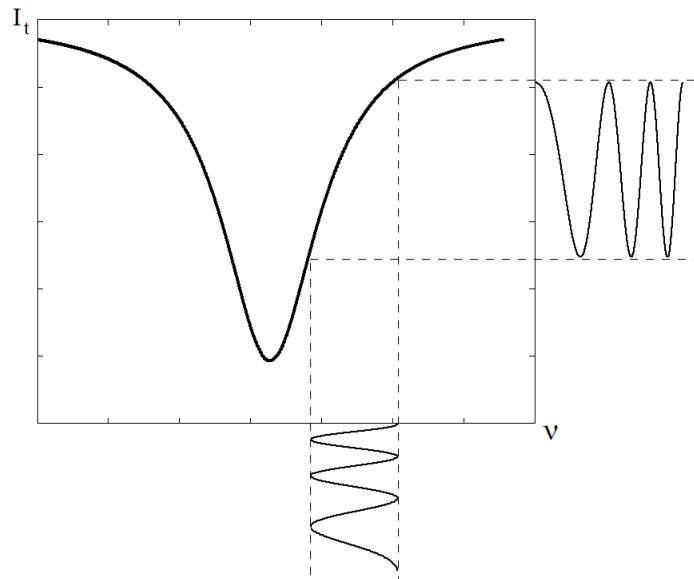


Figura 6: Amplitud de respuesta a una señal que aumenta en frecuencia centrada a una longitud de onda próxima a una línea de absorción.

Mediante el procesado de los datos obtenidos como resultado de cada técnica, es posible estimar la distribución de la concentración del gas estudiado con respecto de la distancia.

### 1.1.4 Aplicaciones

El sistema propuesto tendría muchas aplicaciones sobre todo en el campo de la seguridad. Se podrían colocar varios láseres siguiendo las tuberías en una planta química, que monitoree en todo momento la concentración de gases nocivos. El sistema podría no sólo detectar un escape del gas que se está midiendo, sino que

además podría señalar su ubicación exacta, a la vez que evalúa la magnitud del escape de gas. También se podrían localizar escapes de gases en oleoductos, montando el sistema lidar en un avión que recorra toda la longitud del oleoducto.

Además de gases, el sistema podría detectar también aerosoles e incluso polvo, de manera que otra posible aplicación sería la monitorización de las partículas en suspensión en una mina, haciendo posible, por ejemplo, evaluar exactamente en qué galerías no sería recomendable la entrada de trabajadores.

En el campo del estudio de la atmósfera, el sistema propuesto también tiene aplicaciones inmediatas, ya que podría medir la concentración de cualquier gas o aerosol, y confeccionar la distribución de éste en el espacio.

## 1.2 Objetivos

La implementación del sistema descrito en el apartado 1.1 requiere del estudio detallado de las técnicas WMS y FMCW. Además, ya que el sistema se debe diseñar para detectar distribuciones continuas, se hace necesario un estudio de la técnica FMCW en este contexto, ya que inicialmente esta concebida para la detección de blancos puntuales. Dentro de este marco, en este PFC se pretenden alcanzar los siguientes objetivos:

- **Estudio teórico de la técnica FMCW en lidar**, analizando la solidez matemática de la técnica así como el papel que juegan las diferentes variables involucradas.
- **Elaboración de un modelo de simulación de la técnica FMCW que se ajuste a la teoría**, analizando las diferencias entre este modelo y el estudio teórico, y estudiando mediante simulaciones casos complejos.
- **Montaje de laboratorio del sistema FMCW a frecuencias ópticas**, analizando las diferencias entre este montaje y el modelo de simulación o el estudio teórico, identificando sus posibles limitaciones.
- **Estudio de la interacción de dos o más blancos a distancias comparables a la distancia de resolución**. Este es el objetivo principal del proyecto.
- **Estudio de la respuesta del sistema a un blanco disperso**.

## CAPÍTULO 2: ANÁLISIS TEÓRICO

En este capítulo se analiza un sistema Lidar FMCW desde el punto de vista teórico, explicando su principio de funcionamiento y viendo las posibles limitaciones que pueda tener. Se empieza este análisis haciendo un breve repaso de lo que es un Lidar, su principio de funcionamiento y algunos de los tipos que existen. En la segunda parte del capítulo se pasa a analizar el método FMCW en general, particularizando después para el caso de blancos muy juntos, o incluso un blanco continuo.

### 2.1 Lidar

Los sistemas Lidar (Light Detection And Ranging) son sistemas que utilizan una tecnología óptica de detección remota para medir propiedades de luz reflejada (backscattered) por un blanco remoto, con el fin de estimar su distancia, velocidad u otras propiedades del blanco. El sistema lidar más utilizado es el que determina la distancia de un objeto mediante la medición del retardo de pulsos láser. Como en un sistema radar, que usa ondas de radiofrecuencia en lugar de radiación a frecuencias ópticas, la distancia del objeto se determina midiendo el retardo entre la transmisión de un pulso y la detección de la señal reflejada. La diferencia fundamental entre lidar y radar es que en lidar se usan longitudes de onda mucho más pequeñas que en radar, típicamente el ultravioleta, el visible o el infrarrojo cercano. En general, es posible medir objetos que tengan un tamaño comparable o mayor a la longitud de onda de emisión, por lo que un sistema lidar es en general mucho más sensible en la medición de gases y aerosoles que un radar.

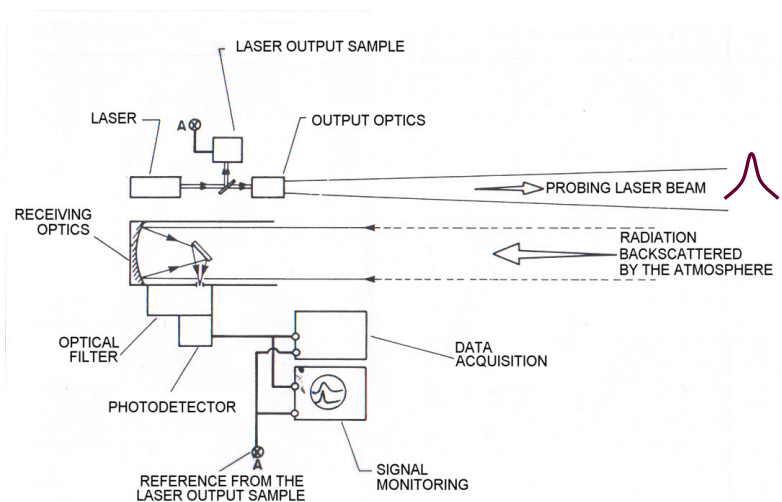


Figura 7: Configuración general de un sistema Lidar [R. M. Measures, 1992]

Son necesarios sistemas de óptica avanzada, componentes electrónicos de alta velocidad y técnicas de procesamiento de señal para obtener buenas medidas en lidar. Podemos ver la configuración general de un sistema Lidar en la Figura 7, dónde observamos las partes esenciales del sistema, cómo el láser, el receptor óptico (acompañado de un telescopio) o los componentes de adquisición y tratamiento de la señal.

### 2.1.1 Principio de funcionamiento: Ecuación Lidar

El principio básico de funcionamiento de un lidar para la detección de gases o aerosoles es el siguiente: la señal luminosa es emitida por el láser, después es reflejada (mediante backscattering), recogida por el telescopio y captada por el detector óptico. Una vez convertida esta señal luminosa en una señal eléctrica, es acondicionada y capturada por el sistema electrónico. Los datos obtenidos son procesados con el fin de obtener la información deseada.

Mediante la ecuación Lidar podemos calcular la potencia recogida por el sistema Lidar, como función de la distancia del blanco que ha reflejado la luz. Emitimos una señal cuya potencia depende del tiempo:

$$P = P(t) \quad (2.1)$$

Esta señal se propaga por el espacio hasta que es reflejada o absorbida por la atmósfera. Parte de la señal reflejada vuelve al sistema Lidar, y es captada por el receptor, tras haber sido concentrada por el telescopio. La potencia detectada tiene la siguiente expresión [R. M. Measures, 1992]:

$$P_r(R) = \int_0^{R_{\max}} P\left(t - 2\frac{R}{c}\right) \frac{A_r}{R^2} \beta(R) e^{-2\int_0^R \alpha(x) dx} dR \quad (2.2)$$

Observamos en la ecuación (2.2) la potencia emitida por el láser  $P(t)$ , el área del telescopio  $A_r$ , la atenuación sufrida por la señal al viajar por espacio abierto, y el coeficiente de backscattering  $\beta(R)$  del medio que refleja la señal. El término  $A_r/R^2$  representa el ángulo sólido en recepción que determina la cantidad de radiación captada por el telescopio.

Para el caso del Lidar pulsado, con un pulso rectangular ideal, la ecuación (2.2) adopta la siguiente forma:

$$P_r(R) = \int_0^{R_{\max}} P_0 \Pi\left(\frac{t - 2R/c}{c\tau/2}\right) \frac{A_r}{R^2} \beta(R) e^{-2\int_0^R \alpha(x) dx} dR \quad (2.3)$$

Asumiendo que el integrando es constante en el intervalo  $c\tau/2$ , podemos llegar a una ecuación en la que  $P_0$  es la potencia de pico, y la longitud del pulso es  $c\tau/2$ :

$$P_r(R) = \frac{P_0 A_r c\tau}{2R^2} \beta(R) e^{-2\int_0^R \alpha(x) dx} \quad (2.4)$$

Para el caso de Lidar de onda continua, la resolución en distancia se obtiene mediante técnicas diferentes, como códigos pseudo-aleatorios (compresión de pulsos) o modulación en frecuencia (FMCW). Describiremos esto en la sección siguiente.

### 2.1.2 Ejemplos de Lidar

Los sistemas Lidar se pueden usar para medir gran variedad de magnitudes físicas. Para cada una de ellas existen muchos tipos diferentes de sistemas Lidar que se

pueden clasificar de diferentes maneras. En este apartado se darán algunos ejemplos de los sistemas Lidar que se suelen utilizar con más frecuencia para la medición de gases:

### 2.1.2.i Lidar Rayleigh

El sistema Lidar Rayleigh utiliza un láser de alta potencia no sintonizable (láser pulsado Nd:YAG) para medir cambios en la densidad o en el coeficiente de reflectividad de la atmósfera debido a partículas, como aerosoles, nubes o polvo. Estas partículas producen reflexiones relativamente altas que son fáciles de detectar. Este sistema es útil para detectar fenómenos meteorológicos como lluvia, nieve, granizo y nubes densas. También se puede utilizar para medir la presión y temperatura en altas capas de la atmósfera, ayudando a la reentrada de vehículos espaciales. En los Juegos Olímpicos de Barcelona en 1992, un sistema basado en el Lidar Rayleigh fue utilizado para medir la velocidad del viento en la ciudad, y así predecir el impacto de la contaminación en la zona del centro.

### 2.1.2.ii DIAL (Differential Absorption Lidar)

El sistema DIAL es un tipo de lidar pulsado, utilizado comúnmente para la detección de gases, especialmente vapor de agua, con alta resolución espacial y temporal. De esta manera, procesos dinámicos atmosféricos, como la capa límite de desarrollo o el transporte turbulento, pueden ser estudiados en detalle.

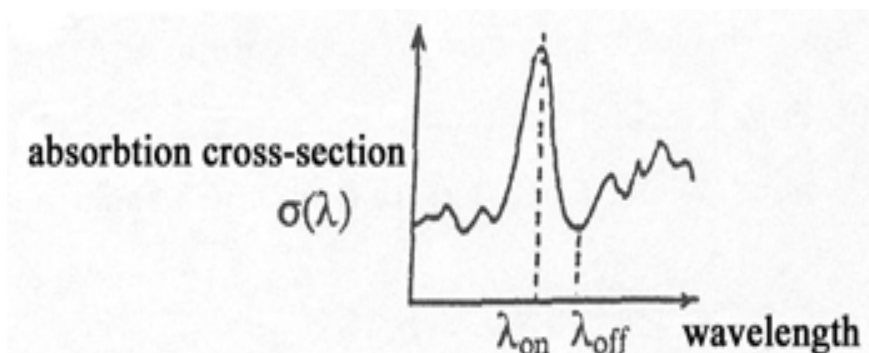


Figura 8: Longitudes de onda de emisión en DIAL (<http://lidar.tropos.de/en/research/dial.html>)

Dos pulsos láser con longitudes de onda  $\lambda_{on}$  y  $\lambda_{off}$  (Figura 8) se emiten casi simultáneamente. La luz de la longitud de onda  $\lambda_{on}$  es más absorbida por el gas de estudio que  $\lambda_{off}$ . Asumiendo que los demás parámetros de backscattering y extinción en la atmósfera son iguales para las dos longitudes de onda, se puede determinar el coeficiente de absorción del gas para cada una a partir de la potencia recibida  $P_{on}$  y  $P_{off}$ . Si el coeficiente de absorción del gas de estudio es conocido para esas dos longitudes de onda, la concentración de los átomos o moléculas del gas pueden deducirse directamente.

### 2.1.2.iii Lidar Raman

Un sistema Lidar Raman detecta, además de la luz reflejada por el gas a la misma longitud de onda de emisión (backscattering elástico), señales a diferentes longitudes de onda. Estas señales proceden de fotones producidos por moléculas (o átomos) que absorben o aportan energía a los fotones incidentes (scattering inelástico). Debido a este efecto, estas moléculas cambian su estado vibracional o rotacional (proceso Raman). La luz emitida por estas moléculas experimenta un cambio en su frecuencia debido a su cambio de energía. Este cambio en frecuencia es característico de la molécula que lo originó, y por lo tanto de cada gas.

En la Figura 9 está representada una simulación del espectro Raman para varios gases. Se observa que para cada gas el espectro Raman consiste en una longitud de onda central de reflexión (que es la longitud de onda emitida), rodeada de otras longitudes de onda cercanas que también reflejan potencia.

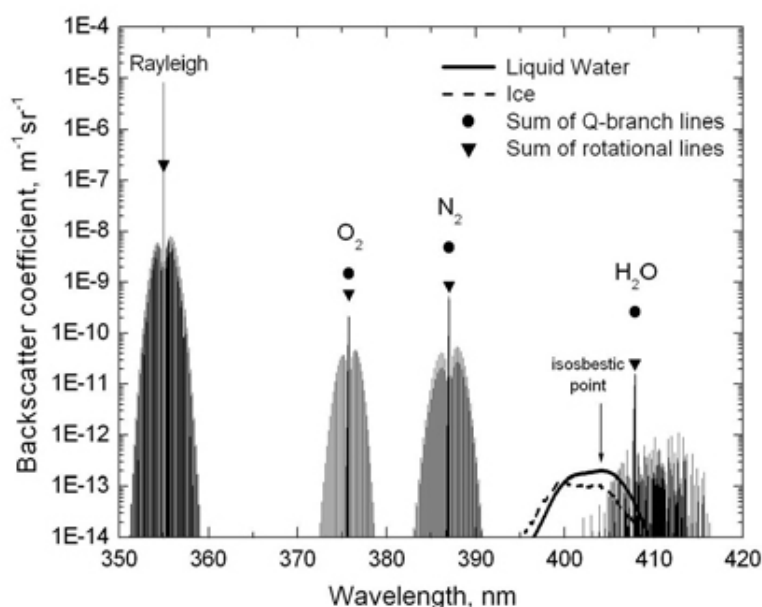


Figura 9: Simulación del espectro Raman a 355nm (<http://lidar.tropos.de/en/research/raman.html>)

La potencia recibida por el efecto del Raman scattering es varios órdenes de magnitud más pequeña que la recibida por backscattering elástico, por lo que normalmente un Lidar Raman está construido con un láser pulsado de alta potencia, telescopios relativamente grandes y detectores muy eficientes.

### 2.1.2.iv Lidar Doppler

El efecto Doppler se ha utilizado en lidar para medir la velocidad del viento, y para diferenciar entre reflexiones de moléculas o aerosoles en la atmósfera. El cambio en longitud de onda producido por este efecto es muy pequeño (una fracción de cambio en la frecuencia de  $10^{-8}$  para velocidades de  $1\text{m/s}$  a una longitud de onda de  $10\mu\text{m}$ ), por lo que son medidos mediante detección óptica heterodina. La detección óptica heterodina implica el mezclado óptico de la señal recibida con otro láser operando a la



longitud de onda de emisión, trasladando así la diferencia frecuencial a una banda más baja, donde es posible su medición.

Este sistema ha sido utilizado comúnmente en aeropuertos para, además de medir la velocidad del aire y predecir cambios en la meteorología, estudiar las turbulencias producidas por los aviones.

## 2.2 FMCW

Las técnicas más utilizadas en lidar están basadas en sistemas pulsados. A pesar de ello, los métodos por onda continua han ido ganando peso gracias a sus características. FMCW (Frequency Modulated Continuous Wave) es una técnica comúnmente utilizada en sistemas Radar y Lidar. Se trata de una técnica de onda continua en la que la frecuencia es modulada, enviándose un tono que aumenta en frecuencia de manera conocida. Gracias a esta modulación se puede medir el tiempo que ha tardado la señal en ir y volver de cierto blanco, comparando las frecuencias emitida y recibida, y por lo tanto estimar a qué distancia se encuentra.

### 2.2.1 Principio de funcionamiento

La técnica FMCW consiste en el envío de un tono cuya frecuencia crece a lo largo del tiempo de forma lineal, tal y como está representado en la Figura 10.

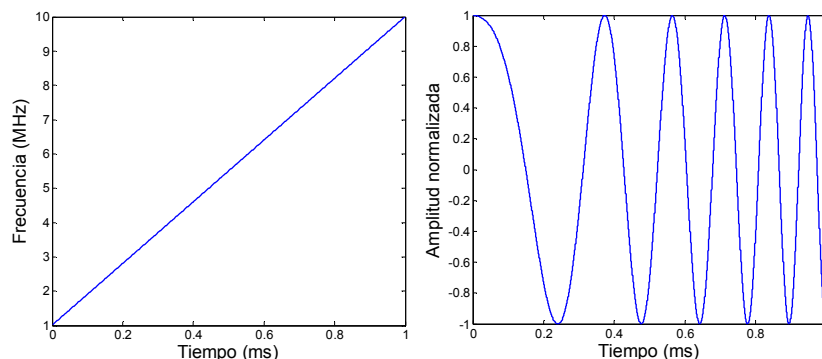


Figura 10: Señal emitida en FMCW

El crecimiento de esta frecuencia es constante en el tiempo, por lo tanto la pendiente de la recta se mantiene constante. Una vez enviada, esta señal rebota en un blanco y vuelve al receptor. La señal que recibimos es la misma que la enviada, pero con un desplazamiento temporal debido al tiempo que ha necesitado la señal emitida para propagarse hasta el blanco y volver. Ya que la pendiente de la recta frecuencial se mantiene constante, al comparar la frecuencia instantánea de la señal emitida con la recibida, observamos que la diferencia de frecuencias es constante, tal y como se observa en la Figura 11.

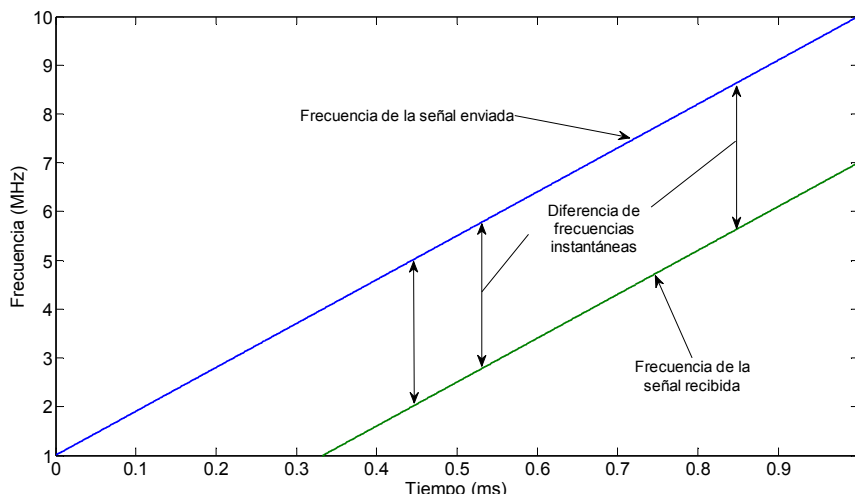


Figura 11: Diferencia de frecuencias instantáneas en FMCW

Esta diferencia de frecuencias se mantiene en todo momento, y es proporcional al retardo de la señal recibida, y por lo tanto proporcional a la distancia a la que se encuentra el blanco en el que se ha reflejado la onda. Resumiendo, la distancia a la que se encuentra el blanco de estudio es proporcional a la resta de las frecuencias recibida y emitida instantáneamente.

Aplicando las señales emitida y recibida a la entrada de un mezclador obtenemos las componentes suma y resta frecuencial. Ya que la información de la distancia a la que se encuentra el blanco está en la frecuencia resta, aplicaremos la señal de salida del mezclador a un filtro paso-bajo, eliminando así la componente suma. La señal a la salida del filtro se puede analizar mediante técnicas *software* o *hardware*:

- La técnica *hardware* consiste en aplicar esta señal a un banco de filtros muy estrechos sintonizados a diferentes frecuencias. El filtro que esté sintonizado a la frecuencia más próxima a la frecuencia de respuesta del blanco de estudio será el que obtendrá mayor potencia de salida, y por lo tanto se puede decidir que existe un blanco asociado a esa frecuencia.
- La técnica *software* consiste en la adquisición de esta señal y su posterior análisis mediante la Transformada de Fourier. El espectro de la señal adquirida presentará mayor potencia a la frecuencia asociada al blanco de estudio.

Independientemente del método de análisis, obtendremos la frecuencia asociada al blanco de estudio. Aplicando una constante de proporcionalidad a esta frecuencia, se obtiene la distancia al origen del objeto.

Este método no está limitado a la detección de un solo blanco, sino que puede detectar múltiples blancos a la vez, por lo que analizando el espectro de la señal resultante del mezclado, y reescalando el eje frecuencial para que refleje la distancia al origen, se puede obtener una distribución de todos los blancos a los que apunte nuestro sistema.

## 2.2.2 Análisis teórico

El principio de funcionamiento descrito en el apartado anterior se puede expresar mediante funciones matemáticas. Partiendo de la expresión de la señal enviada, se pretende demostrar que el sistema genera una frecuencia proporcional a la distancia al origen de cada blanco estudiado. Además, en este apartado se definirán los parámetros que describen el comportamiento del sistema, los cuales determinan la salida y otros parámetros de calidad del sistema. Se empieza este análisis viendo un sistema Radar FMCW, para pasar después a un sistema Lidar FMCW, viendo sus similitudes y diferencias, y justificando el uso de Lidar para este proyecto en concreto.

### 2.2.2.i FMCW en radar

Inicialmente, el método FMCW fue diseñado para utilizar en Radar. En un sistema FMCW en Radar la señal modula directamente la amplitud de la onda electromagnética que se emite. Partimos pues de la expresión de esta señal emitida  $s(t)$ , cuya frecuencia instantánea  $f_i(t)$  aumenta linealmente con el tiempo:

$$s(t) = S_0 \cos \left( 2\pi \int_0^t f_i(\tau) d\tau + \varphi_0 \right) = S_0 \cos \left( 2\pi \left( f_0 t + \frac{K}{2} t^2 \right) + \varphi_0 \right) \quad (2.5)$$

$$f_i(t) = f_0 + Kt \quad (2.6)$$

Donde  $S_0$  es la amplitud de la señal emitida,  $f_0$  es la frecuencia inicial,  $\varphi_0$  es la fase inicial y  $K$  es la pendiente de la recta frecuencial (o también el índice de modulación de la señal FM). Después de ser reflejada por un blanco, se recibe la señal:

$$s_R(t) = \alpha S_0 \cos \left( 2\pi \left( f_0 (t - t_0) + \frac{K}{2} (t - t_0)^2 \right) + \varphi_0 + \varphi_1 \right) \quad (2.7)$$

Donde  $\alpha$  es el coeficiente de reflectividad de este blanco,  $t_0$  es el tiempo de ida y vuelta de la señal y  $\varphi_1$  es la fase de respuesta del blanco. Mezclamos la señal recibida con la enviada:

$$\begin{aligned} s_{RM}(t) &= s(t) \cdot s_R(t) = \\ &= S_0 \cos \left( 2\pi \left( f_0 t + \frac{K}{2} t^2 \right) + \varphi_0 \right) \alpha S_0 \cos \left( 2\pi \left( f_0 (t - t_0) + \frac{K}{2} (t - t_0)^2 \right) + \varphi_0 + \varphi_1 \right) \end{aligned}$$

Aplicando la propiedad del producto de cosenos en suma y resta de fases:

$$\begin{aligned} s_{RM}(t) &= \frac{\alpha S_0^2}{2} \left[ \cos \left( 2\pi \left( f_0 (t - t_0) - f_0 t + \frac{K}{2} (t - t_0)^2 - \frac{K}{2} t^2 \right) + \varphi_0 + \varphi_1 - \varphi_0 \right) + \right. \\ &\quad \left. + \cos \left( 2\pi \left( f_0 (t - t_0) + f_0 t + \frac{K}{2} (t - t_0)^2 + \frac{K}{2} t^2 \right) + \varphi_0 + \varphi_1 + \varphi_0 \right) \right] \end{aligned}$$

Desarrollando los cuadrados, y agrupando términos:

$$s_{RM}(t) = \frac{\alpha S_0^2}{2} \left[ \cos \left( 2\pi \left( f_0 t_0 + \frac{K}{2} (-2t t_0 + t_0^2) \right) + \varphi_1 \right) + \right. \\ \left. + \cos \left( 2\pi \left( f_0 (2t - t_0) + \frac{K}{2} (2t^2 - 2t t_0 + t_0^2) \right) + 2\varphi_0 + \varphi_1 \right) \right] \quad (2.8)$$

Se observa que el segundo término es a frecuencia doble, con un espectro con potencia repartida entre  $2f_0$  y la frecuencia máxima de nuestro sistema. El primer término (correspondiente a la resta frecuencial) es de frecuencia constante que depende sólo del retardo  $t_0$ , y es el que se considera el término útil en FMCW. El término a alta frecuencia se puede eliminar mediante filtrado paso-bajo, quedando una señal cuya frecuencia depende de la distancia del blanco que generó el eco:

$$s_{RM}'(t) = \frac{\alpha S_0^2}{2} \cos \left( 2\pi \left( K t_0 t - \frac{K}{2} t_0^2 - f_0 t_0 \right) - \varphi_1 \right) \quad (2.9)$$

El coseno de la expresión (2.9) tiene señal a frecuencia constante, ya que a parte del término  $K t_0 t$  los demás son constantes en el tiempo. Esto da lugar a que la señal  $s_{RM}'(t)$  sólo tenga una componente frecuencial a frecuencia proporcional a la distancia del objeto:

$$f = K t_0 = \frac{2K}{c} d_0 \quad (2.10)$$

Donde  $d_0$  es la distancia del blanco. Se puede observar además que la fase de la señal  $s_{RM}(t)$  depende de la fase de respuesta del blanco  $\varphi_1$ , pero no de la fase inicial  $\varphi_0$ . La Transformada de Fourier de esta señal da como resultado una distribución en la que los blancos están representados por deltas frecuenciales, cuyas frecuencias están asociadas a la distancia de cada objeto.

### 2.2.2.ii FMCW en Lidar

Aunque a grandes rasgos un sistema Lidar FMCW se comporta igual que un sistema Radar FMCW, existen pequeñas diferencias en su implementación que dan como resultado ciertas propiedades en las fases del sistema Lidar FMCW que lo hacen más adecuado que el otro para situaciones concretas.

En un sistema lidar, la corriente de entrada al láser modula (a la vez) la longitud de onda y la potencia de emisión de la señal luminosa. Para el caso concreto de FMCW en lidar, sólo se tiene en cuenta la modulación de la potencia del láser. De esta manera se obtiene que la información FMCW viaje en la envolvente de la señal luminosa, y no en su amplitud como en el caso de FMCW en radar.

Partimos de la expresión de la tensión de entrada del láser, homóloga a la del caso de FMCW en radar (2.5), con la misma frecuencia instantánea de la ecuación (2.6):

$$v(t) = v_0 + \Delta v \cos \left( 2\pi \left( f_0 t + \frac{K}{2} t^2 \right) + \varphi_0 \right) \quad (2.11)$$

Donde  $v_0$  es la tensión media,  $\Delta v$  es la amplitud de la señal FMCW,  $f_0$  es la frecuencia inicial,  $K$ , como en el caso del radar, es la velocidad con la que cambia la frecuencia del coseno, y  $\varphi_0$  es su fase inicial. Esta tensión se transforma en corriente mediante un *driver* que hace de fuente de corriente, que puede modelarse mediante la resistencia de entrada del emisor láser.

$$i(t) = \frac{v(t)}{R_L}$$

Esta corriente es la que finalmente modula el diodo láser, dando lugar a una señal luminosa de campo eléctrico  $u(t)$  cuya potencia  $P(t)$  es aproximadamente proporcional a la corriente  $i(t)$ :

$$P(t) \propto i(t) - i_{th} \quad (2.12)$$

$$u(t) = \sqrt{P(t)} \cos\left(2\pi\nu_{LASO}t + 2\pi\delta_v \int_0^t i(t')dt'\right) \quad (2.13)$$

Donde  $\nu_{LASO}$  es la frecuencia central de emisión del láser y  $\delta_v$  es el factor de proporcionalidad entre corriente y frecuencia.

Después de que la señal rebote sobre un blanco obtenemos (para el caso de más de un blanco tendríamos la suma de cada uno):

$$u_r(t) = \sqrt{\sigma_t} \sqrt{P(t-t_0)} \cos\left(2\pi\nu_{LASO}(t-t_0) + 2\pi\delta_v \int_0^{t-t_0} i(t')dt' + \varphi_1\right) \quad (2.14)$$

Donde  $t_0$  es el retardo sufrido por la señal asociado al doble de la distancia a la que se encuentra el blanco, y  $\varphi_1$  es la fase de respuesta del blanco.

Después de reflejarse en los blancos, la señal luminosa es captada por un receptor compuesto normalmente por un fotodiodo que hace una detección cuadrática del campo, y por lo tanto detecta la densidad de potencia óptica  $I_r(t)$ :

$$\begin{aligned} I_r(t) &= |u_r(t)|^2 = \\ &= \sigma_t P(t-t_0) \frac{1}{2} \left(1 + \cos\left(2\left(2\pi\nu_{LASO}(t-t_0) + 2\pi\delta_v \int_0^{t-t_0} i(t')dt' + \varphi_1\right)\right)\right) \end{aligned} \quad (2.15)$$

El receptor genera a su salida una corriente proporcional a esta intensidad óptica (o densidad de potencia óptica) de la señal recibida. En nuestro caso se genera una corriente  $i_r(t)$  proporcional a la señal  $I_r(t)$ , y por lo tanto proporcional a la potencia de el campo eléctrico  $u_r(t)$ :

$$i_r(t) \propto P(t-t_0) \quad (2.16)$$

El mismo receptor, a través de una transimpedancia, pasa esta corriente recibida a tensión. Esta será la tensión de salida del receptor:

$$v_r(t) = v_0' + \Delta v' \cos\left(2\pi\left(f_0(t-t_0) + \frac{K}{2}(t-t_0)^2\right) + \varphi_0\right) \quad (2.17)$$

Es importante apreciar que la fase de esta señal útil recibida es la misma que la fase de la tensión enviada (a diferencia del caso FMCW para radares, donde esta fase depende, además de la fase inicial, de la fase de respuesta del blanco). Este efecto es una consecuencia de que la información FMCW en lidar viaje sobre la envolvente de la señal, y por lo tanto no se ve afectada por la fase de respuesta del blanco, que modifica la fase de la portadora óptica.

Igual que en caso de FMCW para radares, para comparar la frecuencia de la señal detectada con la que se está emitiendo en el instante actual, se multiplica la señal emitida con la recibida. Dado que normalmente los mezcladores filtran la continua antes de multiplicar, no la tendremos en cuenta en el producto de las tensiones emitida y recibida:

$$\begin{aligned} v_m(t) &= (v(t) - v_0)(v_r(t) - v_0') = \\ &= \frac{\Delta v \Delta v'}{2} \left[ \cos \left( 2\pi \left( f_0 t_0 + \frac{K}{2} (-2t t_0 + t_0^2) \right) \right) + \right. \\ &\quad \left. + \cos \left( 2\pi \left( f_0 (2t - t_0) + \frac{K}{2} (2t^2 - 2t t_0 + t_0^2) \right) + 2\varphi_0 \right) \right] \end{aligned} \quad (2.18)$$

Como en el caso de FMCW en radar, tras el mezclado se obtienen los términos suma y resta frecuencial. El término buscado es de nuevo la componente resta del batido frecuencial, ya que es el término a frecuencia constante que contiene la diferencia de frecuencias recibida y emitida. Filtrando paso-bajo la componente suma y agrupando términos obtenemos:

$$v_{\text{útil}}(t) = \frac{\Delta v \Delta v'}{2} \cos(2\pi f_R t + \varphi_{\text{final}}) \quad (2.19)$$

Donde:

$$f_R = K t_0 \quad (2.20)$$

$$\varphi_{\text{final}} = 2\pi \left( f_0 t_0 + \frac{K}{2} t_0^2 \right) \quad (2.21)$$

Se observa que la fase final no depende de la fase inicial de la ecuación (2.11)  $\varphi_0$  ni de la fase de respuesta del blanco  $\varphi_1$ , por lo que cada realización nos dará la misma señal de salida (igual fase y frecuencia). Esta no dependencia de las fases iniciales será de gran utilidad para poder reducir el ruido a base de sumar diferentes periodos de la señal útil obtenida.

Analizando la frecuencia  $f_R$  del término útil (2.19) podemos estimar la distancia a la que se encuentra el blanco, ya que esta frecuencia es proporcional al retardo sufrido por la señal:

$$d_0 = \frac{t_0 c}{2} = \frac{c}{2K} f_R \quad (2.22)$$

### 2.2.3 Diagrama de bloques

Este es el diagrama de bloques utilizado en el método FMCW en Lidar:

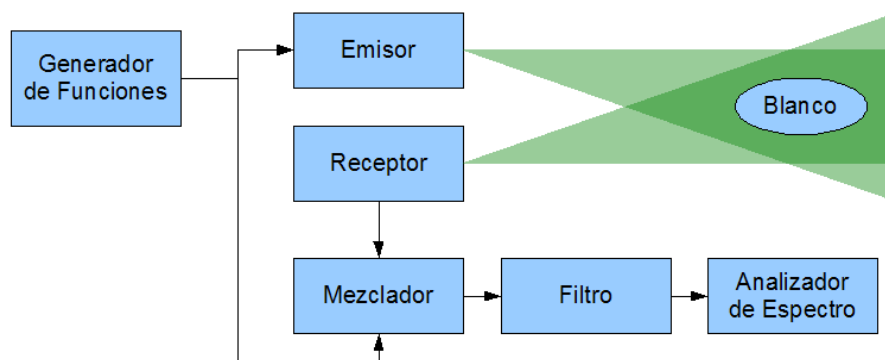


Figura 12: Diagrama de bloques de un sistema FMCW

En la Figura 12 podemos ver los principales bloques de un sistema FMCW:

- **Generador de Funciones:** Es el bloque encargado de la generación de la sinusoide que aumenta en frecuencia linealmente con el tiempo (señal *sweep*).
- **Emisor:** Es el bloque encargado de emitir la señal Lidar. Se trata de un diodo láser excitado por la corriente de entrada, que emite la señal luminosa. Dado que esta corriente cae a través de la resistencia de entrada del emisor, la potencia de la señal luminosa es proporcional a la tensión de entrada de este bloque.
- **Receptor:** Se encarga de captar del medio la señal luminosa que ha rebotado en los blancos, y transformarla en una señal eléctrica. Se trata de un fotodetector que genera una corriente proporcional a la envolvente de la señal luminosa. Esta corriente se transforma después en tensión a través de una transimpedancia.
- **Mezclador:** Realiza el batido frecuencial de las señales emitida y recibida, dando como resultado las componentes frecuenciales suma y resta.
- **Filtro:** Es un filtro paso-bajo que elimina la componente suma del producto de señales.
- **Analizador de Espectros:** Es el último bloque del montaje. Sirve para analizar el espectro de la señal a la salida del filtro, cuyas componentes frecuenciales son proporcionales a los blancos detectados por el sistema.

### 2.2.4 Montaje real: "diente de sierra" frecuencial y resolución

No se puede hacer crecer la frecuencia de la señal FMCW indefinidamente en ningún sistema real. Para hacer que nuestro sistema sea implementable, hacemos que la frecuencia aumente linealmente, hasta alcanzar una desviación máxima  $\Delta f_{dev}$  respecto de la frecuencia inicial  $f_0$ , en un periodo de tiempo  $T_m$ . Después volvemos a la frecuencia inicial  $f_0$ , dando lugar a un "diente de sierra" frecuencial.

El parámetro  $K$  se puede expresar en función de las características de la señal "diente de sierra" frecuencial como:

$$K = \frac{\Delta f_{dev}}{T_m} \tag{2.23}$$

Donde  $T_m$  es el periodo de la modulación (periodo de repetición) y  $\Delta f_{dev}$  es la desviación de frecuencia de la subportadora. Podemos ver estas señales en la Figura 13:

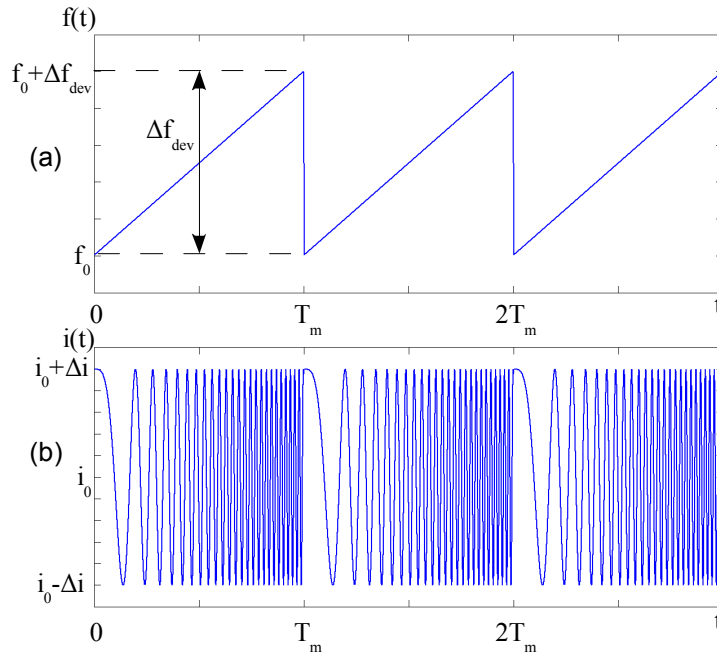


Figura 13: Diagrama temporal de la variación de frecuencia en la señal enviada (a); forma de onda de la señal enviada (b).

Podemos cambiar la ecuación (2.22) para expresar la distancia estimada de la posición de un blanco, en función de los parámetros  $T_m$  y  $\Delta f_{dev}$ :

$$d_0 = \frac{T_m}{\Delta f_{dev}} \frac{c}{2} f_R \tag{2.24}$$

Esta estructura periódica introduce una fuerte limitación en el sistema FMCW. Para cada periodo de la señal, obtendremos una senoide cuya frecuencia depende del blanco detectado, pero limitada a un tiempo  $T_m$ . Esto se puede ver como la multiplicación de la señal para el caso ideal por un pulso de duración  $T_m$ . Después de hacer la Transformada de Fourier, obtendremos la misma respuesta que para el caso ideal (deltas frecuenciales asociadas a los blancos) pero convolucionada por la respuesta frecuencial del pulso rectangular de anchura  $T_m$ :



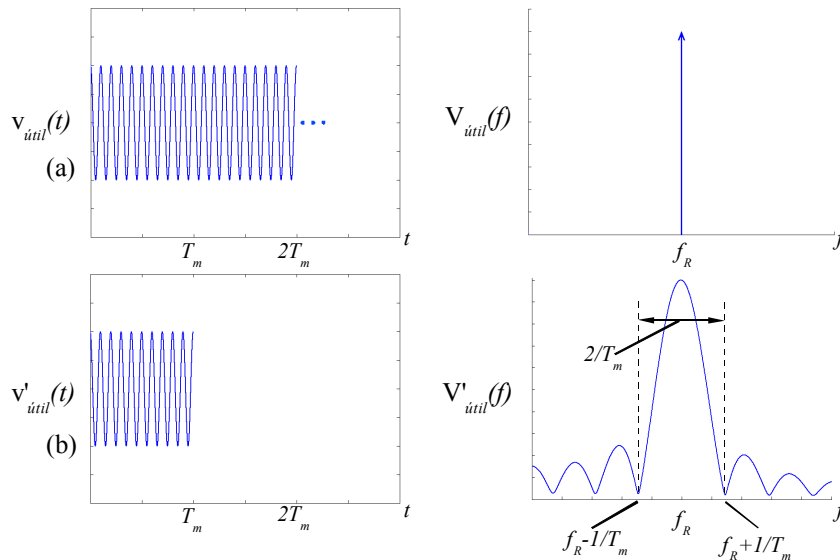


Figura 14: Resultado final ideal en tiempo y frecuencia (a); Resultado final real (caso "diente de sierra frecuencial") en tiempo y frecuencia (b).

El estudio teórico descrito anteriormente sigue teniendo validez para un periodo de la señal, pero teniendo en cuenta que se tiene que convolucionar el resultado final por la transformada de un pulso rectangular de duración  $T_m$  tal y como se observa en la Figura 14.

Para el caso real, los blancos están representados por funciones *sinc* centradas en la frecuencia  $f_R$  asociada a la distancia al origen de ese blanco. Ya que la función *sinc* tiene una anchura mayor que cero, dos o más blancos muy próximos interferirán entre ellos, dando lugar a errores en la medición en su posición, e incluso a la desaparición en la medida de algunos de ellos. Definimos la resolución del sistema  $\Delta d$ , como la distancia mínima a la que pueden estar dos blancos consecutivos, que asociaremos a la distancia entre ceros de las funciones *sinc*. Esta distancia entre ceros es la inversa del periodo  $T_m$ , por lo que usando la expresión (2.24):

$$\Delta d = \frac{T_m}{\Delta f_{dev}} \frac{c}{2} \frac{1}{T_m} = \frac{c}{2\Delta f_{dev}} \tag{2.25}$$

Esto significa que, por ejemplo, para una  $\Delta f_{dev}$  de 30MHz obtenemos una resolución  $\Delta d$  de 5m. Dos blancos que estén separados menos de 5m interferirán apreciablemente entre ellos. El objetivo de este proyecto es precisamente estudiar cómo interfieren estos blancos, tratando de extrapolar los resultados al caso de un blanco continuo.

### 2.2.5 Fase inicial y ruido aditivo

En cada periodo de la señal FMCW se genera una senoide (para cada blanco) de duración  $T_m$  y frecuencia proporcional a la distancia del blanco. Añadiendo la componente de ruido aditivo a la expresión (2.19) obtenemos:

$$v_{util}(t) = \frac{\Delta v \Delta v'}{2} \cos(2\pi f_0 t + \varphi_{final}) + n(t) \tag{2.26}$$

Donde  $\varphi_{final}$  (2.21) es la fase resultante de la señal útil después de todo el proceso, y  $n(t)$  es el ruido aditivo resultante después del multiplicador. Podemos observar que la fase de esta señal no depende de la fase inicial ( $\varphi_0$ ) de la señal *sweep* emitida por el generador de funciones (2.11), ni de la fase de respuesta de los blancos ( $\varphi_1$ ) (2.14). Si suponemos que la frecuencia  $f_0$  es invariante para cada periodo de la señal *sweep*, obtendremos como señal útil un coseno de frecuencia constante y con la misma fase inicial para cada periodo, sumada a un ruido que supondremos de media nula. Teniendo esto en cuenta, si sumamos  $N$  periodos (de tiempo  $T_m$ ) de la señal útil, obtendremos que la potencia del coseno aumentará con  $N^2$  (ya que sumamos cosenos en fase, y por lo tanto sumamos señal, no potencia), mientras que la potencia del ruido sólo aumenta con  $N$ . Idealmente, mediante este proceso se puede mejorar la relación señal-ruido *tanto como queramos* aumentando el número de periodos sumados:

$$SNR_{N\text{ periodos}} = N \cdot SNR_{1\text{ periodo}} \quad (2.27)$$

En la práctica, no es posible aumentar la relación señal-ruido indefinidamente, ya que la deriva introducida por los componentes (sobre todo la del generador de funciones) hace que la frecuencia varíe ligeramente entre un periodo y otro, haciendo que la suma no sea perfectamente en fase, y por lo tanto se suma potencia igual que con el ruido.

## 2.2.6 FMCW en medios continuos

El estudio descrito en los apartados anteriores analiza el sistema FMCW para uno o más blancos discretos. Se hace necesario un estudio más completo para analizar la respuesta del sistema para un blanco continuo. En este apartado se analiza el sistema FMCW para el caso de detección de un blanco continuo y transparente, como en una distribución de aerosoles. Como veremos, para este caso en lugar de sumar las contribuciones de cada blanco, será necesario integrar la potencia reflejada por un blanco continuo, dando como resultado una expresión que podremos asociar a una convolución, y por lo tanto usar sus propiedades para simplificar los cálculos.

### 2.2.6.i Señales ideales

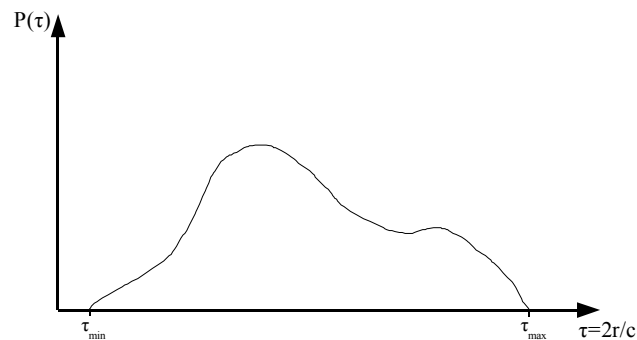


Figura 15: Distribución de potencia reflejada por un blanco continuo

El análisis de la respuesta de un medio continuo en FMCW comienza con la expresión de la señal recibida del medio que está siendo escaneado, representada en la Figura

15. Es importante tener en cuenta que la función  $P(\tau)$  es real no negativa (ya que se trata de la potencia recibida desde cada distancia) y cero fuera del intervalo  $[\tau_{min}, \tau_{max}]$ . Este hecho es esencial para nuestro análisis. La expresión para la potencia recibida es:

$$f(t) = \int_0^{\tau_{max}} P(\tau) \cos\left(2\pi\left(f_0(t-\tau) + \frac{K}{2}(t-\tau)^2\right)\right) d\tau \quad (2.28)$$

Donde  $P(\tau)$  es la distribución de potencia del medio (Figura 15),  $f_0$  es la frecuencia inicial de la señal *sweep* y  $K$  es el índice de modulación. Como  $P(\tau)$  es 0 fuera del intervalo  $[0, \tau_{max}]$ , la señal recibida se puede expresar como la convolución de la señal emitida con la señal enviada, a la que llamaremos  $h(t)$ :

$$h(t) = \cos\left(2\pi\left(f_0 t + \frac{K}{2} t^2\right)\right) \quad (2.29)$$

$$f(t) = P(t) * h(t) \quad (2.30)$$

Se debe tener en cuenta que debido a la incoherencia del sistema, las respuestas de diferentes blancos se suman en potencia, y no en señal. En el caso de un solo blanco, la respuesta es la señal emitida retardada y multiplicada por el coeficiente de reflexión del blanco. En este caso, la función  $P(\tau)$  corresponde a una función delta retardada por el tiempo  $\tau_0$  del blanco ( $P(\tau) = \delta(\tau - \tau_0)$ ).

Tenemos la señal  $f(t)$  que es la expresión de la señal recibida de un medio continuo tras haber enviado la señal  $h(t)$ . Tras la detección, se debe mezclar la señal  $f(t)$  con  $h(t)$ , con lo que obtenemos  $g(t)$ :

$$g(t) = f(t) \cdot h(t) = \int_0^{\tau_{max}} P(\tau) \cos\left(2\pi\left(f_0 t + \frac{K}{2} t^2\right)\right) \cdot \cos\left(2\pi\left(f_0(t-\tau) + \frac{K}{2}(t-\tau)^2\right)\right) d\tau \quad (2.31)$$

$$g(t) = \int_0^{\tau_{max}} P(\tau) \left[ \cos\left(2\pi\left(f_0 \tau + \frac{K}{2}(2t\tau - \tau^2)\right)\right) + \cos\left(2\pi\left(f_0(2t-\tau) + \frac{K}{2}(t^2 + (t-\tau)^2)\right)\right) \right] d\tau \quad (2.32)$$

Aplicando un filtro paso-bajo obtenemos la señal que estamos buscando:

$$g(t) = \int_0^{\tau_{max}} P(\tau) \cos\left(2\pi\left(f_0 \tau + \frac{K}{2}(2t\tau - \tau^2)\right)\right) d\tau \quad (2.33)$$

Podemos observar que para el caso ideal de una señal  $h(t)$  que crece infinitamente en frecuencia, obtenemos una integral de sinusoides de diferentes amplitudes dependiendo de la respuesta del medio bajo estudio. Al hacer la Transformada de Fourier a la expresión (2.33), obtendremos directamente la función de la respuesta del medio, y por lo tanto la información que estamos buscando:

$$G(f) = \frac{1}{2} \int_0^{\tau_{\max}} P(\tau) \left[ \delta\left(f - K\tau\right) e^{+j2\pi\left(f_0\tau - \frac{K}{2}\tau^2\right)} + \delta\left(f + K\tau\right) e^{-j2\pi\left(f_0\tau - \frac{K}{2}\tau^2\right)} \right] d\tau \quad (2.34)$$

$$G(f) = \frac{1}{2} P\left(\frac{|f|}{K}\right) e^{j2\pi \text{sign}(f) \left(f_0 \frac{|f|}{K} - \frac{K}{2} \left(\frac{|f|}{K}\right)^2\right)} \quad (2.35)$$

Observando este resultado final vemos que, excluyendo una cierta fase, la función  $G(f)$  es una versión escalada de la información  $P(\tau)$  (función de concentración) que estábamos buscando, tal y como se observa en la Figura 16.

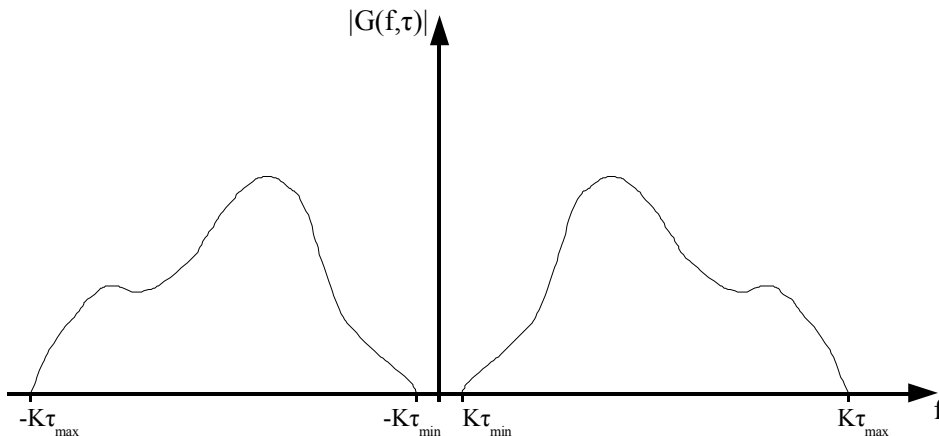


Figura 16: Resultado ideal de la técnica FMCW para medios continuos

### 2.2.6.ii FMCW Real: Diente de sierra frecuencial

Como ya vimos en el apartado 2.2.4, la frecuencia de la señal enviada no puede crecer indefinidamente, por lo que es necesario utilizar otra más realista: el diente de sierra frecuencial. Esta señal se puede ver como el enventanado de la señal  $g(t)$  por un pulso de duración  $T$  (periodo de la señal), y estructurado periódicamente mediante la convolución con un tren de deltas. Partiendo de la expresión (2.33) hallamos la expresión de un periodo de la señal  $g_p(t)$ :

$$g_p(t) = \int_0^{\tau_{\max}} P(\tau) \cos\left(2\pi\left(f_0\tau + \frac{K}{2}(2t\tau - \tau^2)\right)\right) \Pi\left(\frac{t - T/2 - \tau}{T}\right) d\tau \quad (2.36)$$

Para analizar correctamente esta señal, hacemos su Transformada de Fourier:

$$G_p(f) = \int_{-\infty}^{\infty} \int_0^{\tau_{\max}} P(\tau) \cos\left(2\pi\left(f_0\tau + \frac{K}{2}(2t\tau - \tau^2)\right)\right) \Pi\left(\frac{t - T/2 - \tau}{T}\right) d\tau e^{-j2\pi ft} dt \quad (2.37)$$

Cambiando los límites de integración de la integral en  $\tau$  para tener en cuenta el pulso rectangular obtenemos:

$$G_p(f) = \int_0^{\tau_{\max}} P(\tau) \int_{\tau}^{T+\tau} \cos\left(2\pi\left(f_0\tau + \frac{K}{2}(2t\tau - \tau^2)\right)\right) e^{-j2\pi ft} dt d\tau \quad (2.38)$$

Llegados al este punto, calculamos el valor de la integral respecto de la variable  $t$ , con lo que obtenemos el valor de la Transformada de Fourier de  $g_p(t)$ :

$$G_p(f) = \frac{T}{2} \int_0^{\tau_{\max}} P(\tau) \operatorname{sinc}(T(f - K\tau)) e^{+j(\phi_1(\tau) + \phi_2(\tau))} d\tau + PN \quad (2.39)$$

$$\text{donde: } \phi_1(\tau) = 2\pi\left(f_0 + K\frac{T}{2}\right)\tau \quad \text{y} \quad \phi_2(\tau) = 2\pi\left(\frac{K}{2}\tau^2 - f\tau - f\frac{T}{2}\right)$$

En la expresión (2.39), el término  $\phi_1$  corresponde al efecto de la frecuencia central del pulso transmitido respecto al margen de frecuencias del pulso completo; el término  $\phi_2$  corresponde a componentes de segundo orden y fases comunes, y  $PN$  corresponde a la parte negativa de la función. A partir de ahora, se obviará el término  $\phi_2$ , ya que se puede demostrar que su influencia en el resultado final no es importante. Finalmente tenemos:

$$G_p(f) = \frac{T}{2} \int_0^{\tau_{\max}} P(\tau) \operatorname{sinc}(T(f - K\tau)) e^{+j2\pi\left(f_0 + K\frac{T}{2}\right)\tau} d\tau + PN \quad (2.40)$$

Si comparamos este resultado con el ideal con un coseno infinito (2.34), observamos que la función delta se ha convertido en una función *sinc*. Este hecho cambia significativamente el resultado de la detección. De hecho, debido al efecto de la función *sinc* y el término exponencial que multiplica, el resultado se puede ver como una especie de filtrado paso-banda de la función  $P(\tau)$ , y por lo tanto esta versión filtrada de  $P(\tau)$  será lo que recuperemos en nuestro sistema real.

### 2.2.7 Caso particular: Dos blancos

Para entender mejor como afectan las fases de las ecuaciones (2.39) y (2.40) a la suma de blancos, se particulariza para el caso en que es la suma de dos blancos puntuales a distancias  $\tau_1$  y  $\tau_2$ . Para blancos muy separados, la interferencia que puede ejercer el uno sobre el otro es muy pequeña, ya que la amplitud de la *sinc* correspondiente a un blanco en la posición del otro será varios órdenes de magnitud mayor que el lóbulo interferente del otro blanco. Para valores elevados de separación entre blancos, las fases relativas entre blancos no son determinantes. En este apartado se estudian estas fases para separaciones del orden de la distancia entre ceros de las funciones *sinc*.

Para simplificar, se toman los dos blancos de potencia unitaria:

$$P(\tau) = \delta(\tau - \tau_1) + \delta(\tau - \tau_2) \quad (2.41)$$

Sustituyendo esta distribución de potencia en la ecuación (2.40), y haciendo la integral en  $\tau$  obtenemos:

$$G_p(f) = \frac{T}{2} \left[ \sin c(T(f - K\tau_1)) e^{+j(\phi_1(\tau_1) + \phi_2(\tau_1))} + \sin c(T(f - K\tau_2)) e^{+j(\phi_1(\tau_2) + \phi_2(\tau_2))} \right] + PN$$

Donde se ha mantenido las contribuciones  $\phi_1(\tau)$  y  $\phi_2(\tau)$  en la fase. La diferencia de fases entre un blanco y el otro determina la manera en que se suman las funciones *sinc*, causando interferencias constructivas cuando la fase relativa está próxima a cero, y interferencias destructivas cuando está próxima a  $\pi$ .

$$\Delta\phi = \phi_1(\tau_2) + \phi_2(\tau_2) - (\phi_1(\tau_1) + \phi_2(\tau_1)) = \Delta\phi_1 + \Delta\phi_2 \quad (2.42)$$

Las diferencia de fase debida a  $\phi_2$ , como se ha comentado en el apartado anterior, es despreciable. Veamos esta diferencia de fase con más detenimiento, para el caso particular de dos blancos:

$$\Delta\phi_2 = \phi_2(\tau_2) - \phi_2(\tau_1) = 2\pi \left( \frac{K}{2} (\tau_2^2 - \tau_1^2) - f(\tau_2 - \tau_1) \right) \quad (2.43)$$

Para el caso de separaciones del orden de la distancia entre ceros de las *sincs*, y para frecuencias del orden de la posición del blanco más lejano se obtiene:

$$\tau_2 - \tau_1 \approx \frac{1}{KT}; f \approx K\tau_2 \text{ para } (\tau_2 > \tau_1)$$

$$\Delta\phi_2 \approx 2\pi \left( \frac{K}{2} \left( \frac{1}{KT} \right) (\tau_2 + \tau_1) - K\tau_2 \frac{1}{KT} \right) = 2\pi \left( \frac{\tau_2}{2T} + \frac{\tau_1}{2T} - \frac{\tau_2}{T} \right) \rightarrow 0 \quad (2.44)$$

La duración del periodo  $T$  es por definición mucho mayor que los retardos asociados a cada uno de los blancos, por lo tanto obtenemos una fase  $\Delta\phi_2$  despreciable.

Sin embargo, la fase relativa  $\Delta\phi_1$  sí toma valores apreciables según la separación entre los dos blancos:

$$\Delta\phi_1 = 2\pi \left( f_0 + K \frac{T}{2} \right) (\tau_2 - \tau_1) \quad (2.45)$$

Por ejemplo, para el caso límite de  $f_0 = 0$  (el cual quedaría fuera del método FMCW, ya que no se podría separar la componente suma de la componente resta del mezclado), esta fase relativa toma valores de  $\Delta\phi_1 = \pi$  para  $\tau_2 - \tau_1 = 1/KT$  (blancos separados una distancia igual a la distancia entre ceros de la *sinc*) ó  $\Delta\phi_1 = 2\pi$  para  $\tau_2 - \tau_1 = 2/KT$  (blancos separados una distancia igual a la anchura del lóbulo principal de la *sinc*). Se observa pues que los blancos se infieren constructiva o destructivamente dependiendo de la separación entre ambos, pudiendo tomar todos los posibles valores de fase relativa a las distancias de estudio. Mientras mayor sea el valor de  $f_0$ , antes se darán estos cambios de fase, haciendo que pequeñas diferencias en las separaciones de dos blancos den lugar a sumas con fases totalmente diferentes.

En el capítulo de simulaciones se ha profundizado más en esta interferencia inter-blancos para dos y tres blancos, estimando el error que provocan en las medidas.





## CAPÍTULO 3: SIMULACIÓN

Una vez estudiado el método FMCW teóricamente, se ha implementado un modelo en un entorno de simulación. Este modelo simulado tiene dos objetivos: por un lado pretende estudiar casos complejos (como por ejemplo el estudio de la respuesta del sistema a dos o más blancos a distancias del orden de la resolución) en los que el estudio teórico se hace complicado. Además, como segundo objetivo, se pretende ilustrar el comportamiento general del sistema mediante una serie de gráficas que nos ayudarán a interpretar posteriormente los resultados obtenidos en el laboratorio.

Se ha elegido el entorno MATLAB para hacer estas simulaciones, ya que es un sistema orientado al manejo de vectores de datos (muestras), y por lo tanto de fácil programación. Durante las simulaciones toda señal será tratada como un vector de muestras dentro del entorno en MATLAB, por lo que se podrán aplicar herramientas propias de este entorno (como la FFT).

En este capítulo se explica una manera de implementar el método FMCW, partiendo de parámetros de alto nivel, como el periodo de la señal *sweep*  $T_m$ , las frecuencias inicial y final, o la posición de los blancos a detectar. Además, también se han desarrollado simulaciones de la interacción de dos o tres blancos a diferentes separaciones, generando gráficas que se puedan comparar con los resultados obtenidos en las medidas de laboratorio.

### 3.1 Generación de un periodo de la señal *sweep*

La base del método FMCW es la señal senoidal que aumenta de frecuencia linealmente con el tiempo (llamada señal *sweep*). En este apartado se explica una manera de generar un vector que contenga las muestras de un periodo de esta señal de manera que cumpla los requisitos globales de la Tabla 1:

Magnitud	Valor
Frecuencia inicial	10 MHz
Frecuencia final	40 MHz
Periodo	1ms

Tabla 1: Valores de inicialización de la señal *sweep* simulada

Aunque la duración en tiempo de un periodo esté fijada, en función del número de muestras ( $N$ ) que tenga el vector generado se conseguirá una mayor o menor resolución temporal. Esta resolución se define como el tiempo que transcurre entre dos muestras consecutivas, y por lo tanto el tiempo mínimo que podemos retrasar la señal *sweep* (para retrasarla un número entero de muestras). La resolución temporal limita el sistema de simulación, fijando el retardo mínimo que puede sufrir la señal en ir y venir de los blancos, y por lo tanto fijando la mínima separación que pueden tener dos blancos consecutivos simultáneos.

Se puede ver también la resolución temporal como la inversa de la frecuencia de muestreo del sistema, que debe cumplir el criterio de Nyquist, y por lo tanto debe ser mayor que el doble de la frecuencia máxima de la señal *sweep*, y así evitar aliasing.

Utilizando los valores de la Tabla 1, y generando un vector de  $2^{20}$  muestras, obtenemos una resolución espacial de  $14,3\text{cm}$ , suficiente para la mayoría de las simulaciones propuestas. Además la frecuencia de muestreo con esos datos es superior a  $1\text{GHz}$ , por lo que estamos muy por encima de la frecuencia de Nyquist ( $80\text{MHz}$ ).

### 3.1.1 Distancia máxima y frecuencia inicial ficticia

En el apartado anterior se explica que el vector que contiene un periodo de la señal *sweep* contiene  $N$  muestras, que fijan la resolución espacial del sistema. Para simular la etapa de mezclado del sistema FMCW, es necesaria la multiplicación muestra a muestra del vector que contiene la señal *sweep* con una versión de sí mismo retardada. Si estos vectores contienen  $N$  muestras cada uno, su multiplicación contendrá un número menor de muestras. Este menor número de muestras implica una menor duración temporal del resultado del mezclado (virtualmente un  $T_m$  más pequeño), y por lo tanto puede falsear la anchura del lóbulo principal de los blancos detectados (Figura 14).

Se ha tomado la decisión de generar productos de  $N$  muestras útiles y de duración  $T_m$ , por lo que se hace necesario generar primero vectores de un número mayor de muestras. De manera arbitraria se generarán vectores de  $2N$  muestras, y sólo haremos la multiplicación de  $N$ . Podemos observar este proceso en la siguiente figura:

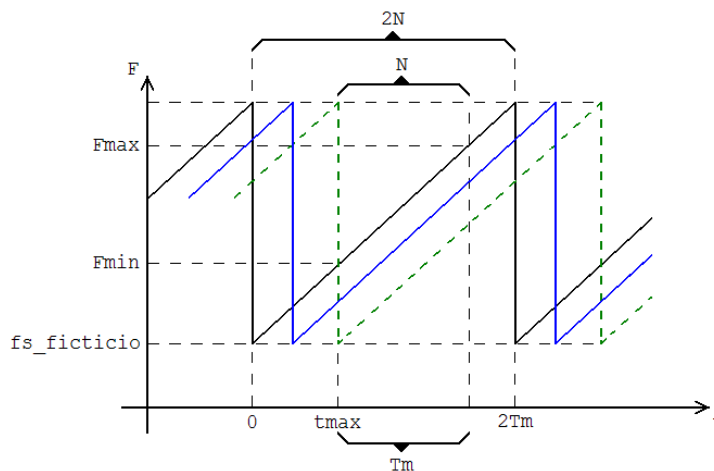


Figura 17: Multiplicación de  $N$  muestras útiles de un periodo de la señal *sweep*

Como se puede observar en la Figura 17, es necesario definir un tiempo  $t_{max}$  que indica el comienzo de las  $N$  muestras útiles. Este tiempo es, por definición, el tiempo máximo que se puede retardar el vector *sweep*, asegurando  $N$  muestras útiles, y por lo tanto fija una distancia  $d_{max}$ , que es la distancia máxima a la que se puede encontrar un blanco de nuestro sistema. También es necesario definir una nueva frecuencia inicial para el periodo de  $2N$  muestras de la señal *sweep*, y así conseguir que las  $N$  muestras útiles estén en el margen frecuencial de  $F_{min}$  a  $F_{max}$ . Llamaremos  $f_{s\_ficticio}$  a esta nueva frecuencia, que viene dada por la expresión:

$$f_{s\_ficticio} = F_{min} - \frac{D_f \cdot t_{max}}{T_m} \tag{3.1}$$

Donde  $f_{\min}$  es la frecuencia inicial de la señal *sweep* ( $f_0$ ),  $T_m$  es el periodo ( $T_m$ ) y  $\Delta f$  es el margen de frecuencias del barrido frecuencial ( $\Delta f_{dev}$ ). Por ejemplo para una distancia máxima  $d_{\max}=200\text{m}$ , la frecuencia inicial ficticia  $f_{s\_ficticia}$  toma un valor de  $9,96\text{MHz}$ .

### 3.1.2 Generación del vector que contiene la señal seep

Con los parámetros definidos anteriormente, la generación del vector que contiene la señal *sweep* (*sweep*) se limita a aplicar la expresión (2.11). Para simplificar los resultados, se ha tomado  $\varphi_0=0$ ,  $v_0=0$  y  $\Delta v=1$ , ya que el valor inicial de estos parámetros no afecta significativamente al resultado de las simulaciones. Obtenemos finalmente un vector que contiene  $2N$  muestras, de las cuales  $N$  son las muestras útiles correspondientes a un periodo de la señal *sweep*:

$$sweep = \cos\left(2\pi\left(f_{s\_ficticio} \cdot t + \frac{K \cdot t^2}{2}\right)\right) \quad (3.2)$$

Donde  $t$  es un vector de  $2N$  muestras de tiempo equiespaciadas de duración  $2T_m$ .

## 3.2 Procedimientos de simulación

En este apartado se describen varios procedimientos utilizados para hacer las simulaciones. Aunque algunos de ellos se usarán prácticamente en todas las simulaciones, otros son específicos para casos concretos, como por ejemplo para el caso de dos blancos simultáneos. Todos los resultados de las simulaciones se han hecho a partir de programas que utilizan estos procedimientos, además de la señal *sweep*, combinándolos convenientemente en cada caso concreto.

### 3.2.1 Vector recibido: Retardo del vector enviado

En este procedimiento se explica una manera de retardar el vector enviado para simular reflexiones en blancos.

Definimos una nueva constante llamada *blanco*, que contiene la distancia en metros a la que se encuentra el blanco de estudio. El retardo de la señal *sweep* se ha hecho mediante la concatenación de un vector al principio del vector *sweep* (inicializado a cero en nuestro caso, pero en principio podría tener cualquier valor) de longitud igual al número de muestras asociado al retardo que induce el blanco en la señal recibida. La longitud de este vector se calcula a partir de la distancia del blanco como:

$$t_d = \frac{2 \cdot \text{blanco}}{c} \quad (3.3)$$

$$\text{long\_vect0} = \lfloor F_{\text{sample}} \cdot t_d \rfloor$$

Donde  $t_d$  es el retardo en segundos inducido por el blanco, y  $long\_vect0$  es la longitud del vector de ceros.

La longitud del vector de ceros tiene que ser entera, por lo tanto tomaremos sólo la parte entera del valor  $long\_vect0$ . Aquí podemos comprobar cómo la resolución (que es el tiempo que transcurre entre dos muestras consecutivas) limita el modelo de simulación, ya que dos blancos separados una distancia menor que esta resolución, darían lugar a la misma longitud del vector\_retardo, y por lo tanto no podríamos distinguir entre ellos.

Después de concatenar el `vector_retardo` al principio del vector `sweep`, sólo nos quedamos con las  $N$  muestras útiles definidas anteriormente. Lo podemos ver en la Figura 18:

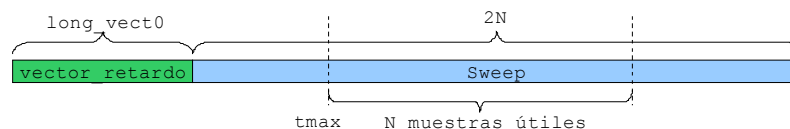


Figura 18: Nuevo vector `sweep` con  $N$  muestras útiles

De esta manera definimos el vector `señal_recibida` que contiene las  $N$  muestras posteriores a la posición  $t_{max}$  (en número de muestras).

### 3.2.2 Multiplicación de las señales enviada y recibida

Este procedimiento corresponde a la etapa de mezclado del método FMCW.

Igual que con el vector `señal_recibida`, sólo cogeremos las  $N$  muestras del vector `sweep` posteriores a la posición  $t_{max}$ . Guardaremos estas muestras en el vector `señal_enviada`, que es el que contiene en realidad el tono cuya frecuencia crece linealmente desde  $F_{min}$  hasta  $F_{max}$  en un tiempo  $T_m$ .

Con los vectores `señal_enviada` y `señal_recibida` ya definidos, sólo queda hacer la multiplicación muestra a muestra de ellos. El resultado lo guardaremos en el vector `r`.

### 3.2.3 Suma de $n$ periodos de la señal sweep

El proceso descrito en los apartados 3.2.1 y 3.2.2 es sólo para un periodo de la señal `sweep`. Para los demás periodos basta con repetir el proceso y sumar cada vector `r` iésimo en un vector acumulado `r`. Como se describe en el apartado 2.2.5, como la fase inicial del resultado de cada periodo es la misma, al sumar diferentes periodos estamos sumando la señal en fase, y por lo tanto su potencia crece más deprisa que la del ruido. Añadiendo ruido aditivo a las simulaciones se puede observar este efecto:

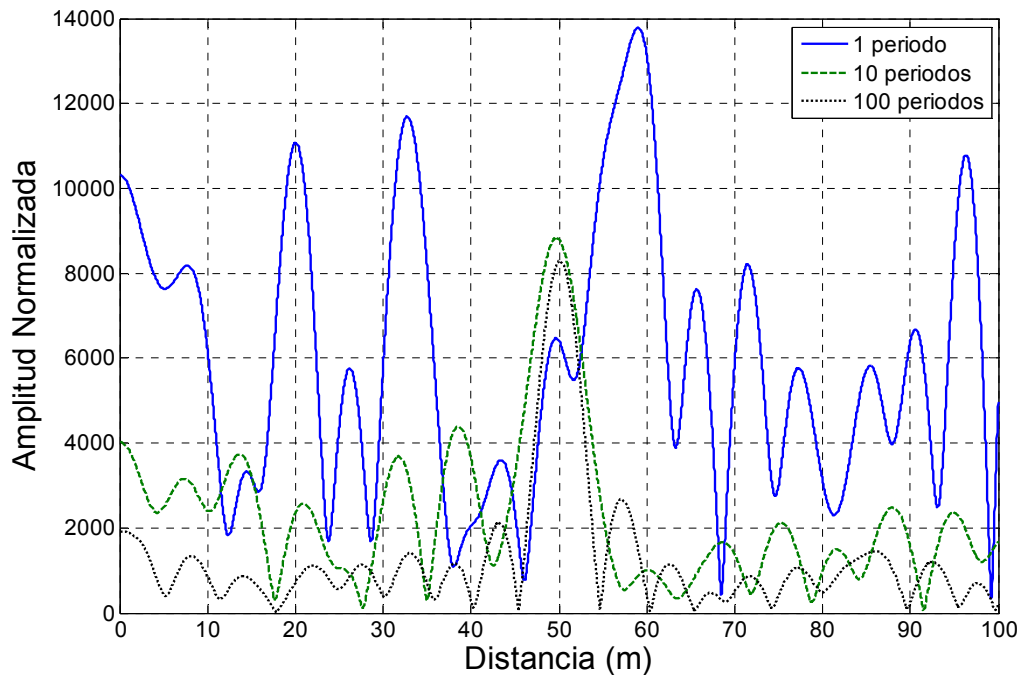


Figura 19: Simulación de un blanco en presencia de ruido para varios promediados de periodos.

En la Figura 19 se observa como el ruido se reduce a medida que se aumenta el número de periodos promediados. De todas maneras, en la mayoría de las simulaciones que se han hecho, no se ha añadido ruido, por lo que la información es exactamente la misma independientemente del número de periodos promediados, por lo que estas simulaciones se harán con  $n=1$ .

### 3.2.4 Cálculo de la FFT y presentación

En la mayoría de programas de simulación utilizados durante el proyecto, el resultado es una gráfica que representa la posición de los blancos detectados. En este procedimiento se explica cómo se ha hecho la Transformada de Fourier (en este caso la FFT), y su posterior representación.

La información que se pretende extraer de la transformada se encuentra en una parte muy pequeña del espectro, por lo que contiene muy pocas muestras comparadas con la longitud del vector de señal. Con el fin de obtener más muestras útiles en esta parte del espectro, se hace la FFT de la señal a la salida del mezclador con un número de puntos mayor al número de puntos  $N$ . Esto es equivalente a añadir ceros al vector a transformar. En la Figura 20 se ha representado el módulo de la Transformada de Fourier de la respuesta a un blanco situado a 50m:

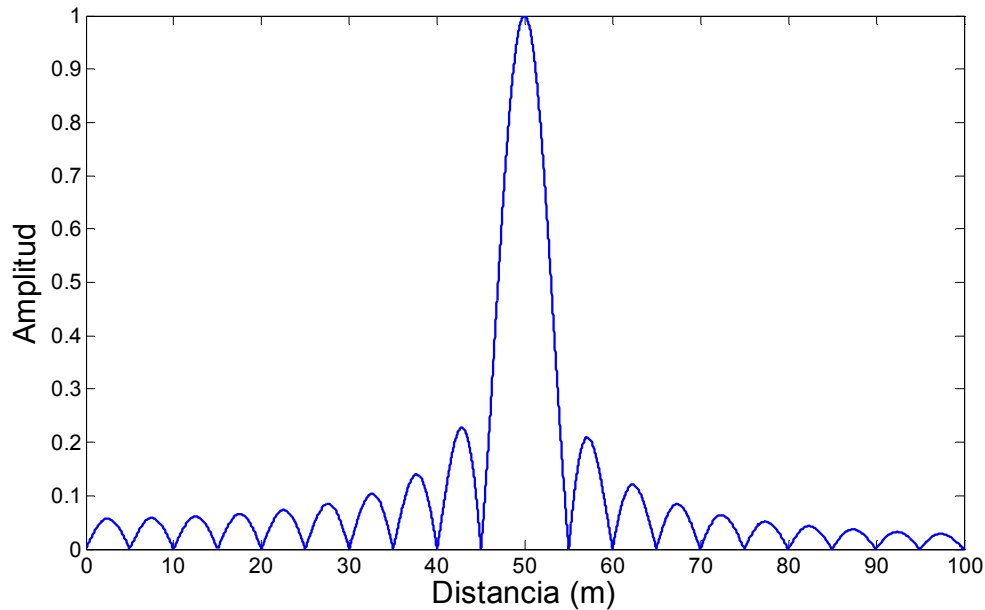


Figura 20: Simulación de un blanco a 50m (50 a 80MHz)

Podemos observar que en la Figura 20 se ha reescalado el eje de frecuencias para representar directamente la distancia asociada a los blancos. El modelo simulado muestra correctamente la detección de un blanco situado aproximadamente a 50m. Vemos también en esta figura que el blanco no viene representado por una delta frecuencial, sino por una función *sinc*. Esto es debido a que el periodo de la señal *sweep* es finito, y por lo tanto el vector  $x$  contiene un tono de longitud finita.

### 3.3 Simulaciones de un blanco

En este apartado se pretende ilustrar el comportamiento general de las simulaciones, representando los resultados obtenidos al simular un blanco, así como las señales intermedias involucradas. También se explica cómo afectan los parámetros iniciales a los resultados de las simulaciones, comparando estos resultados con los predichos con la teoría.

Se parte de la simulación de la señal *sweep*. Esta señal es la que se envía a los blancos y es recibida después de reflejarse en éstos.

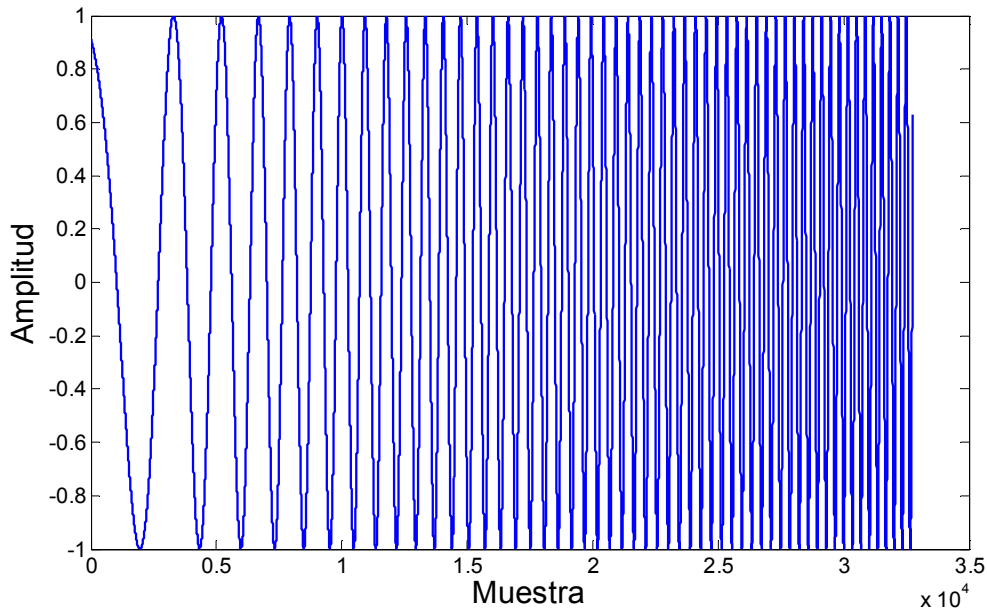


Figura 21: Señal *sweep* generada.

En la Figura 21 está representada la señal *sweep* enviada a los blancos. Se han utilizado valores de  $f_0$  y  $\Delta f_{dev}$  pequeños, para poder representar la señal y que sea visible a simple vista. La señal recibida después de rebotar en el blanco de estudio será la misma que esta, pero retrasada cierto número de muestras.

Después de mezclar las señales emitida y recibida, se obtiene:

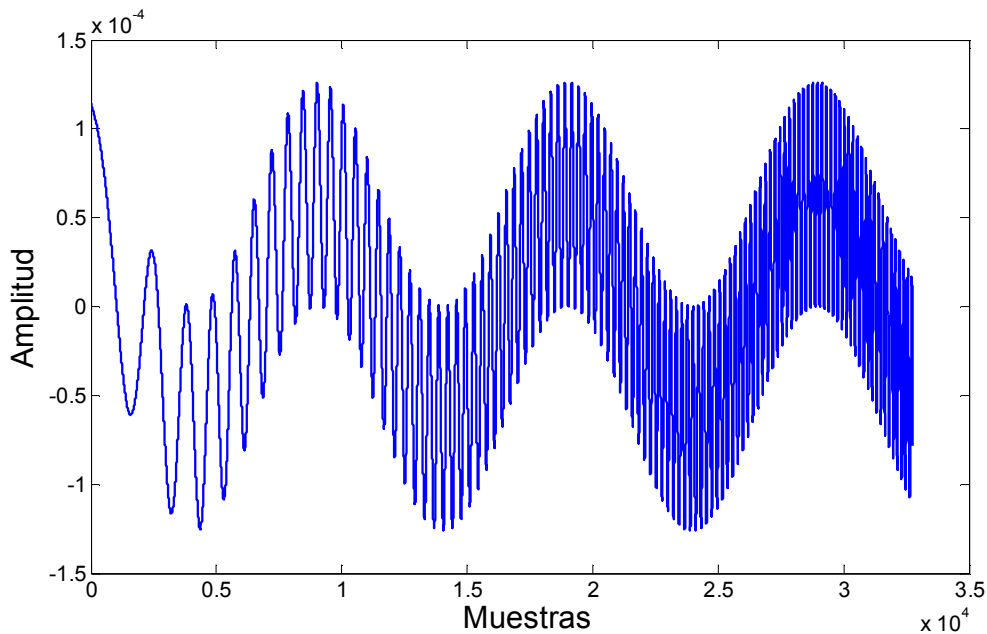


Figura 22: Señal a la salida del mezclado, antes de filtrar la componente suma.

En la Figura 22 se observa un tono de baja frecuencia sumado a otro tono que aumenta su frecuencia en el tiempo. Estos dos tonos corresponden a la componente resta y a la componente suma frecuencial respectivamente. Haciendo la Transformada de Fourier de esta señal obtenemos:

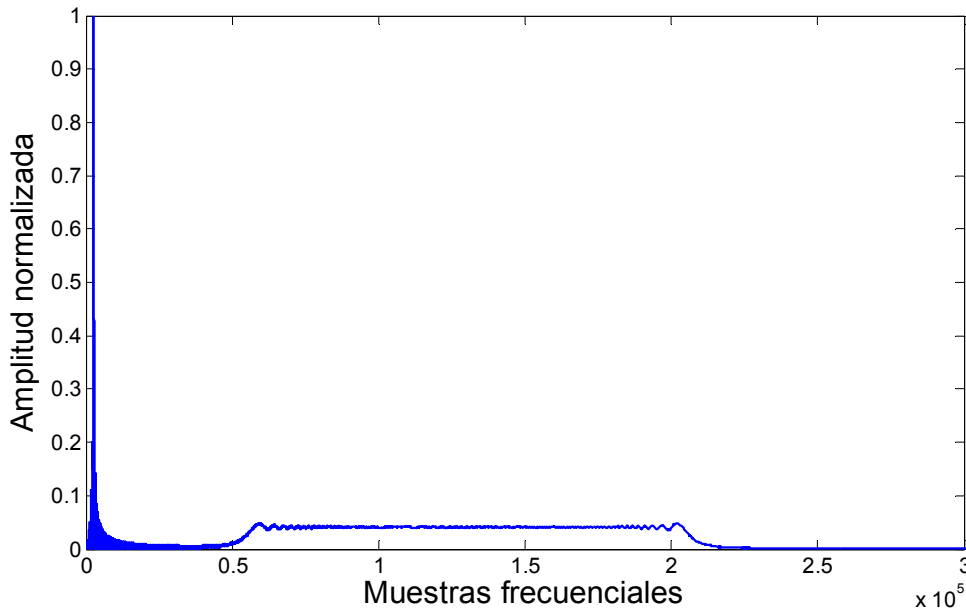


Figura 23: Espectro a la salida del mezclador para un blanco.

En la representación del espectro de esta señal (Figura 23) se observa este tono a baja frecuencia y de gran potencia, asociado a la frecuencia resta y por lo tanto asociado al blanco detectado. A alta frecuencia se encuentra el espectro correspondiente a la componente suma, que debido a su ensanchamiento espectral, tiene una amplitud mucho menor que la componente resta.

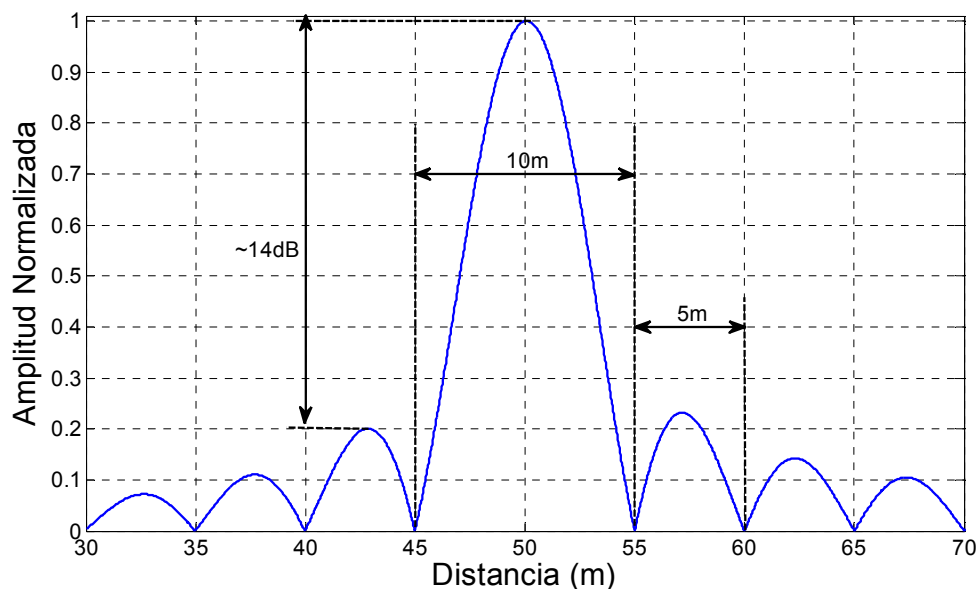


Figura 24: Simulación de un blanco a unos 50m (10 a 40MHz), representando los parámetros más representativos de la función resultante.

A la señal a la salida del mezclador se debe aplicar un filtro paso-bajo para eliminar la componente suma del batido frecuencial, ya que la información que se está buscando se encuentra en la componente resta. En las simulaciones, este filtrado no es necesario, ya que se puede escalar la gráfica resultante, eliminando la alta frecuencia de la representación, y centrándose en la observación únicamente de la componente resta. Además, las frecuencias resultantes del análisis están relacionadas directamente



con la distancia a la que se encuentra el blanco de estudio (2.22), por lo que reescalando el eje horizontal se puede obtener directamente la representación de la distribución de los blancos detectados en función de la distancia.

En resultado de una simulación completa de un blanco a 50m se puede observar en la Figura 24. Se observa que el resultado es una *sinc* centrada en la posición teórica del blanco, cuya distancia entre ceros consecutivos son 5m, y la distancia entre ceros del lóbulo principal son 10m. Estos resultados se ajustan a los predichos por la teoría para un margen de frecuencias de  $\Delta f_{dev} = 30MHz$  (2.25). Para simulaciones (o medidas) de un blanco, este margen de frecuencias es el parámetro fundamental que determina la anchura de la *sinc*, y por lo tanto la resolución en distancia.

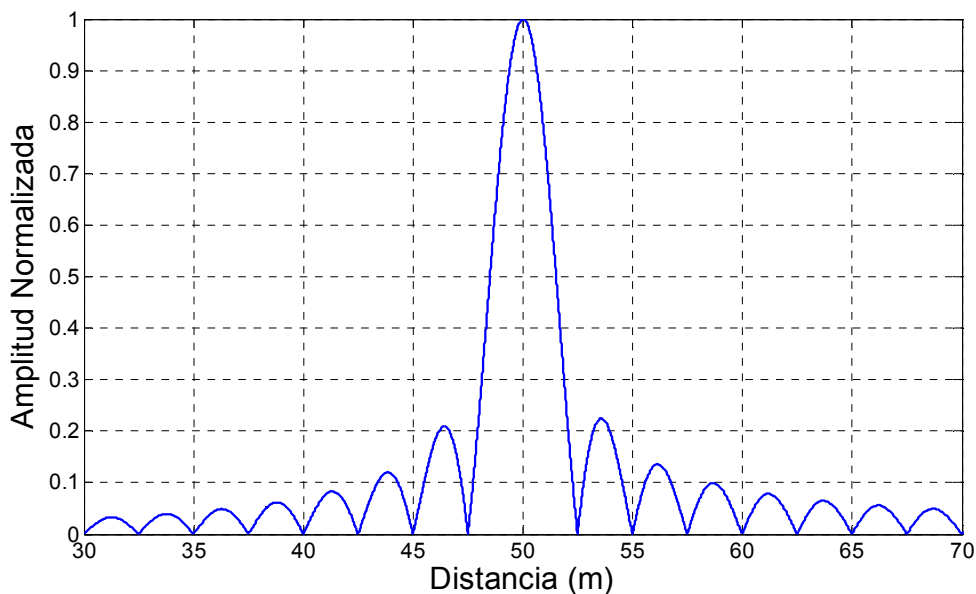


Figura 25: Simulación de un blanco a unos 50m (20 a 80MHz).

En la Figura 25 está representada la simulación de un blanco a 50m, pero esta vez el margen de frecuencias va de 20 a 80MHz, y por lo tanto su margen de frecuencias es de  $\Delta f_{dev} = 60MHz$ . Como el margen de frecuencias es el doble que para el caso anterior, la anchura del lóbulo principal se reduce a la mitad (5m), por lo que la resolución en distancia también es la mitad.

En simulaciones (o medidas) de más de un blanco, además del margen de frecuencias, también será un parámetro determinante la frecuencia inicial del sistema ( $f_0$ ), ya que aunque no influye en la resolución en distancia, determina la fase relativa entre blancos (2.45), y por lo tanto la forma en que se combinan.

### 3.4 Interacción de dos blancos

Como ya hemos mencionado anteriormente, el periodo finito de la señal *sweep* implica que el resultado del método FMCW consista en *sincs* frecuenciales centradas en la posición de cada blanco. En el caso de más de un blanco, las *sincs* frecuenciales se sumarán en amplitud y fase. Si los blancos están suficientemente separados, la interacción de un blanco sobre los demás será prácticamente despreciable. Para el

caso en que se encuentren a distancias del orden de la anchura del lóbulo principal de las *sincs*, se producirán efectos de interferencia (constructiva o destructiva) que pueden falsear la potencia e incluso la posición de los blancos detectados.

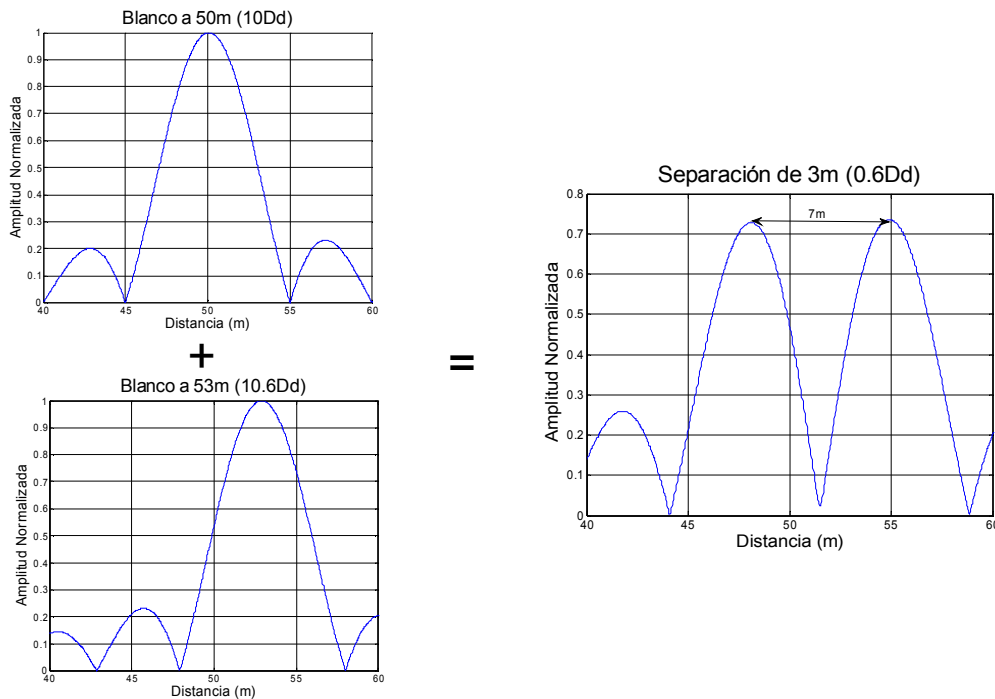


Figura 26: Simulación de dos blancos a 50m y 53m, por separado y juntos (10 a 40MHz).

En la Figura 26 se están representadas las simulaciones de dos blancos separados 3m juntos y por separado. Se puede apreciar que al simularlos juntos se interfieren el uno con el otro, causando la incorrecta detección de la distancia entre ambos.

Esta interacción entre *sincs* viene determinada por los parámetros  $f_0$  y  $\Delta f_{dev}$ , tal y como se ve en la expresión de la fase relativa (2.45), ya que el producto  $KT_m$  es directamente el margen de frecuencias  $\Delta f_{dev}$  (2.23). En lo que sigue de este capítulo se ilustra la combinación de dos blancos para diferentes pares de estos parámetros, estudiando la interacción (en potencia y distancia) de dos blancos, estimando el error asociado a la interferencia que ejercen el uno sobre el otro.

### 3.4.1 Errores en amplitud y distancia

Con el fin de estudiar las interferencias que se producen al simular dos o más blancos, se propone un procedimiento para hacer una serie de simulaciones de dos blancos separados diferentes distancias. Se estima la posición y potencia de los blancos detectados, y se comparan con los valores teóricos utilizados para la simulación. Con estos datos podremos representar el error en potencia y en posición sufrido por un blanco al verse interferido por otro.

Se han utilizado los procedimientos descritos en los apartados I.1 y I.2 para hacer una serie de simulaciones de dos blancos en las que uno de ellos se mantiene a 50m de origen, y el otro (partiendo de 50m) se acerca paulatinamente al origen. Para cada una de las simulaciones de dos blancos, se estima la posición y potencia de los blancos detectados utilizando el algoritmo descrito en el apartado **iError! No se encuentra el**

**origen de la referencia..** Restando la posición detectada de cada blanco con su posición teórica se pueden obtener gráficas de error en distancia.

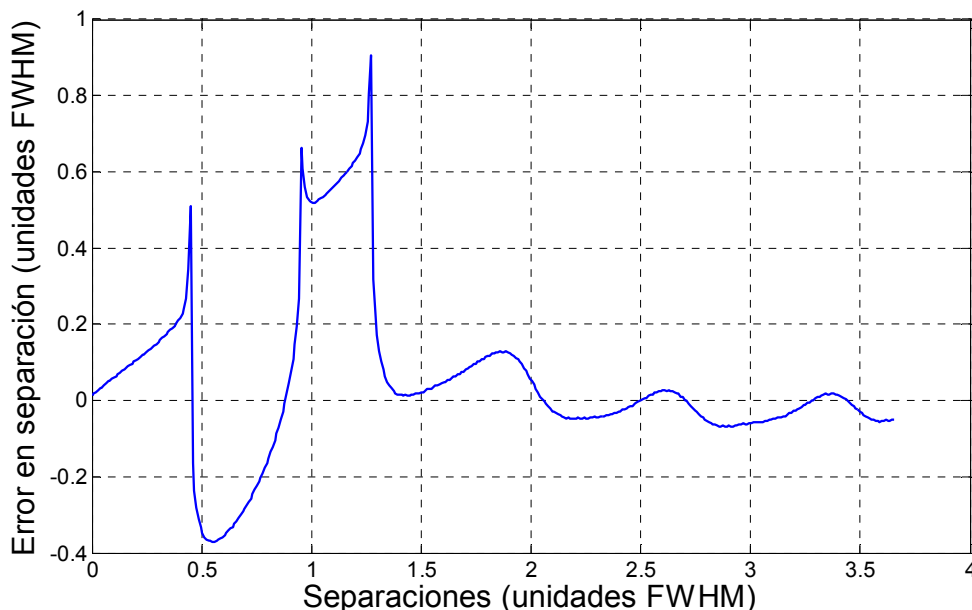


Figura 27: Error en distancia de la detección del primer blanco en una simulación de dos blancos (10 a 40 MHz).

En la Figura 27 está representado el error en la posición detectada del primer blanco cuando su posición teórica se separa de otro blanco fijo. En el eje de abscisas está representada la separación teórica entre estos dos blancos en unidades FWHM (Full Width at Half Maximum), que es aproximadamente la distancia entre ceros de la *sinc* resultante  $\Delta d$  (2.25). Para este caso, con una  $\Delta f_{dev}$  de 30MHz, obtenemos una resolución en distancia de 5m. Si el error es positivo, significa que el blanco se está detectando más alejado que su posición teórica, mientras que si el error es negativo, se está detectando más cerca del origen. Se puede observar que en los intervalos  $[0,0.4]$ , y  $[0.9,1.3]$  la gráfica es prácticamente una recta de pendiente  $\frac{1}{2}$ . En estos intervalos se está detectando sólo un máximo que engloba los dos blancos en el punto medio de sus posiciones teóricas, y por lo tanto el error introducido es la mitad de la separación teórica. Se observa también que este error tiende asintóticamente a cero al aumentar la separación entre los dos blancos, ya que si los blancos están lo suficientemente separados, su interacción es prácticamente nula.

Debido a la simetría del sistema, la gráfica correspondiente al segundo blanco es la una imagen especular de la del primer blanco, ya que cada gráfica esta hecha "desde el punto de vista de ese blanco", y los errores que para uno son positivos, para el otro son negativos. Se puede ver esta gráfica en la Figura 28.

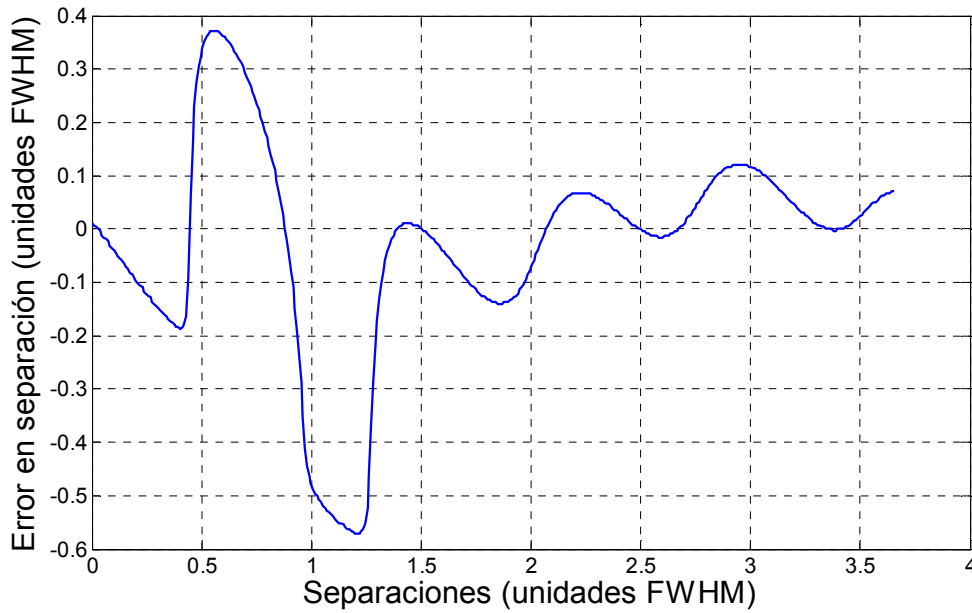


Figura 28: Error en distancia de la detección del segundo blanco en una simulación de dos blancos (10 a 40 MHz).

Análogamente, se puede representar la amplitud (normalizada respecto de un blanco) del primer blanco detectado para cada separación, dando lugar a la siguiente figura:

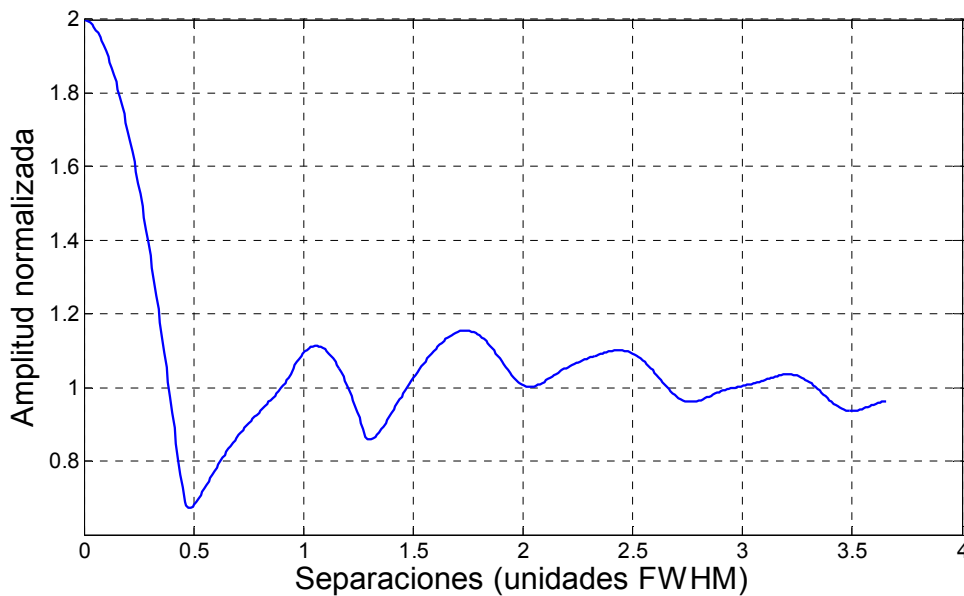


Figura 29: Amplitud normalizada de la detección del primer blanco en una simulación de dos blancos (10 a 40 MHz).

En la gráfica de la Figura 29 se observa que los dos blancos interfieren constructiva (puntos 0 y 1) o destructiva (puntos 0.5 y 1.3) en función de la separación de los blancos. Como en el caso anterior, cuando los blancos están lo suficientemente separados no interfieren entre ellos, por lo que la gráfica tiende asintóticamente a 1.

Debido también a la simetría del sistema, la gráfica correspondiente a la amplitud normalizada del segundo blanco es prácticamente la misma que para el primer blanco, ya que los dos blancos se detectan prácticamente a la misma amplitud. Se puede ver esta gráfica en la Figura 30.

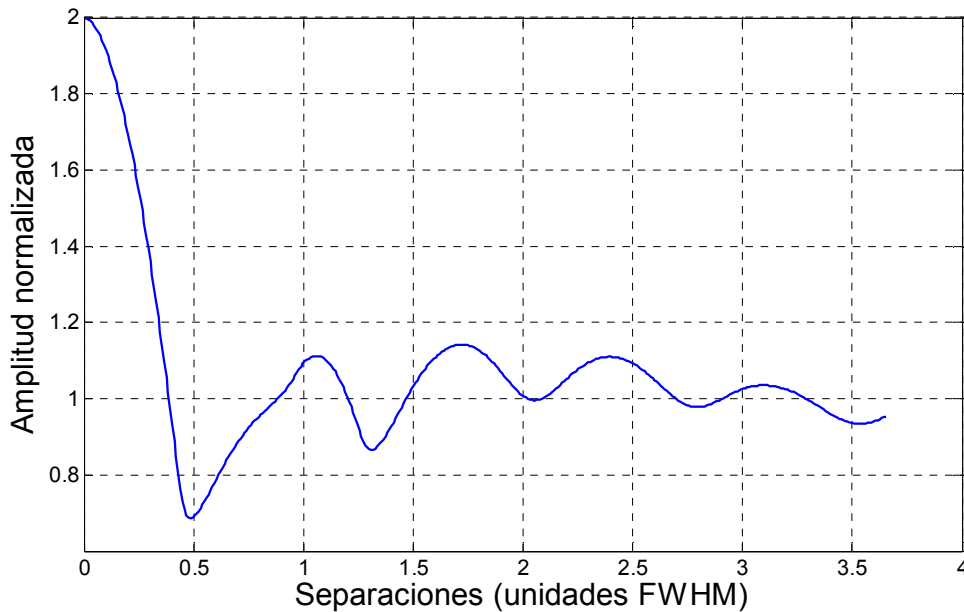


Figura 30: Amplitud normalizada de la detección del segundo blanco en una simulación de dos blancos (10 a 40 MHz).

Para entender mejor las gráficas correspondientes a los errores en distancias, se puede representar la posición esperada y detectada de los blancos a cada separación, consiguiendo la siguiente gráfica:

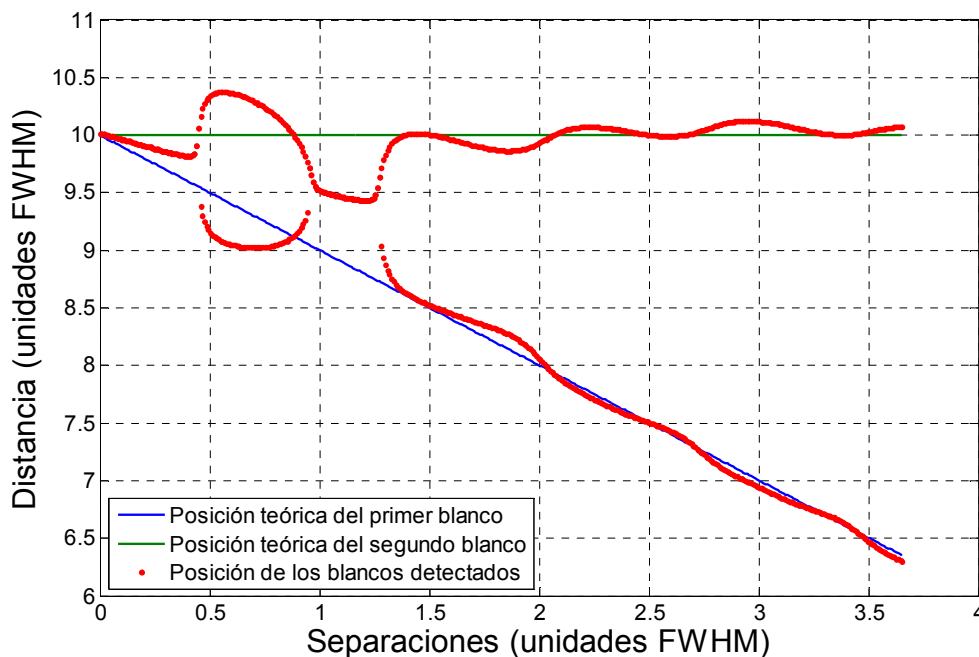


Figura 31: Simulación de la posición teórica y detectada de dos blancos a diferentes separaciones (10 a 40 MHz)

En el eje horizontal de la Figura 31 está la separación teórica de los blancos simulados, en unidades FWHM. En el eje vertical está la posición absoluta del blanco teórico y simulado.

En la gráfica se observa que para separaciones menores de 2m (0,4FWHM) sólo se detecta un máximo aproximadamente en el punto medio de la posición teórica de los blancos. Aproximadamente a una separación de 2,5m (0,5FWHM) se empiezan a

detectar dos blancos, pero su evolución indica que están más separados que lo que deberían, debido a la interferencia que ejercen el uno sobre el otro. Entre 5 y 6m (1 a 1,2FWHM) de nuevo se detecta sólo un máximo, aunque los blancos están más separados que antes. Este es un claro ejemplo de cómo la interferencia entre blancos afecta de manera diferente para cada separación, debido a la fase relativa entre ambos (2.45). A partir de unos 6,5m (1,3FWHM) se vuelven a detectar dos máximos. Se observa cómo la posición medida tiene asintóticamente a la posición teórica, ya que mientras más separados estén los blancos, menor será su interferencia.

Variando los parámetros  $f_0$  y  $\Delta f_{dev}$ , y por lo tanto la fase relativa entre blancos, se consiguen resultados muy diferentes a las gráficas anteriores. Por ejemplo, variando la frecuencia inicial, pero manteniendo el margen de frecuencias:

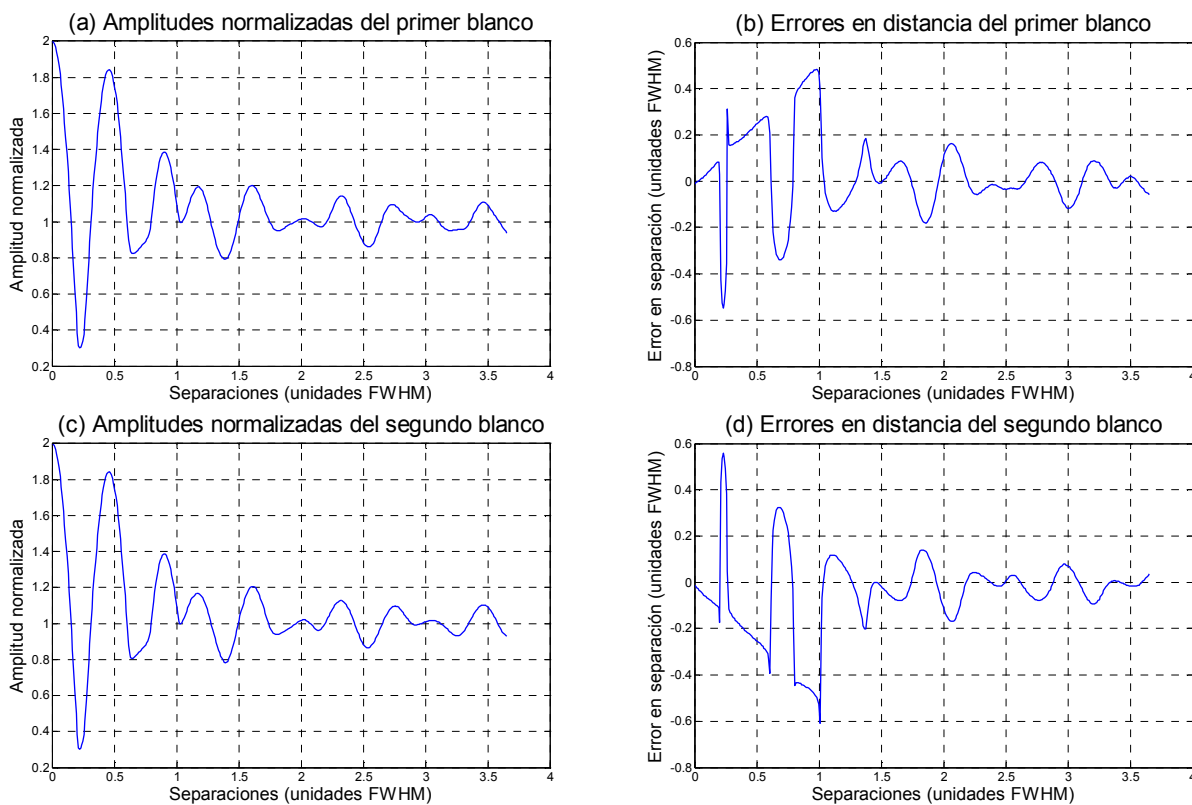


Figura 32: Errores en amplitud y distancia para simulaciones de dos blancos (50 a 80MHz).

Como se observa en la Figura 32, al variar la frecuencia inicial  $f_0$ , sin variar el margen de frecuencias  $\Delta f_{dev}$ , las gráficas de error en amplitud y en distancia cambian drásticamente. En el ejemplo, la frecuencia inicial es 50MHz en vez de 10MHz como en el caso anterior. Este aumento en frecuencia causa que la evolución de la posición de los máximos se produzca "más rápido", ya que la fase relativa entre blancos depende de la frecuencia inicial  $f_0$  (2.45).

Además de la frecuencia inicial  $f_0$ , se puede variar también el margen de frecuencias  $\Delta f_{dev}$ , por ejemplo aumentándolo, y en consecuencia hacer que la resolución espacial del sistema disminuya, teniendo que estar los dos blancos más juntos para producir resultados similares.

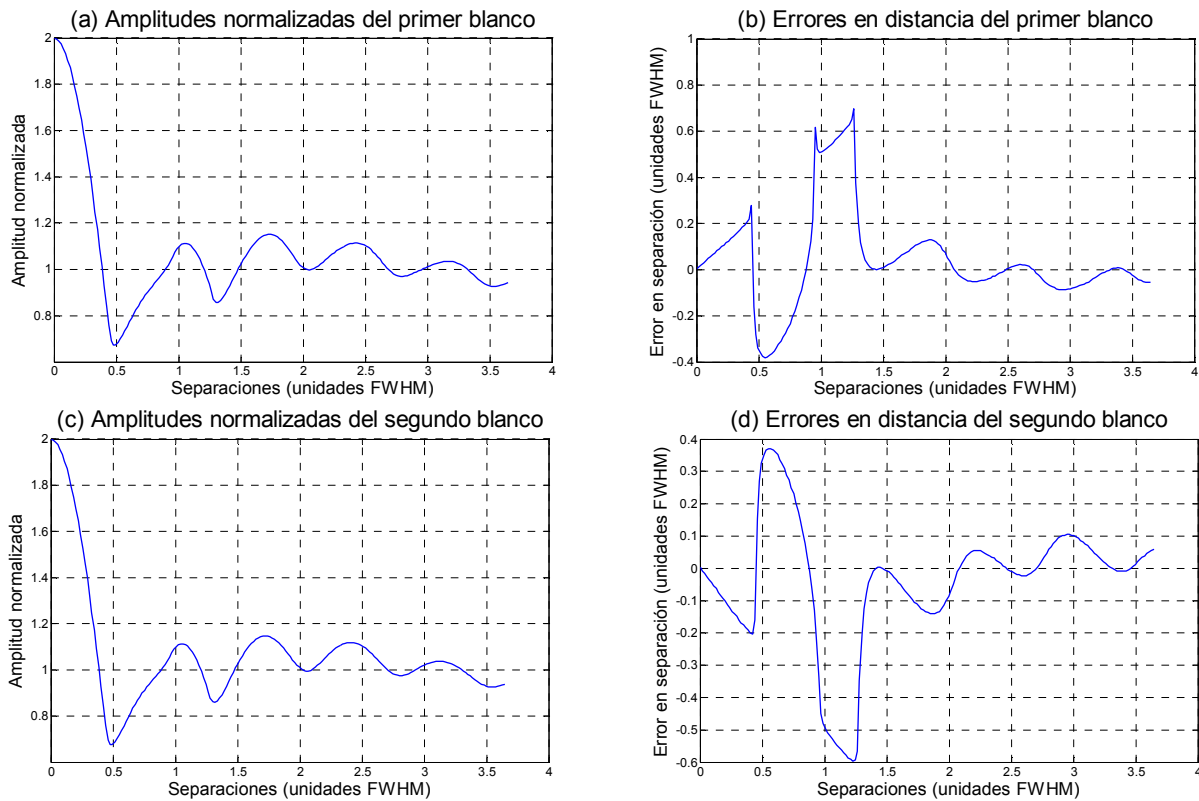


Figura 33: Errores en amplitud y distancia para simulaciones de dos blancos (20 a 80MHz).

En la Figura 33 se han representado los errores en amplitud y distancia, pero para el caso de frecuencias que van de 20 a 80MHz, y por lo tanto el caso con  $f_0=20MHz$  y  $\Delta f_{dev} = 60MHz$ . En este caso la frecuencia inicial y el margen de frecuencias son exactamente el doble que en caso inicial, en que las frecuencias iban de 10 a 40MHz. Este hecho provoca que las gráficas que salen sean prácticamente las mismas (no son exactas debido a los propios errores de simulación) que para el primer caso (Figura 27, Figura 28, Figura 29 y Figura 30), pero teniendo en cuenta que en este caso una unidad FWHM equivale a 2,5m, por lo que estos cambios suceden con los blancos más próximos.

### 3.5 Interacción de tres blancos

Con el fin de acercar un poco más el estudio de blancos interferentes a la realidad, en este apartado se estudia la interacción de tres blancos a distancias del orden de  $\Delta d$ . En este proyecto no se ha pretendido cubrir todos los casos posibles de interacción entre tres blancos, ya que se debería de generar un número significativo de gráficas sin relación aparente entre ellas. Con el fin de ilustrar el comportamiento general de la interacción entre tres blancos, se ha centrado el estudio en dos casos: dos blancos fijos separados  $\Delta d$  y otro que se aleja del primero de ellos; y un blanco fijo del que se separan los otros dos, uno alejándose del origen mientras que el otro se acerca, de manera que los tres blancos están siempre equidistantes.

### 3.5.1 Dos blancos fijos

El programa de simulación propuesto funciona de la misma manera que el de dos blancos del apartado anterior, pero añadiendo otro blanco fijo. En este los blancos fijos se han colocado a 50 y 55m respectivamente, de manera que están separados una distancia  $\Delta d$  para el margen de frecuencias de 10 a 40MHz. Todo el procesado es análogo al del apartado anterior, pero teniendo en cuenta que se pueden detectar hasta tres blancos. Para el caso de frecuencias que van de 10 a 40MHz, la gráfica que representa la posición de los blancos detectados obtenida por esta simulación es la siguiente:

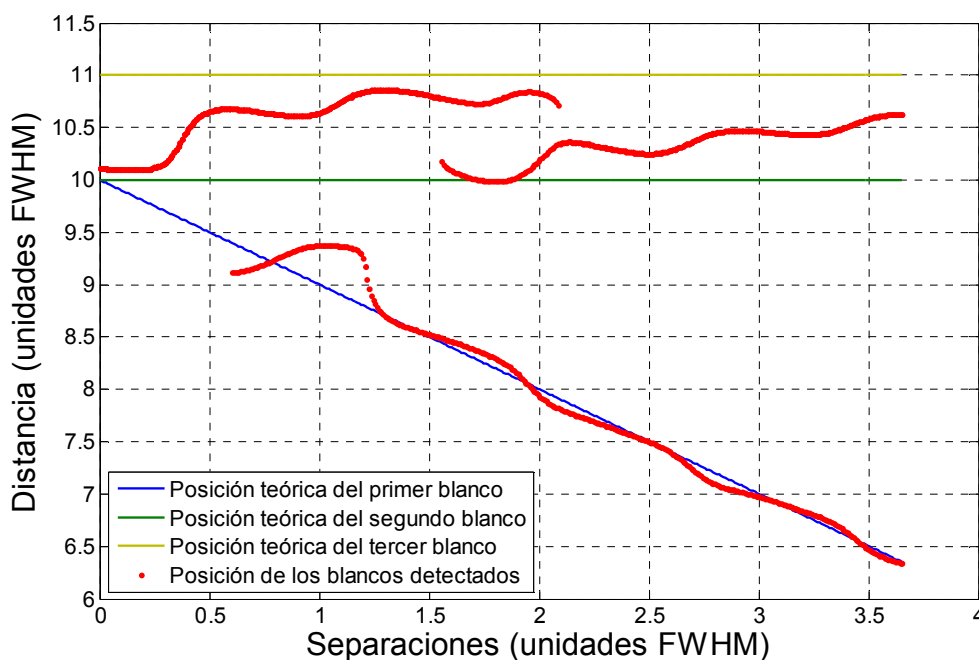


Figura 34: Simulación de la posición teórica y detectada de tres blancos, dos fijos y uno que se aleja de ellos (10 a 40 MHz)

En la Figura 34 se han simulado dos blancos fijos separados 5m (para este margen de frecuencias, 1 unidad FWHM) y un blanco más que se separa de los otros dos. En el eje de abscisas está representada la separación del primer blanco respecto del segundo. Se observa que para separaciones pequeñas del primer blanco, se detecta sólo un máximo, en principio más cercano al segundo blanco, ya que en la posición de partida el primero y el segundo blanco están muy próximos. En el intervalo [0.6,1.6] se detectan dos máximos. En el intervalo [1.6,2.1] se detectan los tres máximos, y por lo tanto es el intervalo donde mejor se detectan los tres blancos. Este hecho no deja de ser curioso, ya que sin la influencia del primer blanco, tenemos dos blancos separados 5m, que según la Figura 31 se detectan como un solo máximo aproximadamente a la distancia media entre ambos. Gracias a la interferencia del primer blanco se detectan el segundo y tercer blanco como dos blancos diferentes. Asintóticamente, la gráfica tiende a la posición teórica del primer blanco, y a la posición esperada de los otros dos blancos a esa separación (que en este caso es un máximo aproximadamente en el punto medio de las posiciones teóricas).

Como en el caso de dos blancos, se puede representar el error de detección asociado a cada blanco. En este caso, ya que la posición de los blancos no es simétrica, las gráficas resultantes para cada blanco pueden ser diferentes:



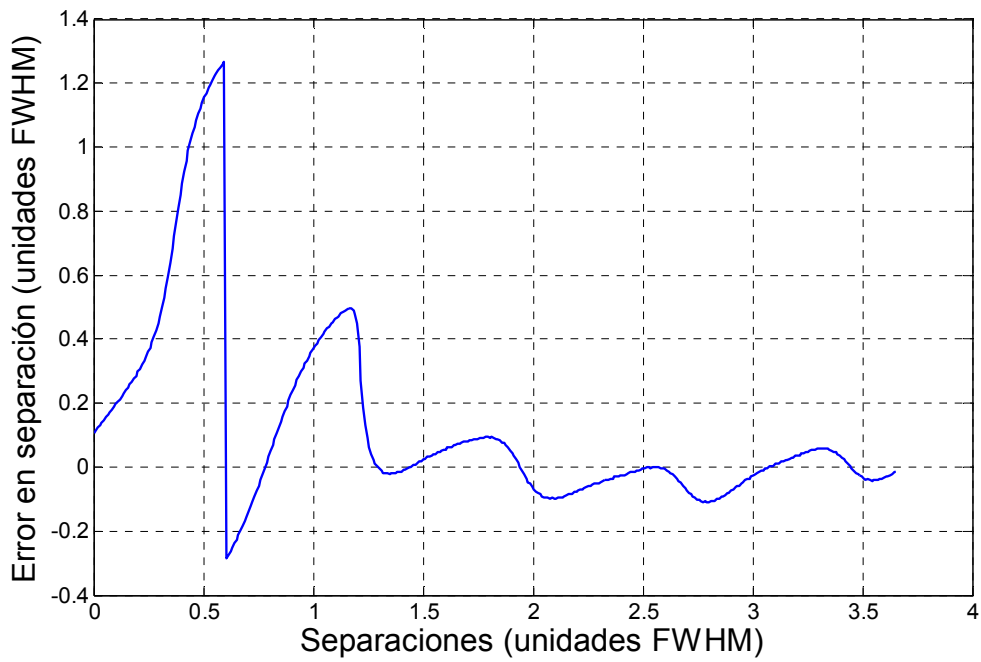


Figura 35: Error en distancia de la detección del primer blanco en una simulación de tres blancos (10 a 40 MHz)

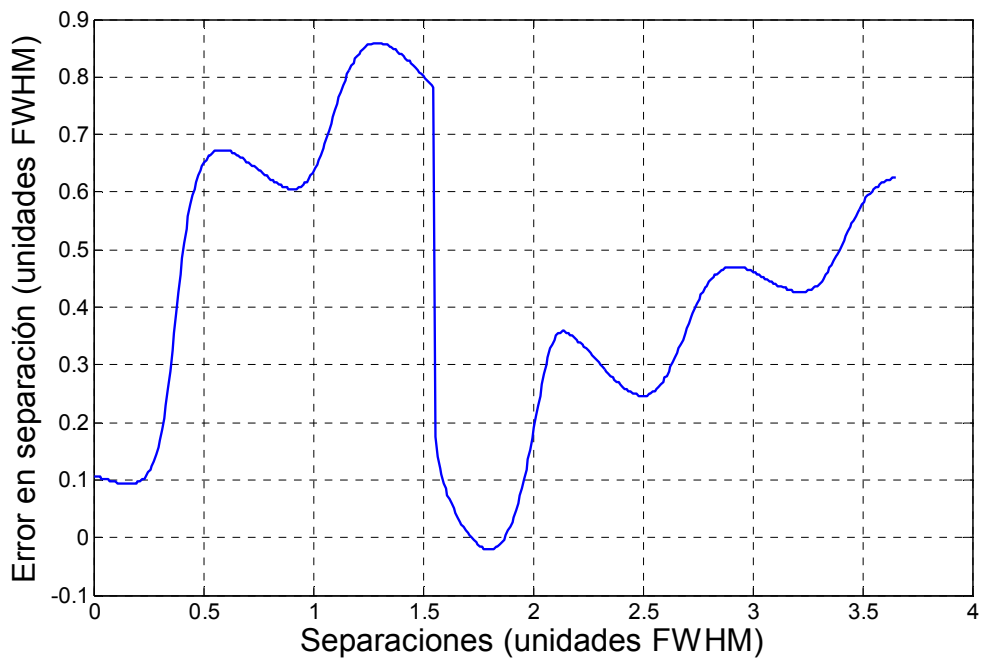


Figura 36: Error en distancia de la detección del segundo blanco en una simulación de tres blancos (10 a 40 MHz)

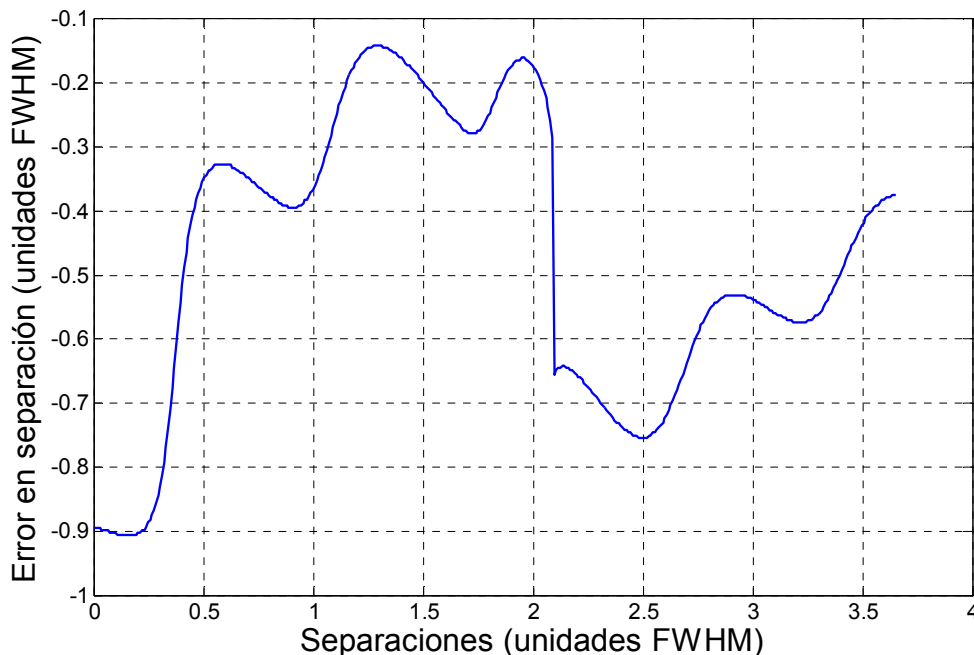


Figura 37: Error en distancia de la detección del tercer blanco en una simulación de tres blancos (10 a 40 MHz)

Las tres gráficas anteriores (Figura 35, Figura 36 y Figura 37) representan las mismas curvas mostradas en la Figura 34, pero "desde el punto de vista de cada blanco". Así para el primer blanco, la gráfica sigue la evolución del máximo que engloba los tres blancos, hasta la separación de 0.6FWHM donde "elige" la curva del máximo que minimiza el error en distancia. En estos puntos la curva resultante puede dar un salto, que en las gráficas está representado mediante una línea vertical, pero esto no significa un máximo cambie súbitamente de posición, sino que algún máximo aparece o desaparece, y en la representación se pasa a dibujar la evolución de otro de los máximos. Este efecto se puede observar en la Figura 38:

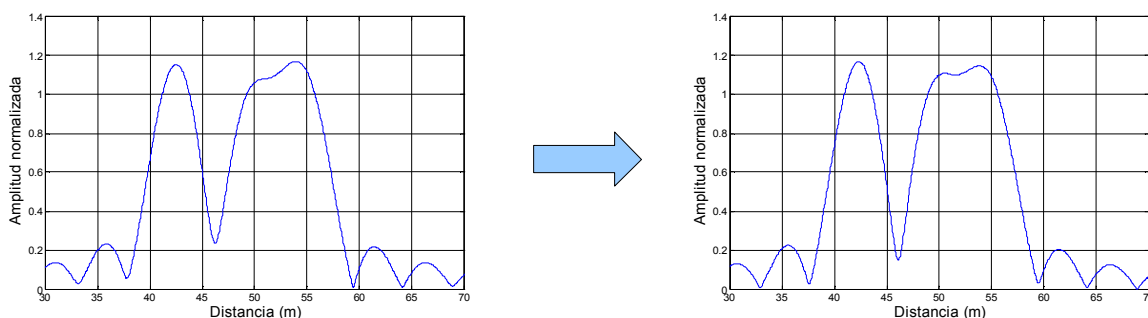


Figura 38: Aparición de un máximo detectado en una simulación de tres blancos.

En la iteración de la gráfica de la izquierda de la Figura 38 se detectan dos máximos, pero pocas iteraciones después, en la gráfica de la derecha, se detectan tres blancos, por lo que la posición detectada del segundo blanco (el de en medio) dará un salto, ya que antes se tomaba como segundo blanco detectado el máximo de la derecha.

La gráfica del primer blanco (Figura 35) tiende asintóticamente a 0, pero las gráficas del segundo (Figura 36) y tercer blanco (Figura 37) tienden a un valor de unos 0.5FWHM, positivo para el primer blanco (ya que "ve" este máximo más lejos) y negativo para el tercero (ya que lo "ve" más cerca).

También se pueden representar las amplitudes de los máximos detectados para cada blanco:

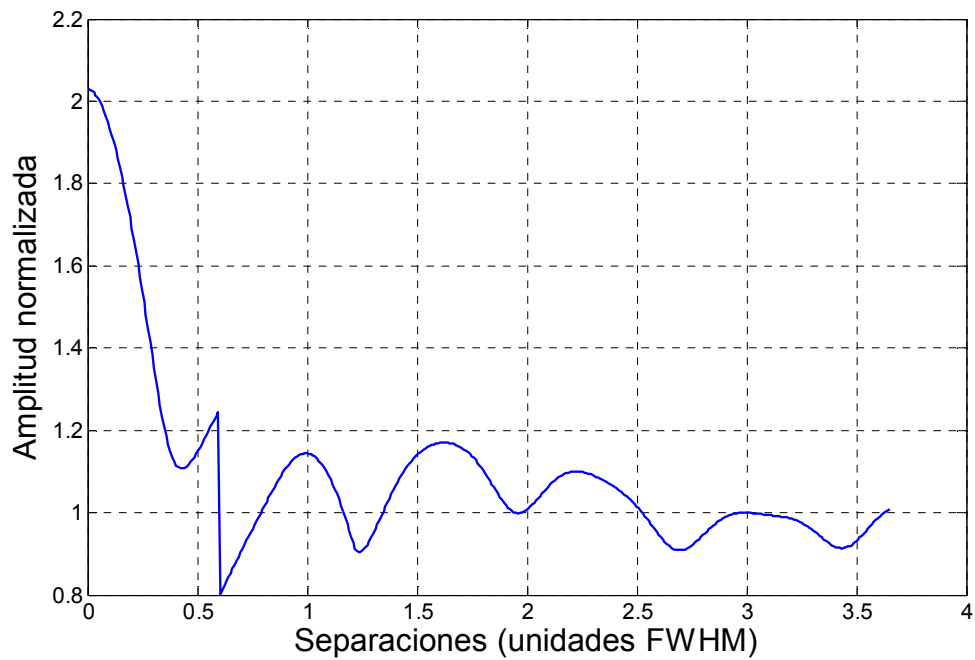


Figura 39: Amplitud normalizada de la detección del primer blanco en una simulación de tres blancos (10 a 40 MHz)

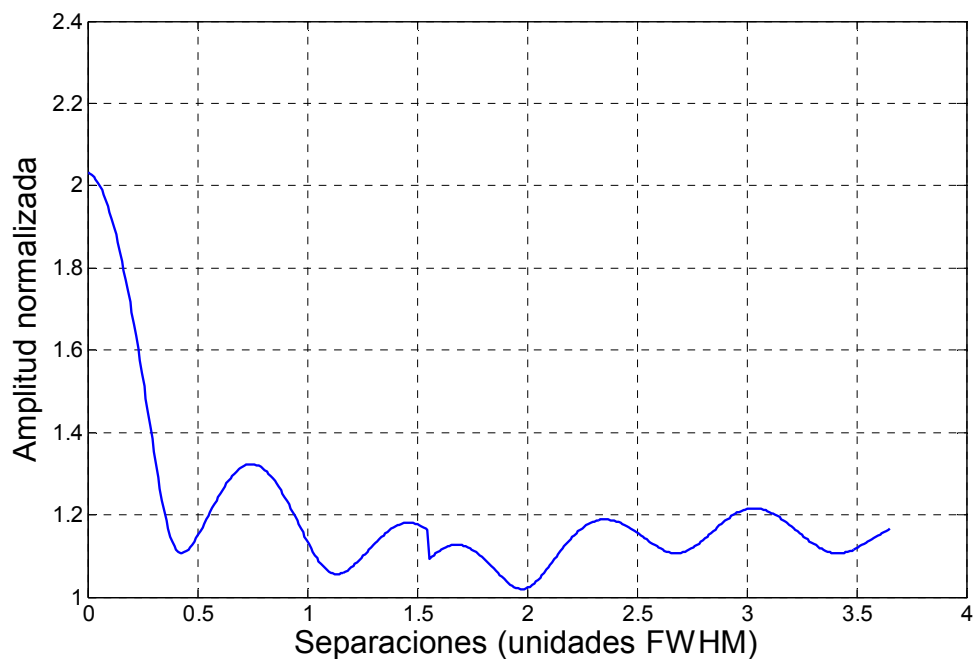


Figura 40: Amplitud normalizada de la detección del segundo blanco en una simulación de tres blancos (10 a 40 MHz)

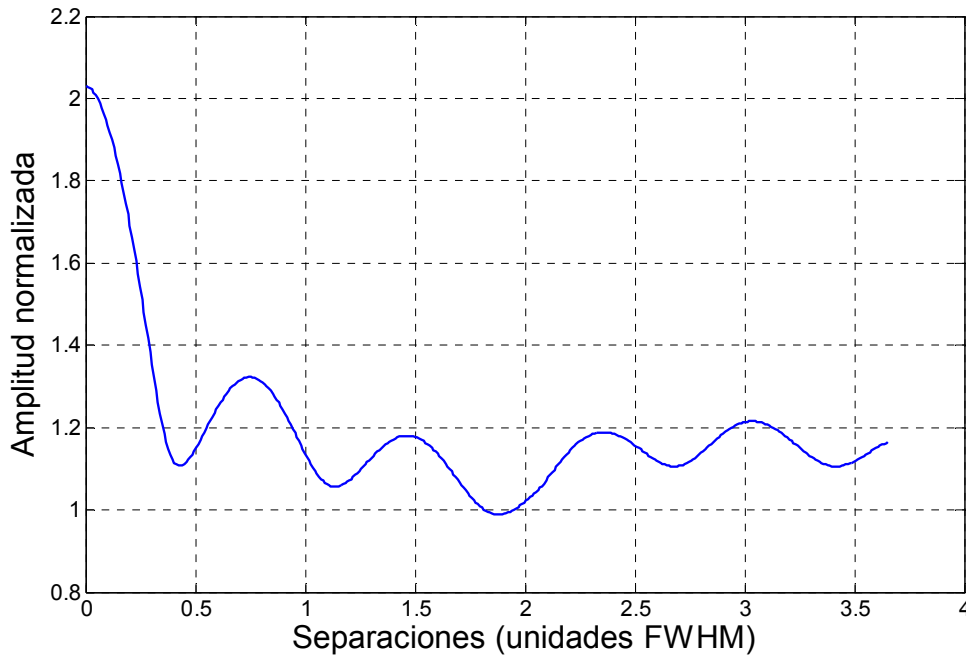


Figura 41: Amplitud normalizada de la detección del tercer blanco en una simulación de tres blancos (10 a 40 MHz)

Como en el caso de dos blancos, en las figuras anteriores (Figura 39, Figura 40 y Figura 41) se aprecian las interferencias constructivas y destructivas en la amplitud de los blancos detectados. Igual que en caso de distancias, estas gráficas tienden asintóticamente a 1 para el primer blanco y a aproximadamente 1.2 para los otros dos (ya que éstos se siguen interfiriendo aún en ausencia del primero). También se observan en estas gráficas saltos en las curvas en los mismos puntos que en las gráficas de error en distancia, y son debidos al mismo efecto.

Análogamente al caso de interacción de dos blancos, aumentando la frecuencia inicial del sistema, pero sin variar su margen de frecuencias ni la posición de los blancos, se deberían generar gráficas de error en distancia y amplitud en las que los cambios suceden "más rápido".

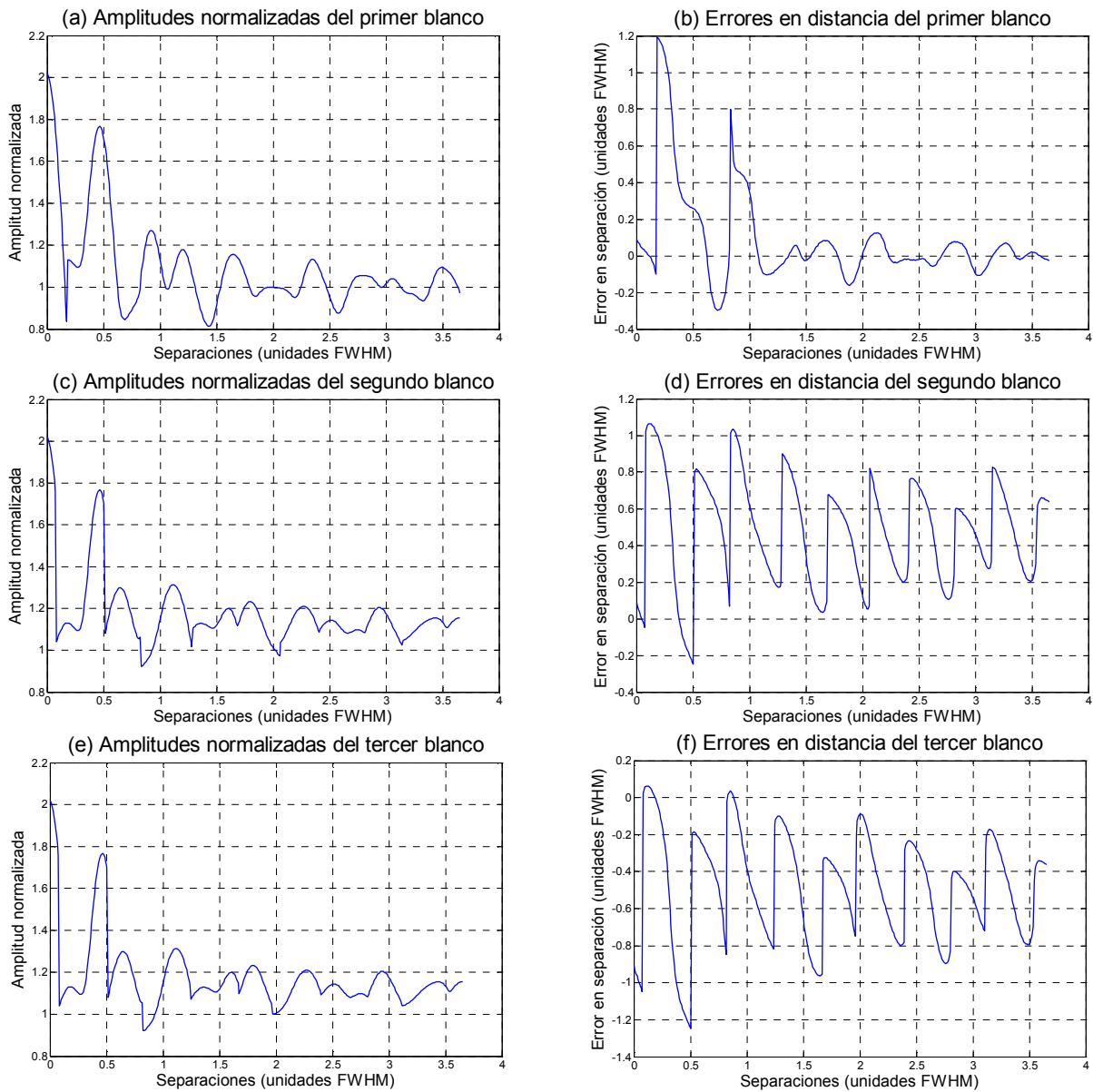


Figura 42: Errores en amplitud y distancia para simulaciones de tres blancos (50 a 80MHz).

En la Figura 42 se ha representado una simulación del error en distancia y en amplitud para el caso de tres blancos, con frecuencias que varían entre 50 y 80MHz. Se observa un comportamiento similar que para el caso de 10 a 40MHz, pero con cambios que se suceden a menor distancia, haciendo que las curvas cambien más rápidamente.

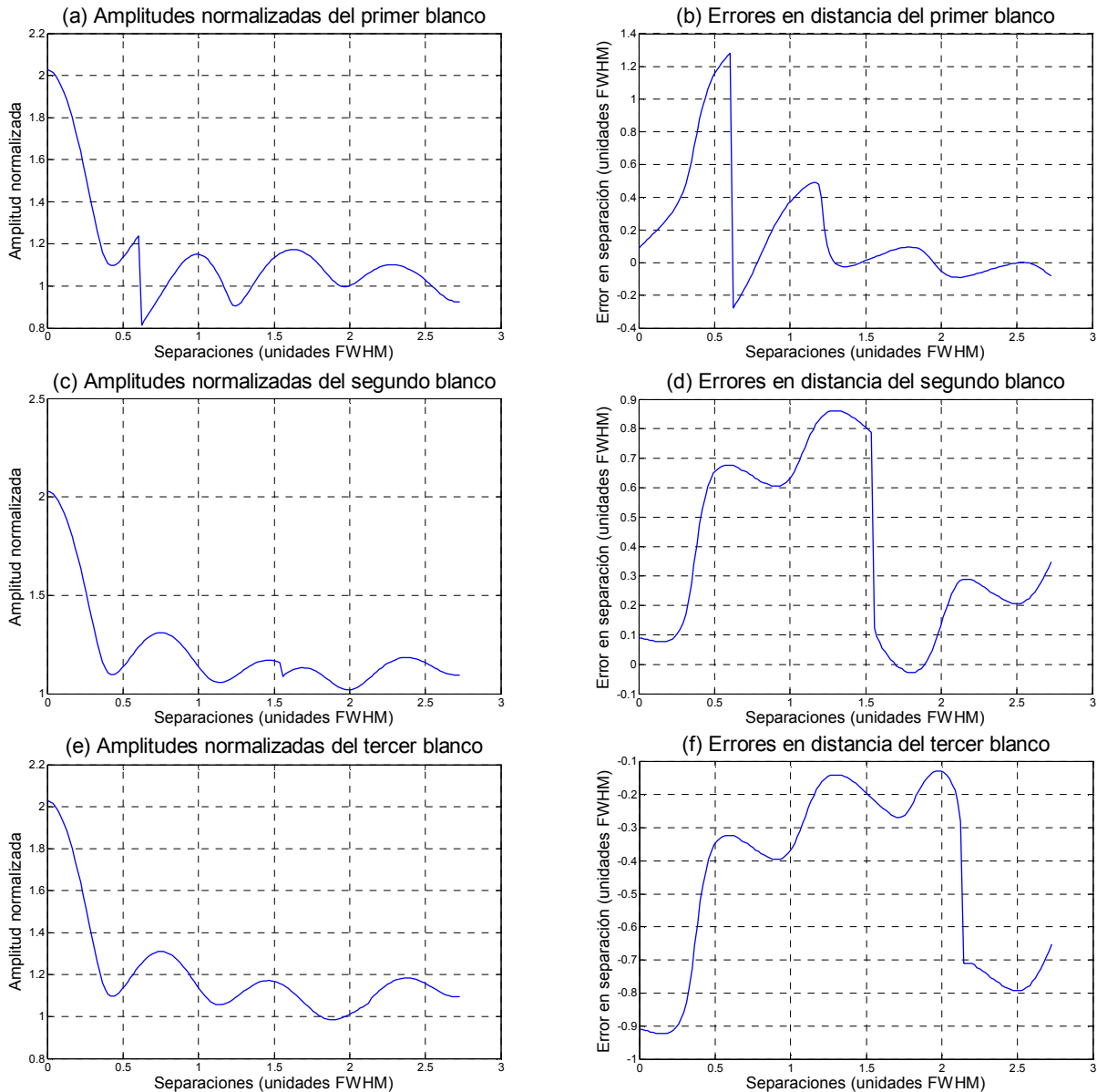


Figura 43: Errores en amplitud y distancia para simulaciones de tres blancos (20 a 80MHz).

En la Figura 43 se han representado las gráficas de error en amplitud y distancia para tres blancos, pero esta vez con frecuencias que van de 20 a 80MHz. De nuevo se observa que estas gráficas son las mismas que para el caso de frecuencias de 10 a 40MHz, pero teniendo en cuenta que ahora el segundo y tercer blanco están separados 2,5m, que es el equivalente para este margen de frecuencias a una unidad FWHM.

### 3.5.2 Tres blancos equidistantes

El programa de simulación propuesto para este caso funciona de forma similar al desarrollado para estudiar la interacción de dos blancos, pero en este caso además de simular un blanco que se acerca al origen, se simula otro que se aleja, alejándose los dos de un blanco fijo en el centro. En este caso se ha simulado el caso en que los blancos parten de una distancia al origen de 50m. Todo el procesado es análogo al del apartado anterior, pudiéndose detectar también hasta tres blancos. Para el caso de

frecuencias que van de 10 a 40MHz, la gráfica que representa la posición de los blancos detectados obtenida por esta simulación es la siguiente:

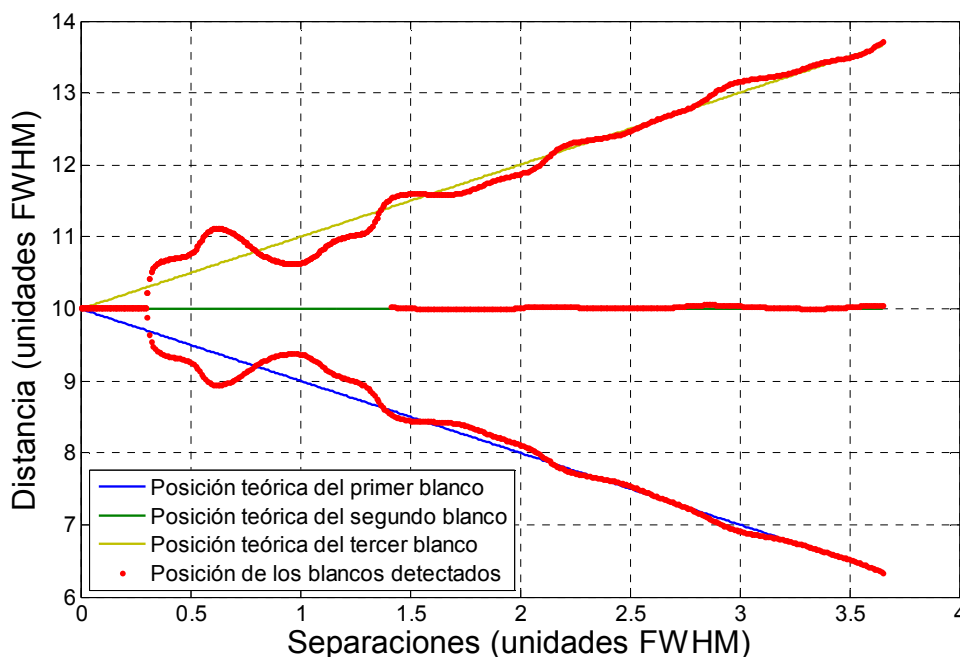


Figura 44: Simulación de la posición teórica y detectada de tres blancos equidistantes (10 a 40 MHz)

En la Figura 44 se han simulado tres blancos equidistantes cuya separación aumenta paulatinamente. En el eje de abscisas está representada la separación del primer blanco respecto del segundo (que es igual a la separación del tercer blanco respecto del segundo). Se observa que para separaciones pequeñas, se detecta sólo un máximo (como en el caso de dos blancos), ya que a separaciones muy pequeñas los tres blancos se comportan como uno sólo. En el intervalo  $[0.3, 1.4]$  se detectan dos máximos que, en un principio se detectan más separados que el primero y el tercer blanco, para después evolucionar a posiciones intermedias entre los tres blancos. A partir de la separación  $1.4FWHM$  estos dos máximos dan paso a tres, los cuales oscilan ligeramente alrededor de su posición teórica, a la que tienden asintóticamente. Es importante notar que la gráfica de la Figura 44 es prácticamente simétrica respecto del segundo blanco. Este hecho es debido a que los blancos primero y tercero se encuentran en todo momento en posiciones simétricas respecto del segundo blanco.

Como en el caso de tres blancos del apartado anterior, se puede representar el error de detección asociado a cada blanco. En este caso, las gráficas para el primero y tercer blanco serán simétricas, tal y como pasaba en el caso de dos blancos:

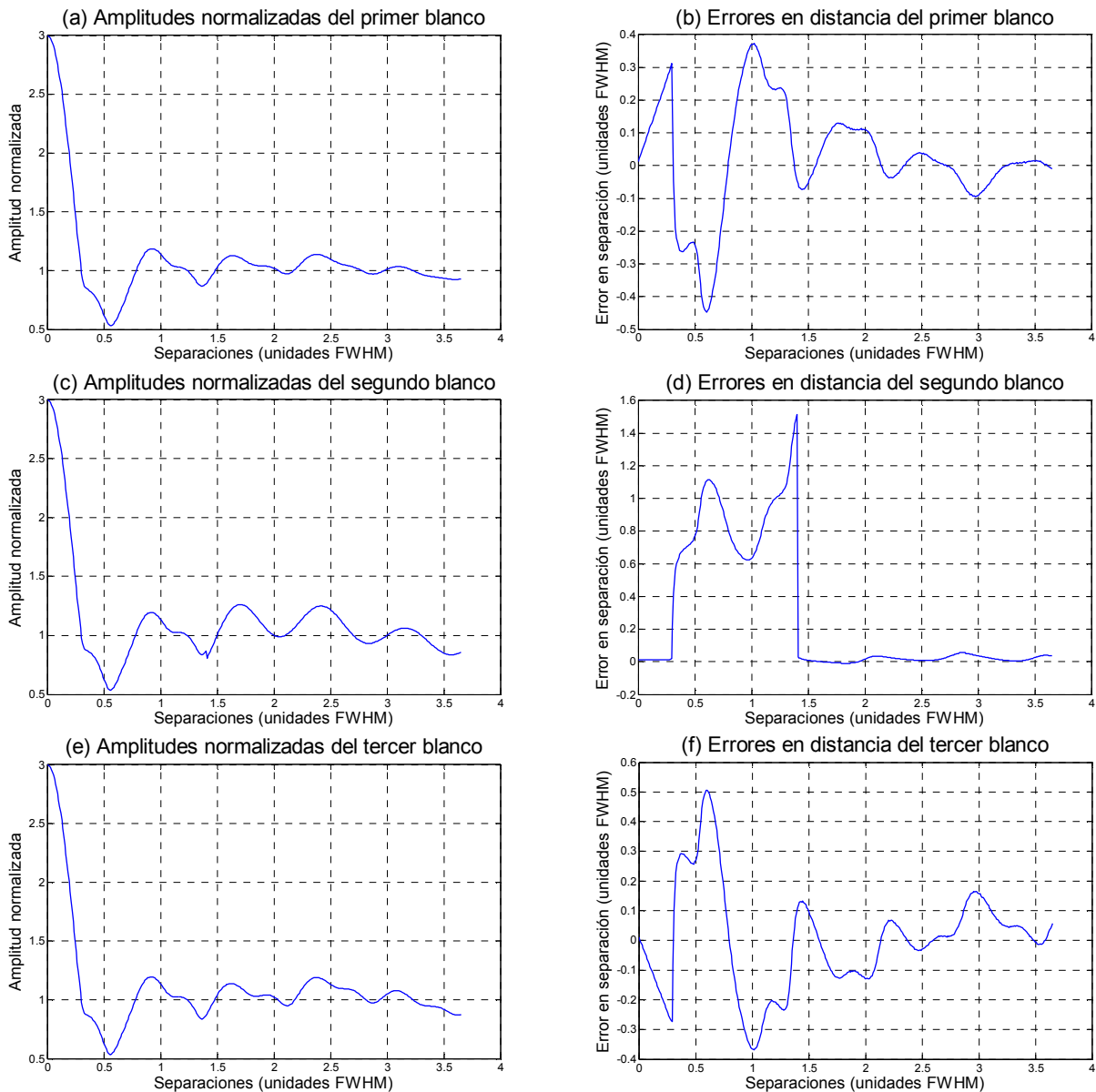


Figura 45: Errores en amplitud y distancia para simulaciones de tres blancos equidistantes (10 a 50MHz).

En la Figura 45 están representadas las curvas de error en amplitud y distancia para cada blanco en el caso de tres blancos equidistantes, para un rango de frecuencias de 10 a 40MHz. Se observa en las gráficas de amplitud normalizada (figuras (a), (c) y (e)) que las curvas de los tres blancos parten de un valor 3. Este efecto se debe a que en la posición inicial, los tres blancos se encuentran en la misma posición, y por lo tanto sus potencias se suman, dando lugar a la detección de un máximo de amplitud el triple que un solo blanco. En el intervalo [0.3,1.4] de la gráfica correspondiente al error en distancia del segundo blanco (d), correspondiente a la parte en la que se detectan dos máximos, se ha dibujado tomando como posición del blanco detectado el máximo más cercano al tercer blanco (el más alejado del origen). Ya que estos dos máximos se encuentran a la misma distancia de la posición teórica del segundo blanco, se podría haber elegido el otro, con lo cual los errores en distancia hubiesen sido negativos para este intervalo, pero con el mismo valor absoluto. Como en el caso de dos blancos, las curvas de las gráficas de error en amplitud (figuras (a), (c) y (e)) tienden asintóticamente a 1, y las de las gráficas de error en distancia (figuras (b), (d) y (f)) tienden a 0. Esta tendencia se debe a que los blancos cada vez se interfieren menos entre sí, por lo que se detecta cada uno prácticamente en solitario.



Como ya se ha hecho en los casos de dos y tres blancos anteriores, se pueden hacer estas simulaciones para valores mayores de la frecuencia inicial manteniendo el margen de frecuencias y la posición de los blancos, generándose gráficas de error en distancia y amplitud en las que los cambios suceden "más rápido". En la siguiente figura se han representado los errores de amplitud y distancia para simulaciones de tres blancos equidistantes, con frecuencias que van de 50 a 80MHz:

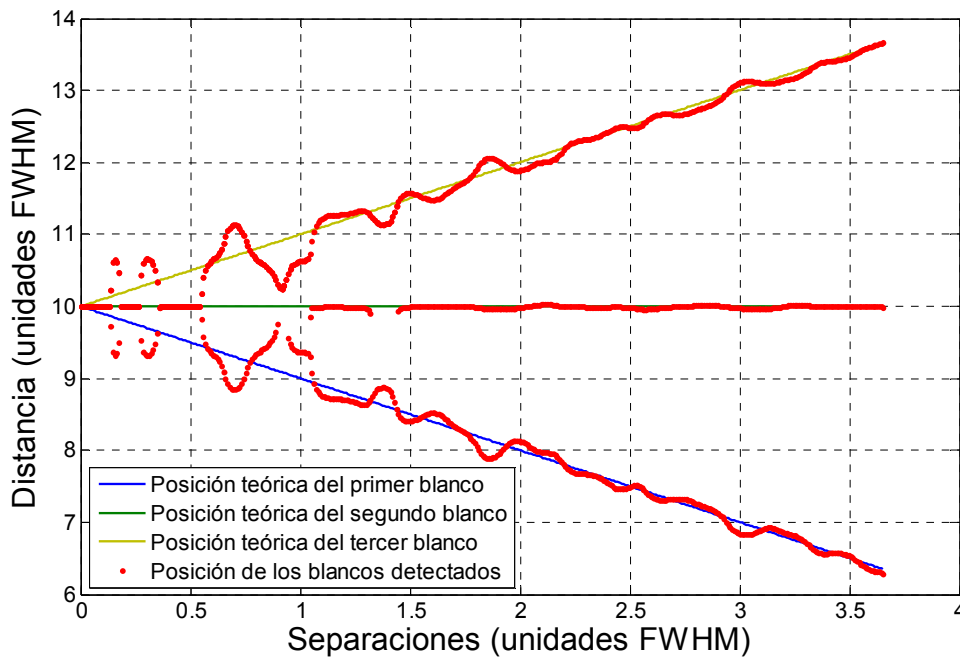


Figura 46: Simulación de la posición teórica y detectada de tres blancos equidistantes (50 a 80 MHz)

En la Figura 46 se observa un comportamiento parecido al ocurrido para los casos de dos y tres blancos: al aumentar la frecuencia inicial de la señal *sweep* las fases evolucionan más deprisa, dando lugar a cambios que suceden a menores separaciones. Como en el caso anterior, hay intervalos en los que se detecta un máximo ([0,0.14], [0.2,0.25] y [0.35,0.55]), intervalos en los que se detectan dos máximos ([0.14,0.2], [0.25,0.35], [0.55,1.05] y [1.3,1.45]), y para separaciones mayores se detectan tres blancos, que tienden asintóticamente a las posiciones teóricas de los blancos.

Tras ver la evolución de estos errores para dos y tres blancos, y a pesar que se ha hablado de la unidad FWHM como la unidad de resolución del sistema, es más acertado considerar que el error en la posición de un blanco sólo está realmente acotado en la mayoría de situaciones cuando no hay otro blanco comparable a una distancia menor de 1.5FWHM.



## CAPÍTULO 4: MONTAJE Y TÉCNICA EXPERIMENTAL

Aunque las simulaciones describen de manera muy precisa el comportamiento de un sistema FMCW, es necesario comparar estos resultados con un modelo de laboratorio. La diferencia fundamental entre el modelo de simulación y el modelo de laboratorio es que en la realidad los sistemas son "imperfectos", y por lo tanto se producen efectos que no se han tenido en cuenta debidos al ruido, a fases que no saltan exactamente  $\pi$ , o incluso a desajustes o derivas de los componentes. Además, como veremos más adelante, la coherencia del propio láser hará que algunas medidas de laboratorio salgan más atenuadas de lo que deberían.

En este capítulo se describe una manera de hacer el montaje completo de un sistema FMCW en lidar.

### 4.1 Montaje general

En la Figura 47 está representado el esquema del montaje utilizado en el laboratorio:

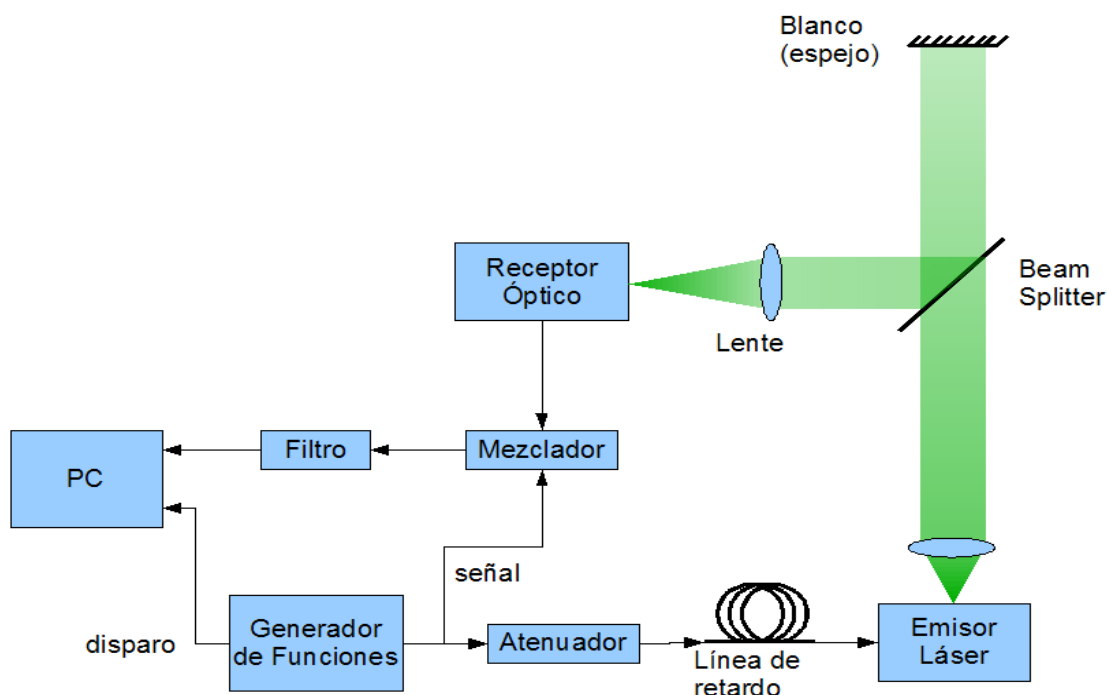


Figura 47: Montaje del Laboratorio

El Generador de Funciones es el encargado de generar la señal senoidal que aumenta en frecuencia a lo largo de un periodo (señal *sweep*). Se han utilizado los valores de la Tabla 2 para su configuración. La señal eléctrica de salida se divide en dos ramas, una para alimentar el emisor láser, y la otra se utiliza como señal de oscilador local en el Mezclador. La señal de entrada del emisor láser se debe atenuar para no saturarlo. En este caso se ha utilizado un atenuador de 5dB.

Magnitud	Valor
Tensión de pico	3,5V
Offset	0V
Frecuencia inicial	10, 20 ó 50 MHz
Frecuencia final	40 ó 80 MHz
Periodo	1ms

Tabla 2: Valores de inicialización del Generador de Funciones

Antes de aplicar esta señal al emisor láser, se retarda mediante un cable de unos 50m. Este elemento no formaba parte del montaje inicial. Fue necesaria su incorporación tras comprobar en las primeras medidas que el efecto provocado por la influencia de la parte frecuencial negativa, distorsionaba cualitativamente los resultados obtenidos. Añadiendo esta línea de retardo eléctrico al sistema, virtualmente se alejan los blancos detectados, y por lo tanto se reduce también la interferencia de la parte negativa.

El emisor láser genera la señal luminosa a partir de la señal eléctrica de entrada. La corriente de entrada excita un diodo láser que emite una señal luminosa de potencia proporcional a esta corriente. La longitud de onda del láser también es modulada ligeramente por esta corriente pero, para este caso, podemos ignorar la modulación de la longitud de onda, ya que no afecta al resultado final del montaje. Así pues obtenemos que la señal *sweep* eléctrica se traslada a la envolvente de la señal luminosa de salida. Justo a la salida del emisor láser se ha colocado una lente que colima el haz luminoso.

El haz luminoso proveniente del emisor láser atraviesa un *beam splitter* formado por un cristal inclinado  $45^\circ$  en la dirección horizontal respecto de la señal luminosa. Aproximadamente el 15% de la potencia de la señal luminosa se reflejará, saliendo perpendicularmente del sistema. Esta señal no se utilizará, pero, como veremos más adelante, será uno de los puntos a optimizar para minimizar la potencia de background recibida por el sistema. La mayor parte de la potencia (un 85%) atravesará el *beam splitter*. Esta será nuestra señal útil, la que enfocaremos hacia el blanco que queramos detectar.

Después de rebotar en los blancos, la señal luminosa devuelta volverá a incidir en el *beam splitter*. De nuevo el 15% de la potencia de la señal se reflejará y el 85% seguirá, pero en este caso la señal útil será la que se refleje, que saldrá con la dirección adecuada para ser captada por el detector óptico.

La entrada del receptor óptico es un orificio de unos 0.8mm de diámetro, por lo que es necesario concentrar la señal luminosa para captar la máxima potencia posible. Se ha utilizado una lente biconvexa para concentrar el haz de luz en el foco de ésta, donde colocaremos la entrada del receptor óptico. El receptor óptico está provisto de un fotodetector que genera una corriente proporcional a la potencia (o la envolvente) de la señal luminosa detecta. Esta corriente se transforma en tensión a través de un amplificador de transimpedancia, generando finalmente la señal de salida de este bloque.

La señal a la salida del receptor óptico se aplica a una de las entradas del Mezclador. En la otra entrada se aplica la señal de salida del Generador de Funciones. Mediante un Filtro paso-bajo se elimina a componente suma del producto frecuencial. De hecho

este filtro se podría obviar del montaje, ya que la tarjeta de adquisición del PC del siguiente bloque filtra la alta frecuencia.

El último bloque del sistema es un PC, que adquiere la señal a la salida del filtro. Esta señal se *sincroniza* a partir de la señal de disparo proporcionada por el Generador de Funciones. El PC es el encargado también de sumar cada uno de los periodos resultantes del producto.

#### 4.1.1 Programa en LabView del laboratorio

El programa en LabView que se ha usado en el laboratorio es el encargado de adquirir la señal a la salida del filtro, promediar los periodos requeridos, y guardar estos datos en un fichero.

El PC del laboratorio está provisto de una tarjeta de adquisición. A dos de las entradas de esta tarjeta se conectan la señal a la salida del filtro y la señal de disparo proveniente del Generador de Funciones. El programa en LabView que se ha desarrollado adquiere muestras de estos dos canales. Cuando se detecta un flanco en el canal correspondiente al disparo, se guardan en memoria el número de muestras correspondientes a un periodo de la señal de entrada.

Este periodo adquirido se sumará con los siguientes periodos, hasta promediar el número de periodos indicado por el usuario. Estos datos se guardan en un fichero que posteriormente podrá ser leído por otro programa que analice los resultados.

El programa, además, muestra los datos capturados por pantalla, así como la FFT de los mismos. Esta función será muy útil para apuntar correctamente los elementos ópticos, ya que podremos visualizar su respuesta en tiempo real, y así optimizarla.

#### 4.1.2 Programa de visualización de los datos

En el laboratorio se adquiere la señal eléctrica a la salida del mezclador y se guarda en un fichero. Estos datos se pueden analizar después mediante otros programas que además traten digitalmente la señal para extraer su información.

El programa más sencillo que se ha implementado es el utilizado para la señal adquirida en el laboratorio, ya sea en tiempo o en frecuencia. Este programa lee los datos de un fichero de texto que contiene el promedio de  $n$  periodos de la señal a la salida del filtro. Estos datos son guardados en memoria, y posteriormente se les hace la FFT. El resultado final de todo el montaje del laboratorio es la representación del módulo de esta transformada, en la cual están representados los blancos detectados mediante *sincs* (tal y como predice la teoría).

Opcionalmente, el programa puede leer los datos de dos medidas diferentes y restarlas. Esta operación resultará útil para estudiar la respuesta de blancos muy atenuados (blanco disperso), ya que a la medida del blanco se le restará la medida de la sala en ausencia del blanco (background) y así facilitar su estudio (ver apartado 5.4.2).

## 4.2 Procedimientos de Laboratorio

En este apartado se describen diversos procedimientos utilizados en el laboratorio. Según el tipo de medida que se desee realizar, será necesario utilizar unos procedimientos o otros, pero igualmente los procedimientos que se refieren a la alineación del sistema deberán efectuarse siempre.

### 4.2.1 Alineación del sistema

El primer paso a dar en el montaje del laboratorio es la alineación correcta de los elementos ópticos del sistema. Esta alineación es uno de los procesos más complicados y delicados del montaje del laboratorio, debido a que cualquier mínimo desajuste (debido a golpes accidentales, o el simple asentamiento de componentes) hace que el sistema deje de ser efectivo. La alineación de los componentes ópticos, además, se hace complicada debido a que el láser utiliza una longitud de onda infrarroja, y que por lo tanto no es visible. En el laboratorio se ha usado un visor de infrarrojos para poder ver la luz emitida por el láser, pero igualmente ha sido complicado ver exactamente dónde incide la luz, ya que el visor oscurece la visión de todo lo demás. Además de esto, la luz que reflejan los blancos, en especial los cristales, es muy tenue, y por lo tanto difícil de detectar con el visor.

En este apartado se explica los pasos que se han seguido para alinear el montaje del laboratorio para el caso de la detección de un blanco. En cada punto se describe como es físicamente cada elemento del montaje, además de cuales son sus grados de libertad y cual es su posición correcta. Los elementos ópticos se han alineado en el orden que sigue la señal luminosa.

#### 4.2.1.i Láser

El láser es una caja de unos 12x7x4cm que está montado sobre un soporte fijado a la mesa del laboratorio. Al chasis del láser se ha fijado una lente colimadora a unos 5cm de la salida de la señal luminosa. Esta lente concentra la luz proveniente del láser en un haz de rayos paralelos, con el objetivo de que la señal luminosa se expanda lo mínimo posible y así no perder potencia. Este haz luminoso ha enfocado hacia el blanco a detectar.

#### 4.2.1.ii Espejo (blanco)

Para nuestro sistema, un blanco es cualquier objeto que refleje la señal luminosa del láser hacia el receptor óptico. Con el fin de maximizar la potencia que refleja nuestro blanco, se ha utilizado un espejo de unos 18x13cm. El soporte utilizado para fijar este espejo permite un movimiento preciso en rotación y elevación. El blanco se ha colocado a una distancia de entre 1 y 9m del emisor láser, reflejando la señal luminosa de vuelta hacia este.

#### **4.2.1.iii Beam splitter**

La señal luminosa reflejada por el blanco vuelve al emisor láser, por lo que se debe desviar hacia el receptor óptico mediante un *beam splitter*. Se ha utilizado un cristal inclinado  $45^\circ$  respecto de la dirección emisor-blanco que está situado sobre un rail, y que permite ubicarlo a cualquier distancia en la dirección del haz luminoso proveniente del emisor láser con el fin de ajustar el reflejo a la posición del receptor óptico. Con esta configuración, aproximadamente el 85% de la potencia de la señal luminosa atraviesa el *beam splitter* en dirección a los blancos. Análogamente, cuando la luz vuelve tras reflejarse en los blancos, aproximadamente un 15% de su potencia es reflejada hacia la lente concentradora y el receptor óptico.

#### **4.2.1.iv Lente**

Antes del receptor óptico es necesario colocar una lente que concentre la señal luminosa. Esta lente está colocada sobre un soporte fijado en la mesa de trabajo. Es posible controlar la altura y ángulo de incidencia horizontal de la lente. Para su correcta alineación, se coloca perpendicularmente a la señal luminosa que viene del *beam splitter*, focalizando la mayor parte de esta luz en su foco, a unos 25cm.

#### **4.2.1.v Receptor óptico**

El receptor óptico también está montado sobre un soporte fijado a la mesa, y con libertad de movimientos en rotación y de elevación.

La entrada del receptor óptico debe estar colocada exactamente el punto donde se focaliza el haz luminoso, y así maximizar la potencia recibida de la señal luminosa reflejada en los blancos.

#### **4.2.1.vi Resto del montaje y afinado**

Toda esta alineación se debe afinar con el fin de maximizar la potencia del blanco detectado. Para hacer esto, se montan las restantes partes del sistema (parte eléctrica, que sólo necesita ser conectada, y la implementación del programa en LabView).

Si el sistema está correctamente alineado, se detecta una senoide de frecuencia proporcional a la distancia del blanco en la última etapa del montaje (el PC en nuestro caso, aunque también se podría utilizar por ejemplo un osciloscopio). Tomando por objetivo la maximización de la amplitud de esta senoide, se deben retocar todos los elementos ópticos, afinando su alineación, y así conseguir la máxima potencia posible.

Este proceso de alineación se ha tenido que repetir al principio de cada sesión de laboratorio, ya que, como los sistemas de fijación no son perfectos, pequeños desajustes en las posiciones de los elementos hace caer significativamente la potencia detectada en el sistema.

#### 4.2.1.vii Colocación de dos o más blancos

Para realizar el experimento con dos o más blancos, se han utilizado vidrios corrientes para modelar los blancos intermedios y se mantiene el espejo para el blanco más alejado del origen. Estos blancos intermedios se han alineado uno a uno empezando por el blanco más alejado. Para cada blanco, se tapan los demás más alejados para que no interfieran y se alinea correctamente. El proceso de alineación de los cristales es similar a la de los demás elementos ópticos (apuntado usando el visor, y afinado posterior para maximizar la amplitud de la sinusoide generada), pero con la particularidad de que los demás elementos ópticos no se pueden cambiar de posición, para asegurar que el sistema sigue alineado respecto a los demás blancos.

#### 4.2.2 Línea de retardo

Según el esquema general del método FMCW (como la del diagrama de bloques del apartado 2.2.3), para su implementación no es necesaria la incorporación de líneas de retardo. Si se miden blancos suficientemente alejados del cero (comparado con la anchura del lóbulo principal de la función *sinc* detectada), la interferencia generada por la parte frecuencial negativa no será importante. Pero si se miden blancos poco separados del sistema de medida, la contribución de la parte negativa del espectro puede ser apreciable, deformando cualitativamente la medida.

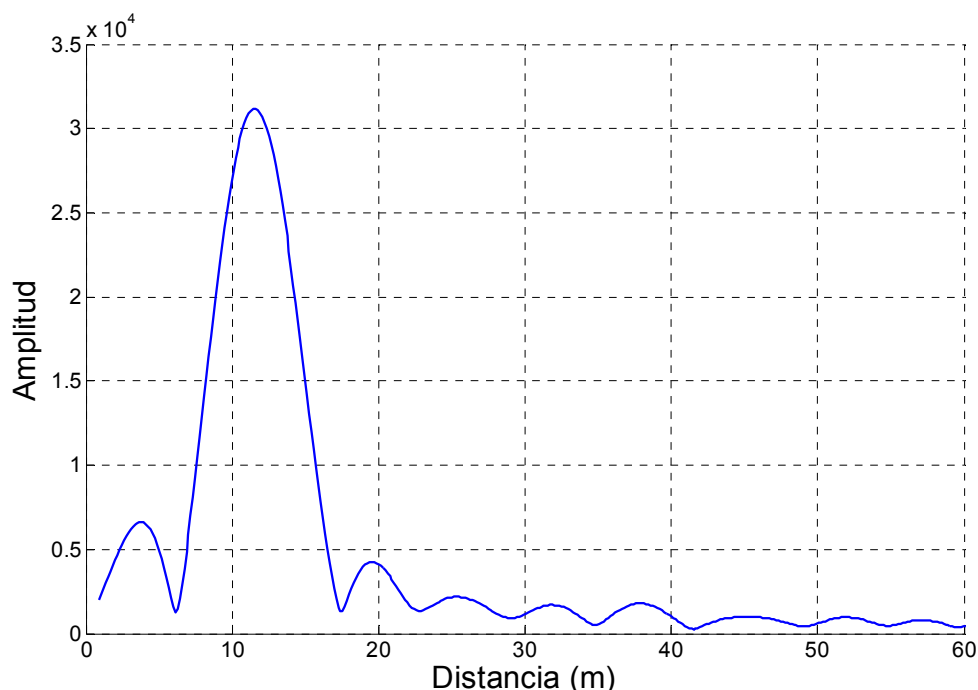


Figura 48: Medida de un blanco sin utilizar la línea de retardo (10 a 40MHz).

En la Figura 48 se observa cómo la cercanía del 0 (y por lo tanto de la parte frecuencial negativa) distorsiona la forma de la *sinc* detectada asociada al blanco medido. En medidas de más de un blanco, este efecto complica el análisis, ya que no es posible diferenciar entre la distorsión producida entre blancos, o la debida a la parte frecuencial negativa.



Ya que las dimensiones del laboratorio impiden la colocación de blancos suficientemente alejados del sistema para desprestigiar el efecto de la parte negativa, ha sido necesario añadir una línea de retardo en el sistema. Esta línea de retardo está compuesta por un cable de unos 50m, que retarda la señal de salida del Generador de Funciones antes de ser emitida por el emisor láser. Desde el punto de vista del receptor, los blancos se han alejado virtualmente, ya que la señal reflejada por éstos viene más retardada que en el caso sin línea de retardo.

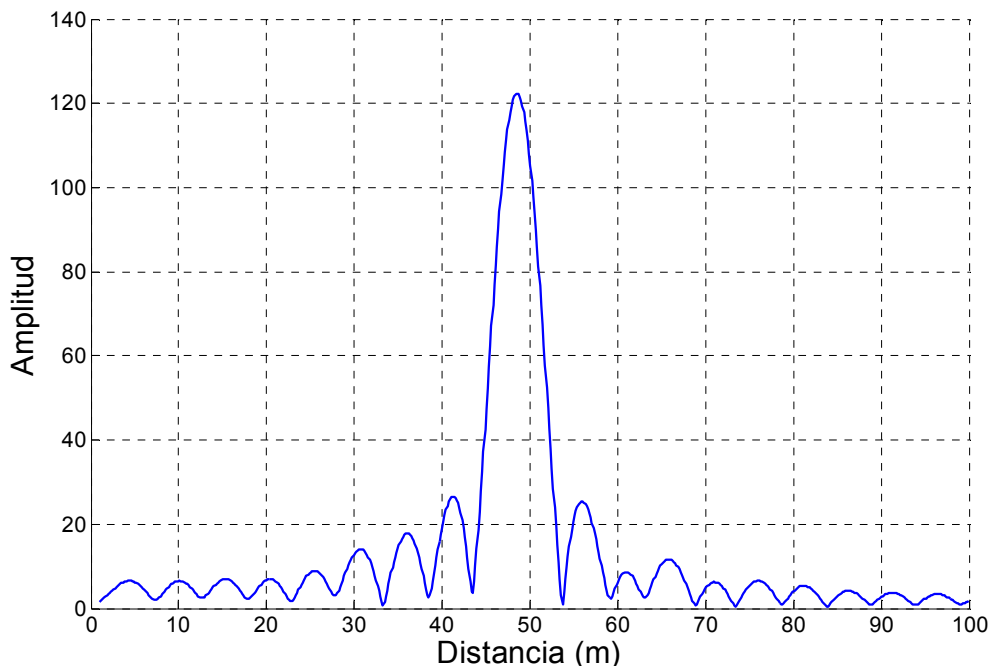


Figura 49: Medida de un blanco con línea de retardo (10 a 40MHz).

En la Figura 49 se puede ver el efecto de la línea de retardo en la medida de un blanco. Como se aprecia, el blanco se detecta a poco menos de 50m, cuando originalmente se detectaba a poco más de 10m. Comparándolo con el blanco detectado en la Figura 48, se observa que los lóbulos secundarios son mucho más simétricos, y aunque no llega a eliminarse la interferencia de la parte negativa del espectro, la gráfica resultante se asemeja lo suficiente a la función *sinc* esperada.

### 4.2.3 Enventanado

El generador de señales que genera la señal *sweep* presenta una cierta variación en la amplitud dependiendo de la frecuencia instantánea. Este efecto se hace visible en la representación temporal de la respuesta a un blanco. La frecuencia instantánea de cada punto de la respuesta temporal se calcula a partir de la resta de frecuencias emitida y recibida. Aunque esta resta se mantiene constante (para un blanco), la amplitud de la señal recibida disminuye a medida que aumenta la frecuencia de la señal *sweep*. Este comportamiento da lugar a un enventanado en la señal de respuesta.

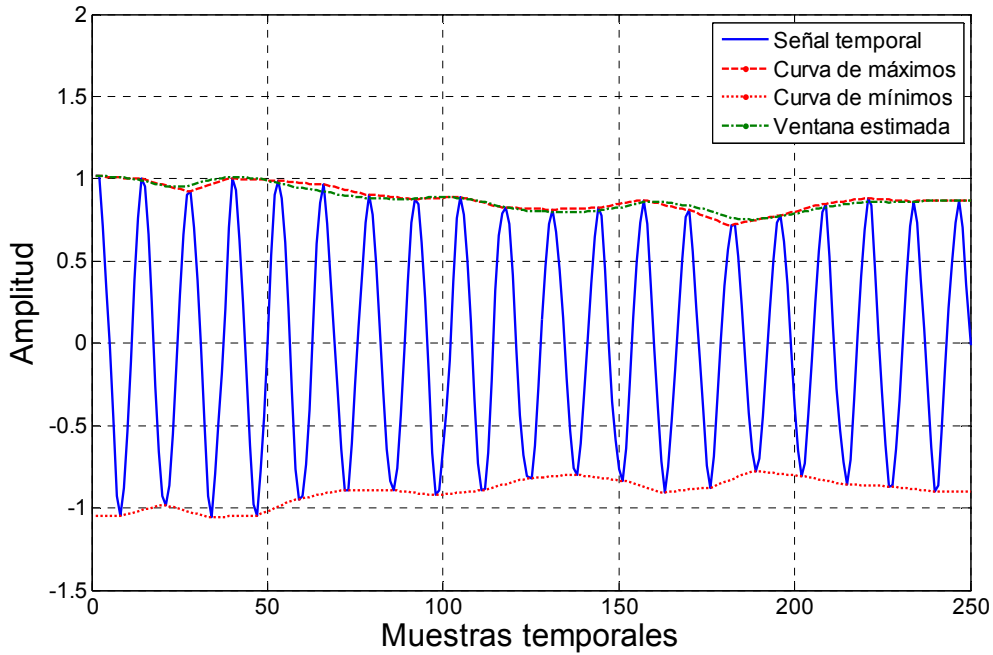


Figura 50: Representación de la respuesta temporal a un blanco (20 a 80MHz). Sobre la gráfica están representadas la curva que pasa por todos los máximos, la que pasa por todos los mínimos y la ventana estimada.

Con el fin de reducir este efecto, se propone una estimación de este enventanado a partir de la media de las curvas que recorren todos los máximos y todos los mínimos de la señal temporal, tal y como se observa en la Figura 50. Aplicando la ventana inversa a la señal anterior, se obtiene:

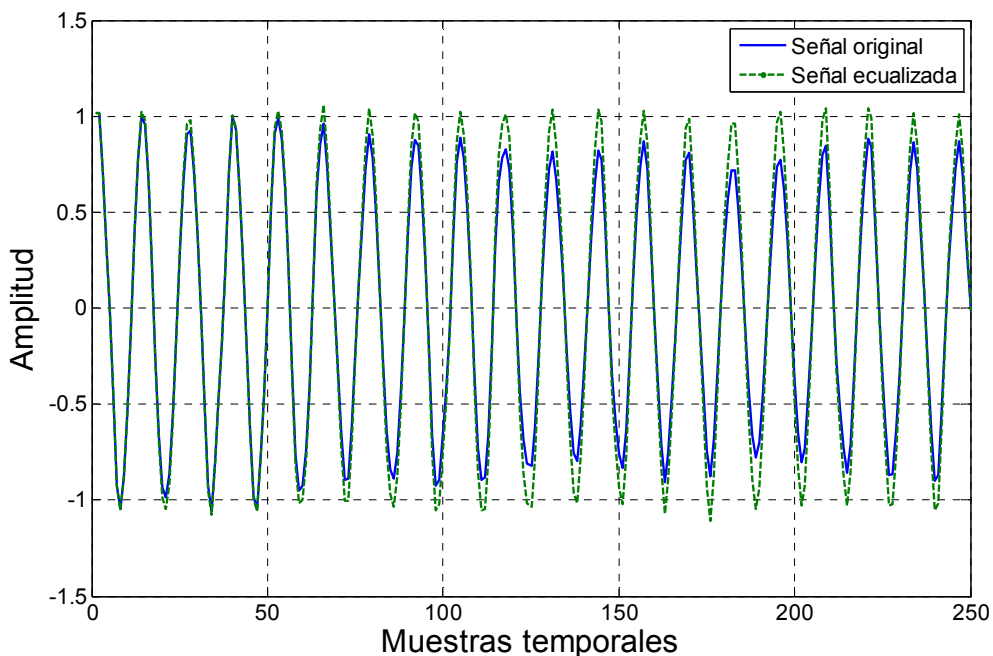


Figura 51: Respuesta temporal a un blanco, original y ecualizada (20 a 80MHz).

Se observa en la Figura 51 que la señal ecualizada tiene prácticamente amplitud constante, y por lo tanto se ha reducido el efecto paso-bajo del sistema completo. De todas maneras, como se observa en la Figura 52, este enventanado apenas modifica la forma de la *sinc* resultante, corrigiendo mínimamente algunos los lóbulos secundarios.

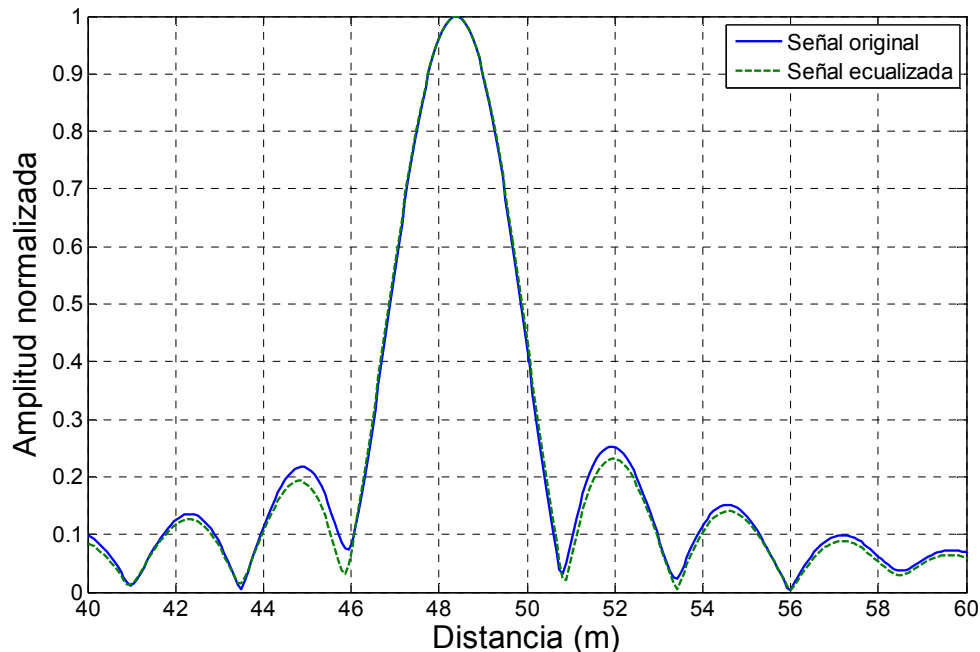


Figura 52: Respuesta a un blanco, original y ecualizada (20 a 80MHz).

Aunque el efecto de enventanado es evidente en la representación temporal de la señal, los cambios inducidos en su transformada son prácticamente despreciables, sobre todo en el lóbulo principal de la función *sinc*. Durante el resto del proyecto, no se tendrá en cuenta el efecto de enventanado, suponiendo que la respuesta del sistema completo es totalmente plana.

#### 4.2.4 Igualación de potencias

En las simulaciones, a no ser que se diga lo contrario, se ha tenido en cuenta que cuando se mide más de un blanco a la vez, cada uno por separado tiene la misma potencia. Para comparar correctamente las medidas de laboratorio con las simulaciones es necesario que las condiciones iniciales sean lo más parecidas posibles, para introducir el mínimo error de medición.

Una vez tenemos los blancos correctamente alineados, es posible que la potencia reflejada por cada uno sea diferente, y se hace necesaria su igualación. Para simplificar, supondremos que dos blancos reflejan la misma potencia en recepción cuando la amplitud del lóbulo principal de la transformada de su respuesta sea la misma. Se toma la amplitud del blanco que refleje menos potencia como base y se desalinean los demás hasta conseguir que la amplitud de cada uno por separado sea la del menos potente. Para desalinearse cada blanco se ha optado por modificar su elevación, y por lo tanto hacer que el espejo/cristal refleje en una dirección diferente de la dirección de máxima reflexión.

Como en el caso de la alineación de los blancos, la igualación de la potencia se debe de hacer por orden de lejanía, empezando por el más lejano (normalmente un espejo). Para cada blanco se tapan los blancos que están más lejos, y se desapuntan totalmente los blancos que están más cerca del sistema, anulando así todos los blancos de la medida menos el que se pretende igualar. En esta posición se ajusta la elevación

del blanco hasta igualar su amplitud con la de los demás. Notar que para anular los blancos más cercanos no se pueden quitar físicamente, ya que los cristales atenúan la señal luminosa, y por lo tanto la potencia recibida por el sistema de los blancos más lejanos no es la misma con o sin el blanco que tiene antes. Se ha asumido que la atenuación introducida por un cristal es la misma tanto si está perfectamente orientado como si se desalinea ligeramente, pero se han hecho medidas para varios casos, y para los ángulos de inclinación que se han usado en los blancos para anularlos, la atenuación es prácticamente la misma, por lo que la hipótesis es realista.

A lo largo de la elaboración de las medidas, se ha ido refinando el proceso de igualación de blancos, desarrollando métodos cada vez más eficientes y precisos. Básicamente, se han utilizado tres métodos:

#### ***4.2.4.i Igualación de la amplitud en tiempo***

El primer método utilizado para la igualación de potencias consistía en tratar de igualar la amplitud de la senoide generada por cada blanco. Este método es el menos preciso, ya que debido al inventanado y a otros efectos se hace difícil precisar la amplitud de la senoide de un blanco de manera objetiva, así como reproducirla en otros blancos. Además, esta igualación de potencias no se comprobaba hasta el análisis de los datos en MATLAB, por lo que en la mayoría de los casos, los blancos estaban mal igualados.

#### ***4.2.4.ii Varias medidas de amplitud en tiempo***

Ya que el método anterior de igualación de potencias no conseguía la calidad requerida en las medidas, se optó por variar ligeramente el método en otro más laborioso. Se hacen medidas de varias amplitudes de la señal en tiempo, y para cada blanco. Además, en esa posición se mide también la medida de los blancos conjuntos. Este método requiere mucho más tiempo que el anterior, ya que si se miden cuatro amplitudes diferentes para dos blancos, son necesarias un total de 16 medidas de los dos blancos conjuntos.

Por ejemplo para el caso de la Figura 53 tenemos medidas de dos blancos a varias potencias. Ahora sólo queda elegir una de las gráficas de cada blanco, tratando de que las potencias de ambos sean lo más parecidas posibles.

El número de medidas a realizar se puede reducir drásticamente si se cambia la potencia de sólo uno de los blancos, manteniendo la potencia del otro fijo. Aún así, es necesario hacer medidas que después se desecharán, por lo que este método sigue siendo muy ineficiente.

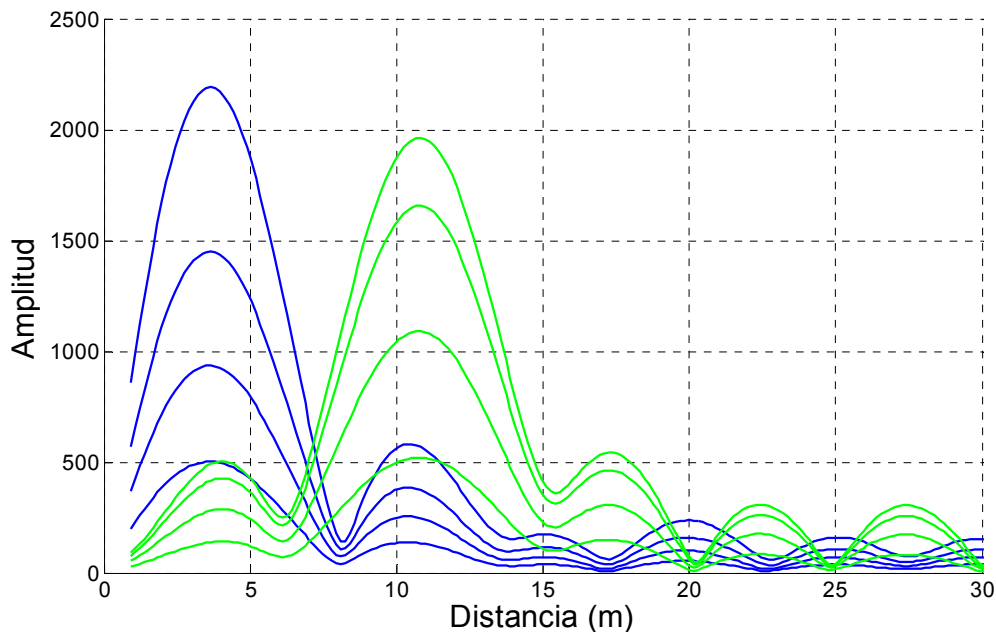


Figura 53: Medidas de varias potencias para cada blanco sin línea de retardo (10 a 40MHz).

#### 4.2.4.iii Igualación de la amplitud en frecuencia

El problema principal de los métodos anteriores es que las medidas se hacen "a ciegas". Como se explica anteriormente, se considerará que dos blancos se detectan a igual potencia cuando la amplitud de la función *sinc* resultante para cada uno sea la misma. En los métodos anteriores se trata de estimar esta amplitud a partir de la amplitud de las sinusoides en tiempo.

El tercer método propuesto consiste en implementar la Transformada de Fourier de la señal detectada directamente en el programa en LabView (aunque el programa siga guardando la señal en tiempo). De esta manera se hacen las medidas fijándose directamente en las amplitudes de las *sincs*, y por lo tanto consiguiendo potencias muy parecidas para cada blanco, y además haciendo sólo las medidas necesarias. En la Figura 54 está representada la medida de dos blancos debidamente igualados, utilizando este método.

Este es el método que se ha utilizado en la mayoría de las medidas de laboratorio. Se ha fijado como factor de calidad de las medidas que la amplitud de un blanco no debe ser un 10% superior a la de los otros.

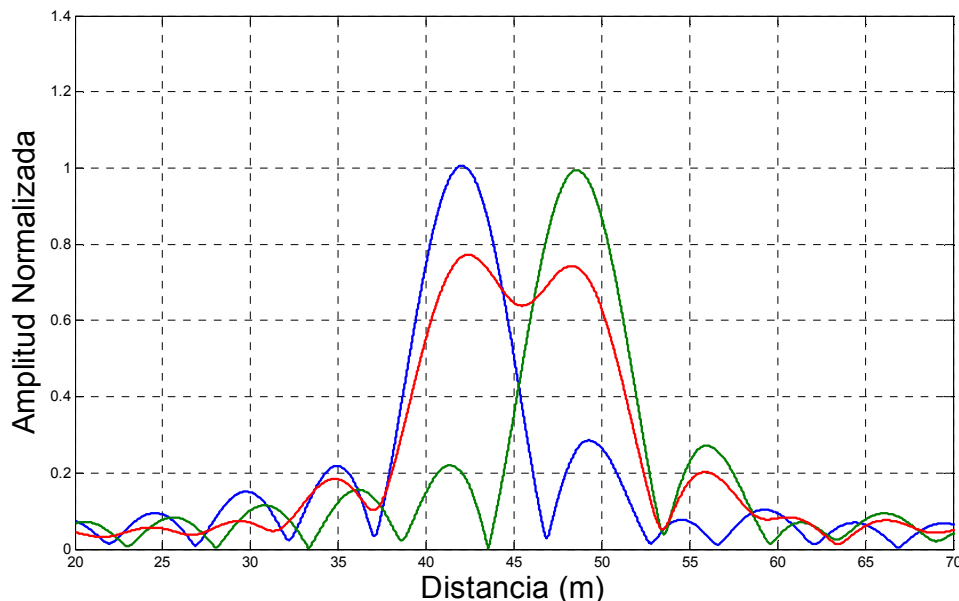


Figura 54: Medida de dos blancos a 6,5m con potencias debidamente igualadas (50 a 80MHz).

#### 4.2.5 Normalización

En general, las amplitudes de los blancos a medir no son importantes para este estudio, por lo que las gráficas de los resultados están normalizadas respecto de la amplitud del lóbulo principal para un blanco. Cuando se hacen medidas que involucran más de un blanco, es necesario que esta normalización se haga respecto la amplitud de los blancos aislados (que como vimos en el apartado anterior, se ha forzado que sea para todos la misma), ya que las interferencias entre blancos puede variar la amplitud de los máximos detectados.

Teniendo esto en cuenta, después de hacer una medida que incluya más de un blanco, es necesario obtener la medida del espectro recibido de cada blanco por separado, pero sin variar su potencia respecto de la medida conjunta. Además, siempre existirá un error a la hora de igualar la potencia de cada blanco, por lo que la medida de cada uno puede servir para comprobar que estamos dentro de un margen de error aceptable.

La mejor manera de hacer estas medidas hubiese sido hacer una medida de cada blanco quitando los demás, pero esto nos daría unos datos probablemente falsos, ya que después tendríamos que colocar de nuevo los blancos restantes exactamente igual que estaban, y esto no es posible. De forma similar a la igualación de blancos, estas medidas deben hacerse en un orden concreto.

Antes que nada, con el sistema alineado y las potencias correctamente igualadas, se hace la captura de los datos de todos los blancos. Después se mide cada blanco por separado en orden empezando por el más cercano al sistema. Para cada blanco, se tapan los blancos más lejanos que él y se desapuntan los más cercanos. De esta manera se hace la medida de el blanco aislado, pero teniendo en cuenta la atenuación de los cristales que hay entre él y el sistema.

## CAPÍTULO 5: RESULTADOS

Con los procedimientos descritos en el capítulo anterior se han hecho abundantes medidas tratando de ilustrar las posibles diferencias existentes entre un modelo real de FMCW y el modelo teórico o simulado. En este capítulo se muestran algunas de esas medidas, comentando las igualdades y diferencias entre las medidas de laboratorio y las que predicen la teoría o el modelo de simulación.

### 5.1 Medidas de un blanco

Comenzamos este capítulo con las medidas de un blanco. Estas medidas son las más sencillas que se han hecho, e ilustran el comportamiento general del montaje del laboratorio.

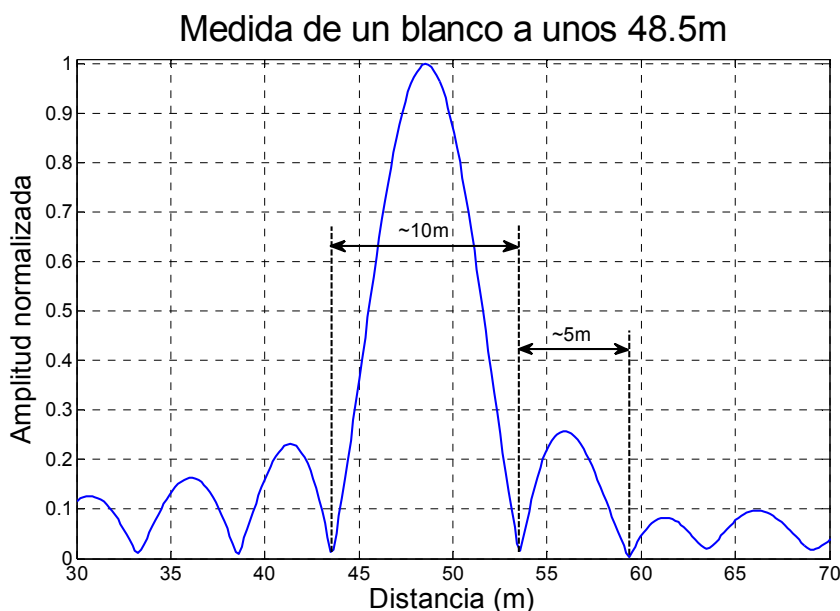


Figura 55: Medida de un blanco a unos 48.5m. 50 a 80MHz, 1ms, 100 periodos.

En la Figura 55 se observa una medida de laboratorio de un blanco a unos 48.5m, con frecuencias que van de 50 a 80MHz, un periodo de 1ms y un promediado de 100 periodos. Tal y como predice la teoría, la respuesta a un blanco es una *sinc* centrada en la posición detectada del blanco, con una anchura entre ceros del lóbulo principal de unos 10m, el doble de la distancia entre ceros de 5m (2.25). Esta anchura del lóbulo principal depende directamente del margen de frecuencias  $\Delta f_{dev}$  usado en la medida. Se han hecho medidas con otros valores de periodo  $T_m$  y número de periodos  $n$ , pero los cambios no son apreciables ya que la potencia de señal es mucho mayor a la del ruido. A lo largo de este capítulo se tomará  $T_m=1ms$  y  $n=100$  periodos a no ser que se diga lo contrario.

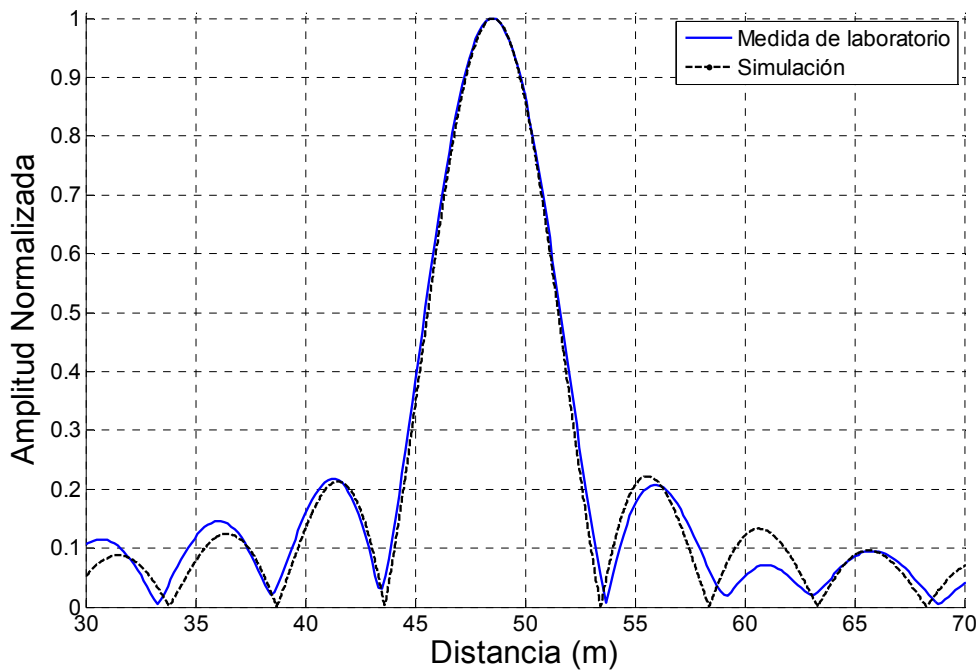


Figura 56: Medida de laboratorio y simulación de un blanco (10 a 40MHz)

Los las medidas realizadas para un blanco dan prácticamente el mismo resultado que las simulaciones, tal y como se aprecia en la Figura 56. Se observan básicamente dos pequeñas diferencias, los ceros de la *sinc* no son exactamente cero en las medidas; y la influencia de la parte negativa del espectro parece afectar de forma diferente. De todas maneras, estas diferencias prácticamente no se aprecian en el lóbulo principal de la *sinc*, y por lo tanto no afectan cualitativamente a la medida.

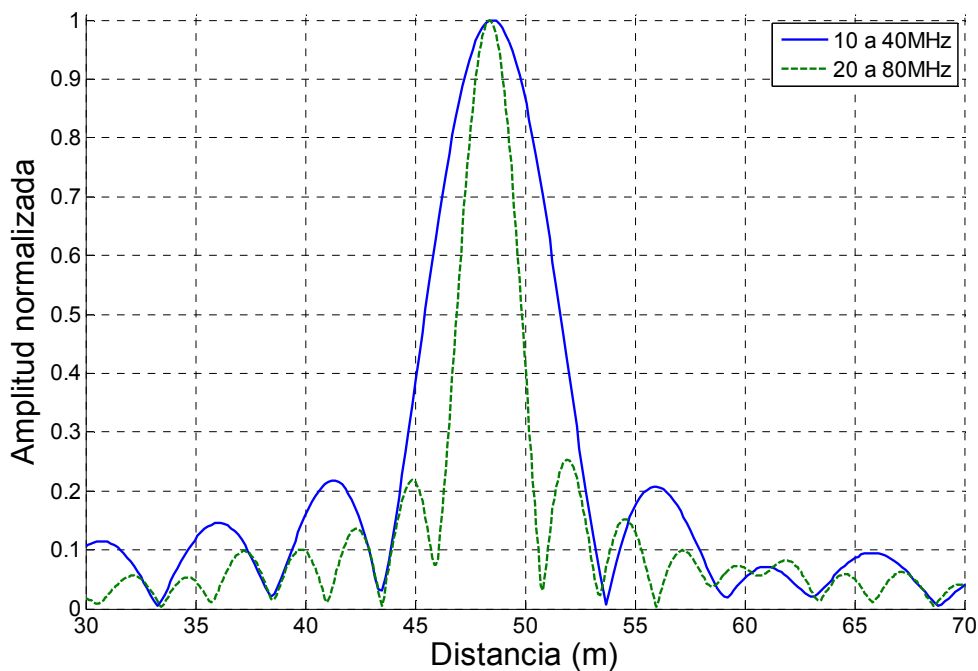


Figura 57: Medidas de un blanco con dos diferentes márgenes frecuenciales.



En la Figura 57 están representadas medidas de un blanco para dos anchos de banda diferentes. Se aprecia el efecto del margen de frecuencias  $\Delta f_{dev}$  en la anchura de los lóbulos de las *sincs* medidas, siendo la anchura del lóbulo principal de aproximadamente 10m para el caso de 10 a 40MHz, y de aproximadamente 5m para el caso de 20 a 80MHz.

## 5.2 Medidas de dos blancos

Se han hecho varios conjuntos de medidas de dos blancos separados diferentes distancias, con el fin de poder contrastar las simulaciones obtenidas en el apartado 3.3. Una medida de ellas puede ser la siguiente:

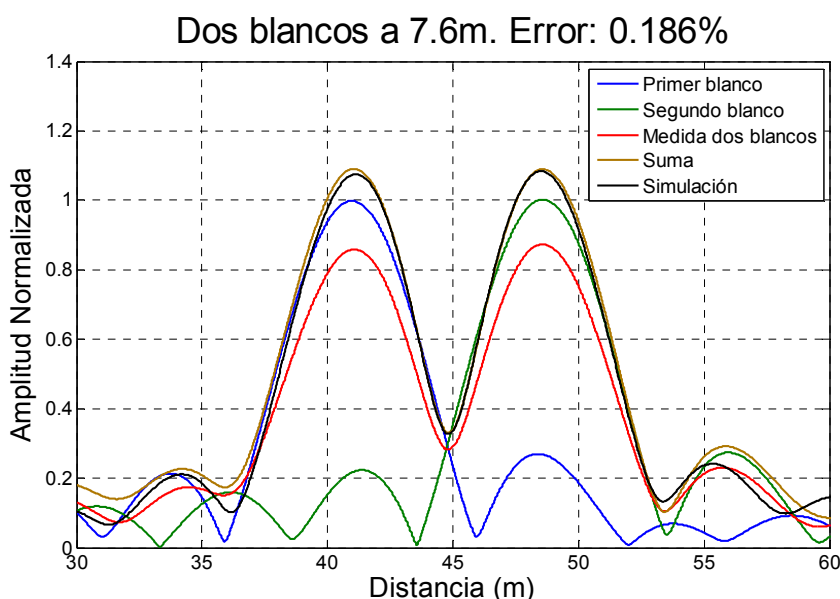


Figura 58: Medida de dos blancos separados unos 7,6m (50 a 80MHz). Se ha añadido en la gráfica las medidas de los blancos por separado, la suma de ambos y la simulación.

En la Figura 58 está representada la medida de dos blancos separados unos 7,6m. Además de la medida conjunta, se han añadido a la gráfica las medidas de cada blanco por separado, la suma de los dos blancos medidos por separado, y la simulación de dos blancos en las posiciones teóricas de los blancos. El error porcentual que aparece en el título de la gráfica corresponde al error cometido al igualar la potencia de los dos blancos por separado. Se observa que las gráficas suma y simulación prácticamente coinciden, sin embargo, la gráfica de la medida conjunta se encuentra atenuada, aunque mantiene la forma de onda.

Una de las primeras medidas de dos blancos que se hicieron durante la elaboración del proyecto está representada en la Figura 59.

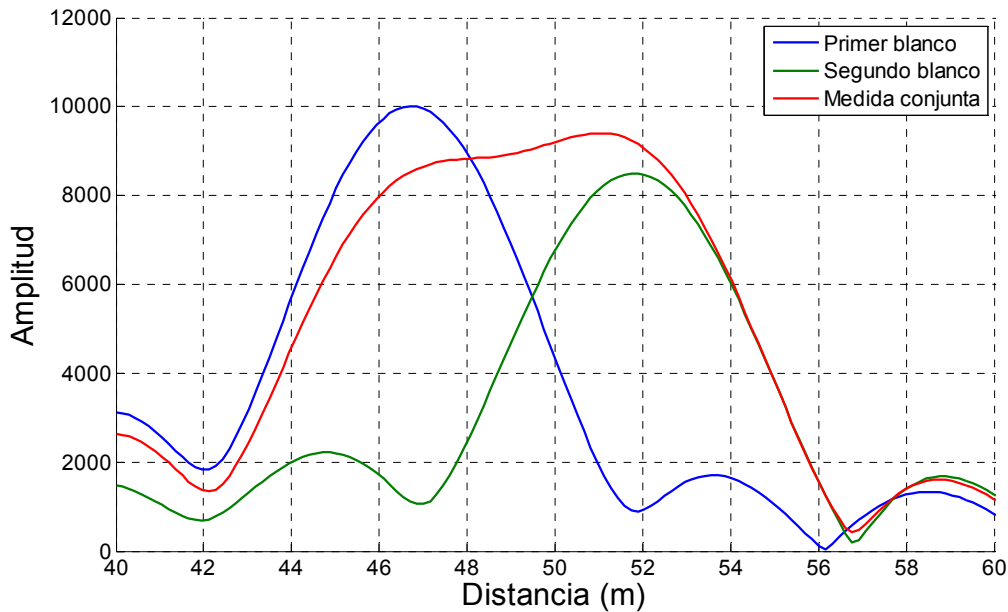


Figura 59: Medida de dos blancos a unos 5m (10 a 40MHz).

En la Figura 59 se observa la medida de dos blancos a unos 5m por separado y juntos. En un principio, creímos que la medida de los dos blancos juntos estaba mal, ya que como se aprecia en la gráfica, en el punto de mayor amplitud del primer blanco, al sumarse la contribución del segundo la medida conjunta sale inferior al primer blanco; por otro lado, en el punto de mayor amplitud del segundo blanco, al sumarse la contribución del primero la medida conjunta sale por encima del segundo blanco. Por su puesto la explicación de este efecto se encuentra en las fases:

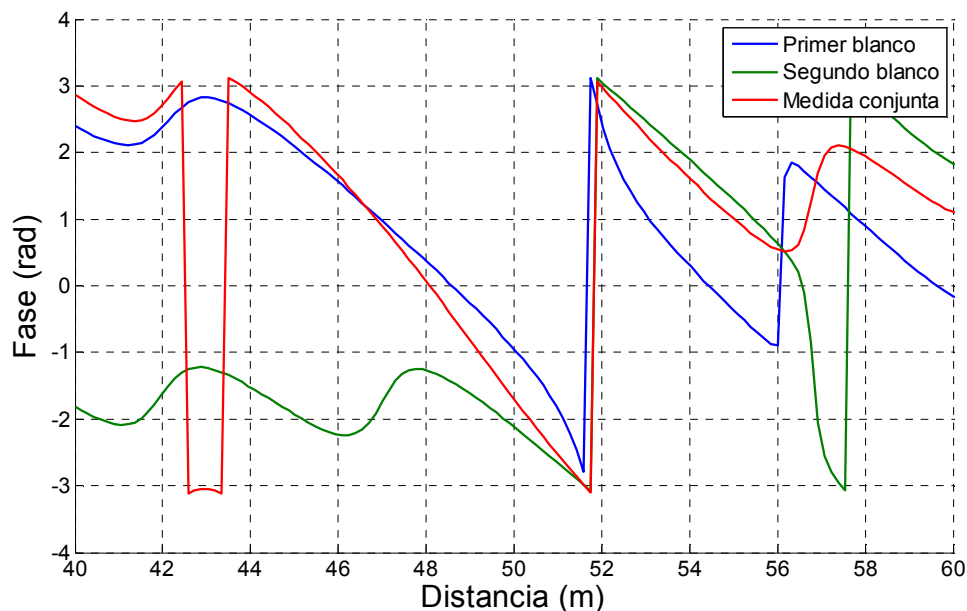


Figura 60: Fase de la medida de dos blancos a unos 5m (10 a 40MHz)

En efecto, en la Figura 60 se observa que la fase relativa entre los blancos en el punto de máxima amplitud del primer blanco está próxima a  $\pi$ , por lo que se crea una interferencia destructiva, resultando una amplitud en la medida conjunta menor a la amplitud del primer blanco. De igual manera, la fase relativa entre los blancos en el

punto de máxima amplitud del segundo blanco está próxima a  $\theta$ , por lo que en este caso la interferencia creada es constructiva, resultando una amplitud en la medida conjunta mayor a la amplitud del primer blanco.

Entender correctamente esta medida fue de gran ayuda para poder entender muchas otras, y poder ver en qué casos la medida estaba bien hecha aunque no entendiésemos el resultado.

Ya que la forma de la medida de dos blancos cambia drásticamente en función de su separación (debido a la fase relativa entre ambos) se probaron varias maneras de representar dos blancos separados varias distancias en una misma gráfica:

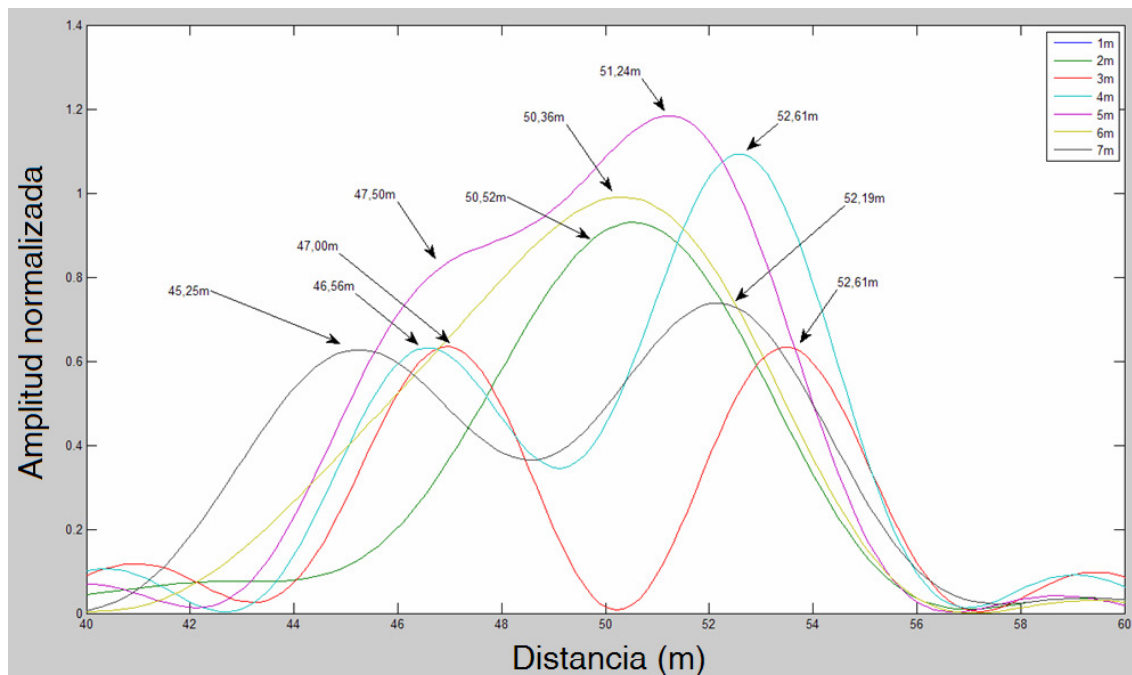


Figura 61: Varias medidas de dos blancos a diferentes separaciones (10 a 40MHz).

En la Figura 61 están representadas varias medidas de dos blancos a diferentes separaciones, señalando en cada una la posición detectada de cada máximo. Aún facilitando estas posiciones de los máximos, se hace difícil analizar el error cometido con cada medida a simple vista. Además, si se intentasen representar más medidas a otras separaciones diferentes, la gráfica se volvería absolutamente ininteligible.

Como en el caso de las simulaciones, se pueden detectar automáticamente los máximos de las medidas que puedan asociarse a blancos detectados y, por lo tanto, representar su amplitud y su error de posición. Se han hecho una serie de medidas en diferentes puntos de las gráficas simuladas de error de potencia y distancia vistas en el apartado 3.4.1 con el fin de comprobar que las medidas de laboratorio se ajustan a las simulaciones. Se ha procurado que las medidas se hiciesen a separaciones tales que los máximos a detectar fuesen muy abruptos, ya que un máximo muy plano podría detectarse en un rango relativamente amplio debido al ruido.

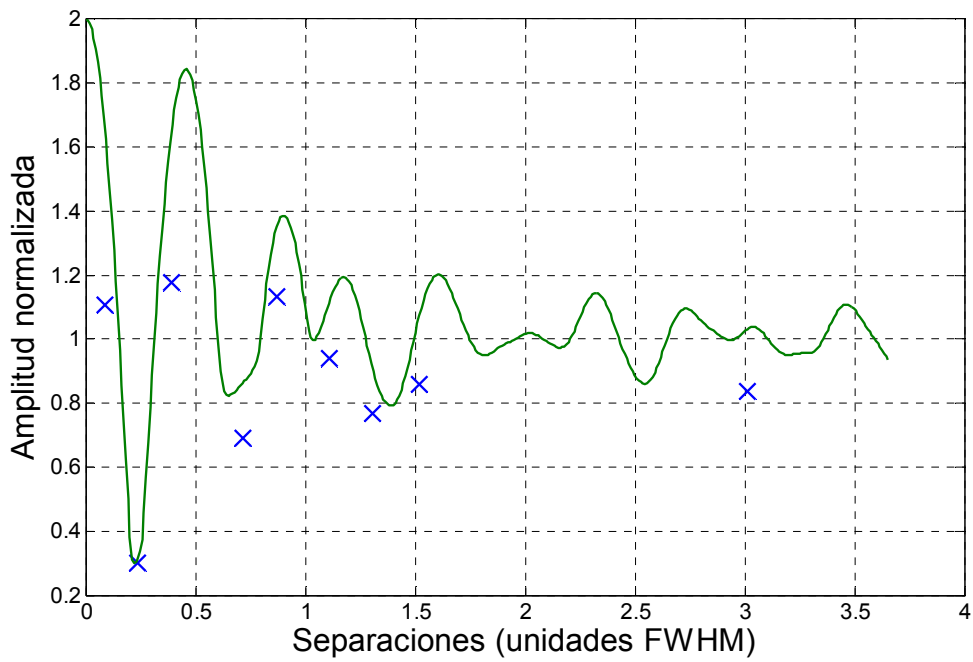


Figura 62: Error de amplitud del primer blanco en medidas de dos blancos (50 a 80MHz).

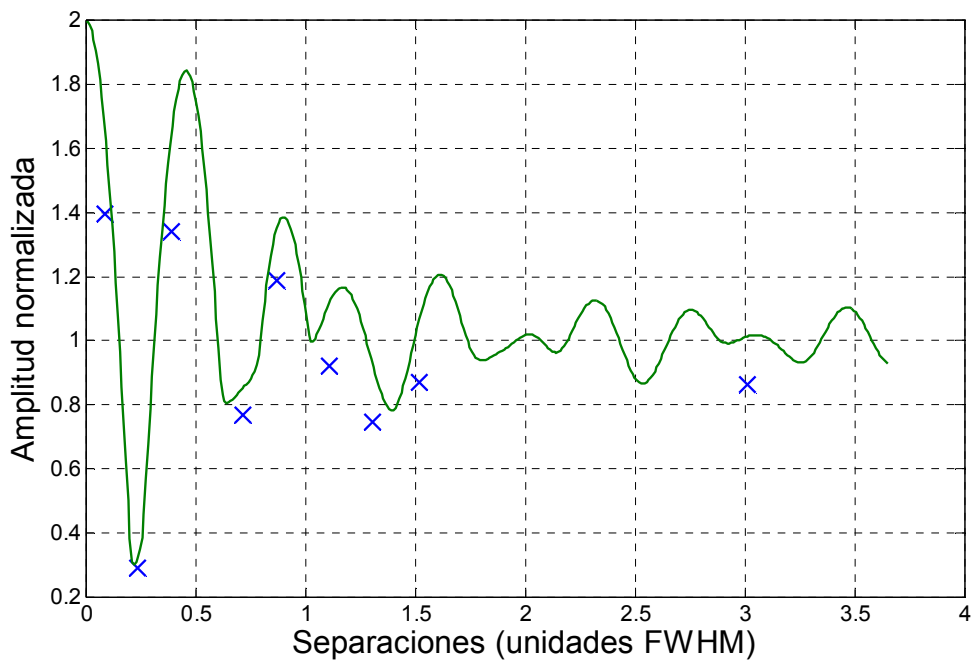


Figura 63: Error de amplitud del segundo blanco en medidas de dos blancos (50 a 80MHz).

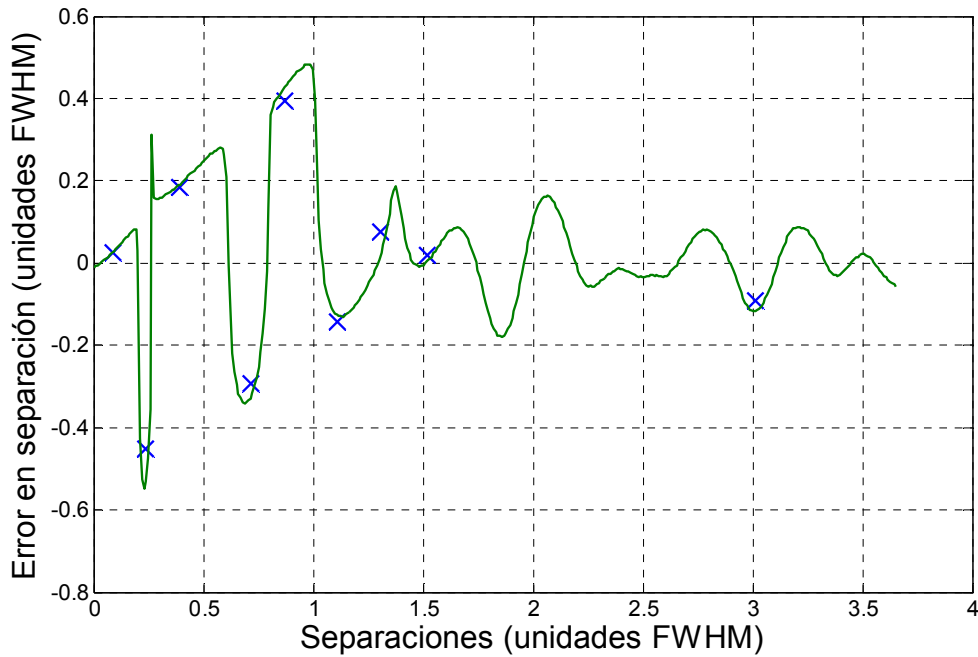


Figura 64: Error de posición del primer blanco en medidas de dos blancos (50 a 80MHz).

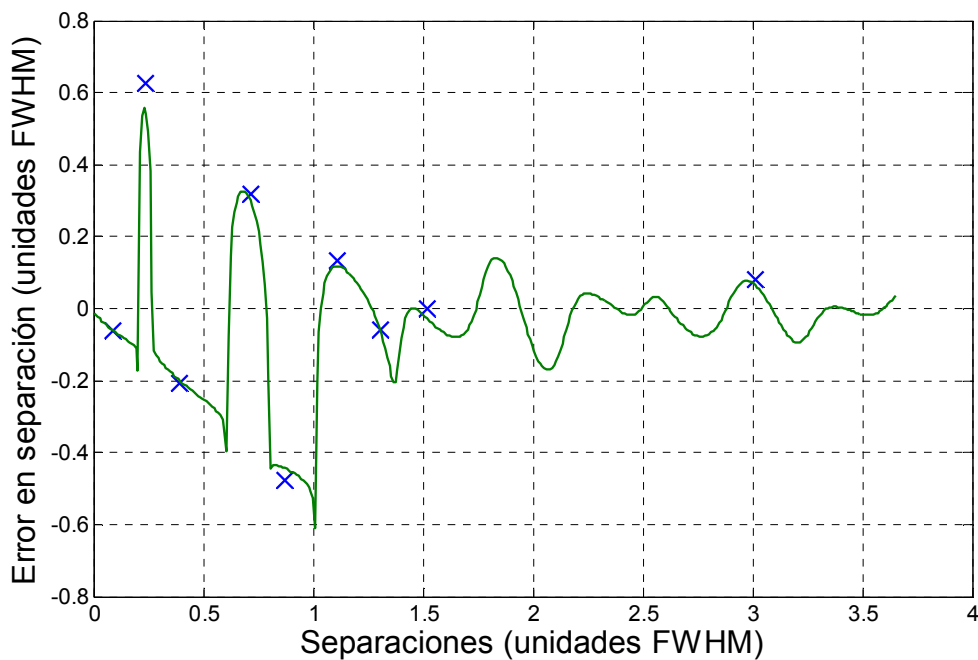


Figura 65: Error de posición del segundo blanco en medidas de dos blancos (50 a 80MHz).

En las gráficas de las figuras Figura 62, Figura 63, Figura 64 y Figura 65 están representados el error en amplitud y distancia de varias medidas de laboratorio, superpuestas a las gráficas simuladas en el apartado de simulación, para el margen de frecuencias de 50 a 80MHz. Se observa que, en general, la posición de los blancos detectados se ajusta a las simulaciones, dando unos errores en distancia muy parecidos a los simulados. Como ya se ha visto en la Figura 58, la amplitud de los blancos detectados queda por debajo de la amplitud simulada. Este efecto sólo se da en medidas de más de un blanco, y en principio no se sabe a qué es debido (aunque se cree que puede ser a la coherencia del láser). En las gráficas de amplitud se puede

ver que este efecto afecta a todas las medidas, y por lo tanto se puede conjeturar que existe para cualquier distancia.

### Corrección: 1.29

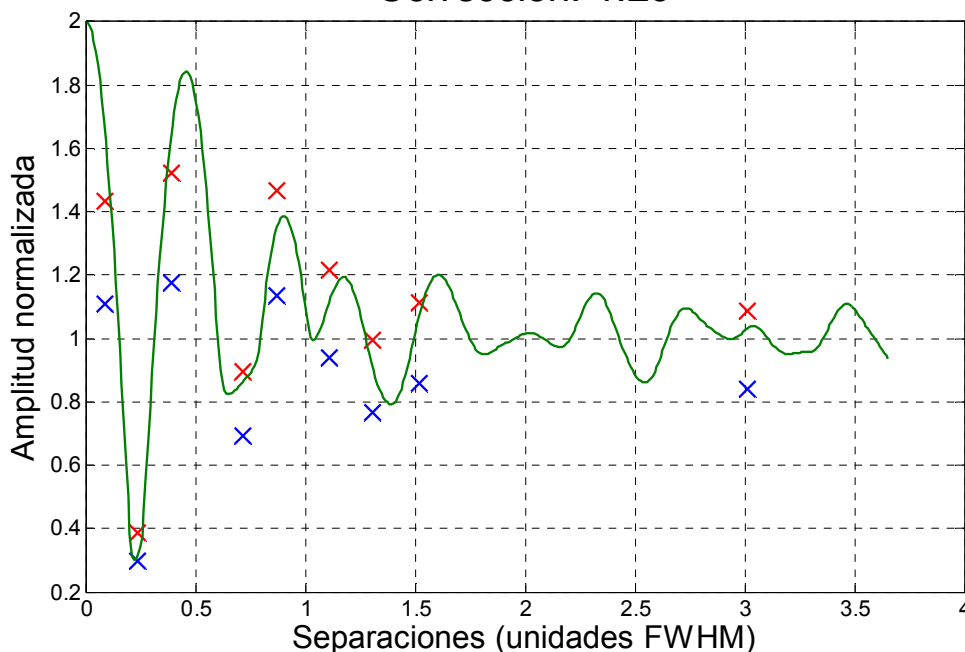


Figura 66: Error en amplitud para el primer blanco. Corrección que minimiza el error cuadrático medio en rojo (50 a 80MHz).

Se han elaborado gráficas (como la de la Figura 66) en las que esta amplitud se ha corregido multiplicándose por un factor que minimiza el error cuadrático medio entre las medidas y las simulaciones. Se observa que los puntos de la gráfica corregida se ajustan mucho mejor a las simulaciones, pero no llegan a cuadrar con exactitud. En el caso de ejemplo de la Figura 66, las medidas se han corregido multiplicando por un factor 1.29.

Se han hecho estas mismas medidas para otros rangos frecuenciales, dando lugar a resultados similares.

De nuevo se observa que los errores en distancia de las figuras Figura 67 (b) y (d) se ajustan correctamente a las simulaciones. Se observan igualmente dos puntos en la gráfica del error en distancia del segundo blanco (d) muy alejados de la simulación. Estos puntos no salen bien debido a que a estas separaciones uno de los blancos detectados es muy poco pronunciado, por lo que con muy pequeñas variaciones de los elementos del montaje del laboratorio puede o no detectarse ese máximo. Por lo tanto esas medidas, aunque son erróneas, están dentro de las limitaciones esperadas del sistema.

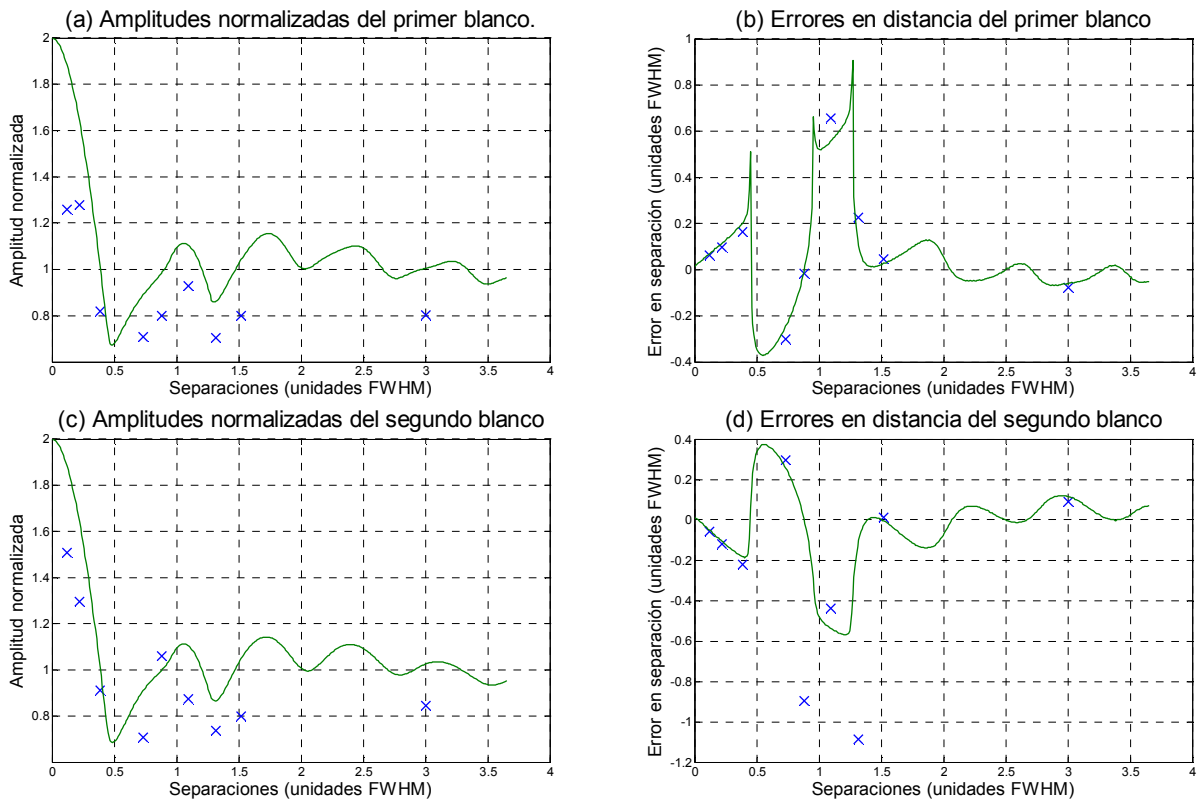


Figura 67: Error de amplitud y distancia para medidas de dos blancos (10 a 40MHz).

Aunque se ha mencionado que se eligen las separaciones a medir que generan máximos suficientemente pronunciados como para evitar este tipo de errores, estas separaciones se eligen para un margen frecuencial en concreto (normalmente de 50 a 80MHz), pero se medie también a otros márgenes, en los cuales esta medida de calidad puede no darse. Como en el caso anterior, las medidas de amplitud están afectadas por una atenuación, que no se ha detectado en las simulaciones, de valor parecido al caso anterior.

En el apartado 5.5 se han hecho varias medidas en diferentes situaciones con el fin de hallar el origen de esta atenuación no esperada.

### 5.3 Medidas de tres blancos

Como en el caso de dos blancos, para analizar las medidas de tres blancos se ha optado por superponerlas a las simulaciones realizadas en el apartado 3.5. Las medidas de tres blancos realizadas son prácticamente las mismas que las de dos blancos, pero añadiendo un blanco fijo más, tal y como se ha explicado en el apartado de simulaciones. Por ejemplo, para dos blancos fijos separados 5m, de 50 a 80MHz se obtiene:

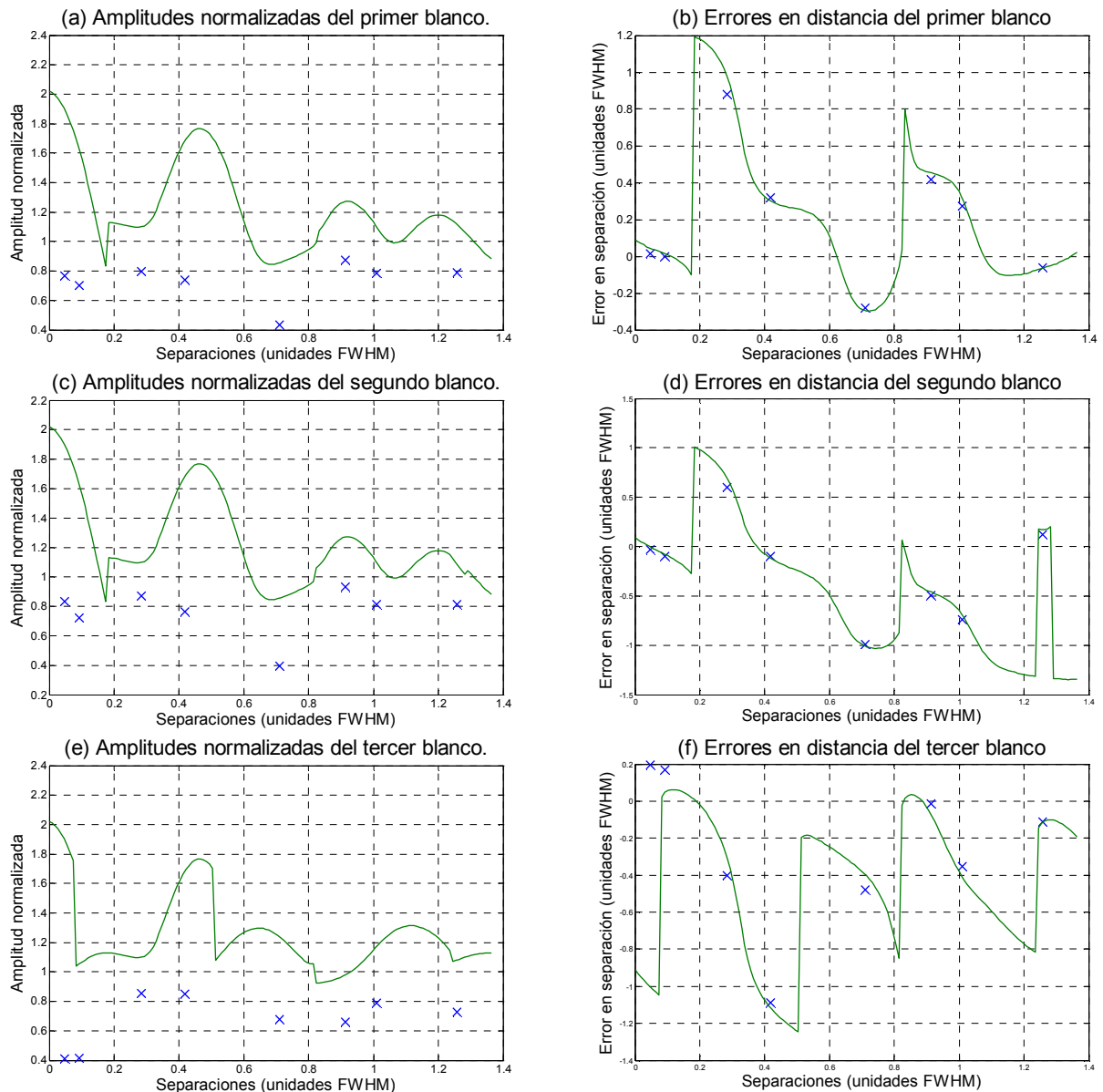


Figura 68: Medidas de tres blancos, dos fijos separados 5m (50 a 80MHz). En las gráficas de amplitudes, en azul tenemos la medida original, y en verde la medida multiplicada por el factor de corrección del título.

Como en el caso de dos blancos, se observa en la Figura 68 que las posiciones medidas de los máximos se corresponden con las simuladas (ya que coincide aproximadamente el error en la posición del blanco), pero las medidas están atenuadas por factores parecidos a los de el caso anterior. Este efecto se da de igual manera para los diferentes rangos frecuenciales medidos.

### 5.4 Medidas de blanco disperso

En un caso general en el que el sistema pretenda detectar aerosoles, la potencia reflejada por los mismos es muy pequeña en comparación con el ruido. Como ya vimos en el apartado de teoría, podemos reducir este ruido tanto como queramos (con ciertas limitaciones) a base de sumar periodos de la señal a la salida del sistema. Con el fin de ilustrar este comportamiento, en el laboratorio se ha tratado de detectar un



crystal que no estuviese perpendicular al haz de luz, y por lo tanto no tenemos en cuenta la máxima reflexión que pueda dar el cristal.

En una primera aproximación el problema parece sencillo, basta con poner el cristal en el camino del haz luminoso, y capturar suficientes periodos hasta detectar el blanco. Pero no es así, ya que de este modo no sólo se reduce el ruido, sino que aumenta en gran medida la sensibilidad del sistema, y se captan también muchos ecos débiles inesperados.

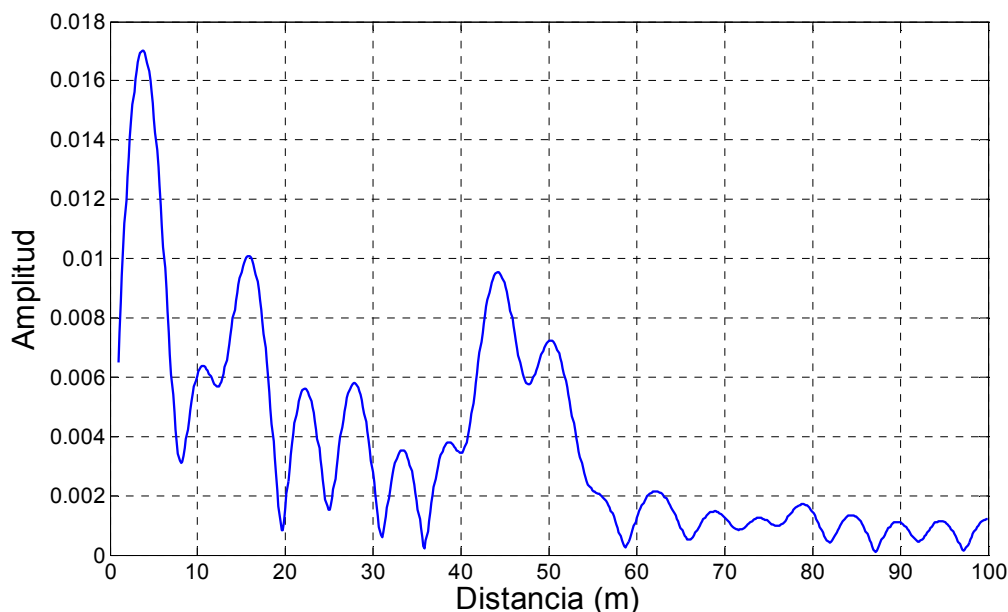


Figura 69: Respuesta a un blanco disperso a unos 50m (promediado de 100.000 periodos).

En la Figura 69 podemos observar la respuesta de un blanco disperso a unos 50m. Como vemos, el sistema detecta multitud de objetos que reflejan luz, y que además enmascaran el blanco. Se hace necesario pues un estudio del background antes de insertar el blanco a medir. Además, es posible minimizar algunas partes del background con unas pequeñas modificaciones del sistema.

#### 5.4.1 Estudio y optimización del Background

En la Figura 70 podemos observar una realización del background, obtenida después de integrar 100.000 periodos de 1ms.

La primera parte del background, correspondiente a aproximadamente los primeros 40m, es debida a rebotes de la señal eléctrica, y por lo tanto no será optimizado, ya que esta parte queda fuera del rango de distancias de estudio.

Observamos un rebote considerable aproximadamente a 52m. Este rebote lo origina la pared del fondo del laboratorio, que pese a ser negra, refleja suficiente luz como para ser detectada cuando se integran muchos periodos. Para minimizar y alejar este rebote, se ha colocado un espejo que desvía el haz luminoso hacia otra esquina de la sala, dónde rebota oblicuamente contra una mampara negra. Tras esto, el efecto del laboratorio "finito" se traslada unos 10 metros, además de atenuarse considerablemente.

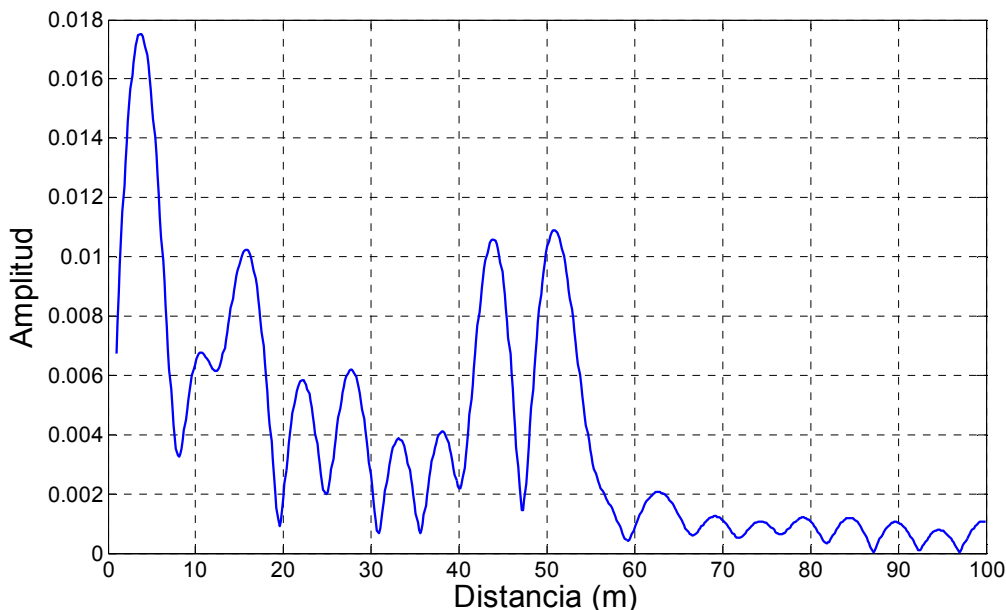


Figura 70: Medida de una realización del background sin optimizar (promediado de 100.000 periodos).

Además del haz principal, existe un haz luminoso secundario, que es el resultante del primer rebote con el *beam splitter*. Este haz secundario es interpretado por el sistema exactamente igual que el haz principal, y por lo tanto también puede reflejarse y aparecer en el background. Igual que en el caso anterior, estos rebotes se reducen y alejan considerablemente colocando un espejo que desvíe el haz luminoso secundario hacia un lugar con reflexión mínima.

En la Figura 70 podemos observar también que existen reflexiones muy cercanas al *beam splitter* que está situado aproximadamente a la distancia de 44m. Estos falsos blancos son debidos principalmente a dos fenómenos:

1. Captación directa de luz del láser:

El emisor láser no tiene un diagrama de radiación perfecto, por lo tanto una pequeña parte de la luz emitida se radia en direcciones diferentes a la del haz principal, emitiendo así luz directamente al receptor óptico. De la misma manera, el receptor óptico es capaz de detectar señal luminosa fuera de la dirección de máxima detección. Aunque la suma de estos dos efectos da lugar a una señal directa detectada muy pequeña, es comparable a la reflejada por un blanco disperso, y por lo tanto es necesario reducirla.

Se ha optado por colocar conos de cartulina negra que evitan el camino directo (o el proveniente de rebotes en la mesa) entre el emisor y el detector se consigue reducir notablemente este efecto.

2. Iluminación del *beam splitter*:

El *beam splitter* refleja una pequeña parte de la luz proveniente del emisor láser hacia el detector óptico. De nuevo es una señal pequeña, pero comparable al blanco disperso.

La única manera de eliminar esta contribución es eliminar el *beam splitter* del montaje (ya que el haz luminoso no puede eludir este elemento), pero este cristal es un

elemento indispensable del montaje actual, y por lo tanto esta señal siempre estará presente.

Con estas modificaciones, el montaje general cambia ligeramente, como se muestra en la Figura 71.

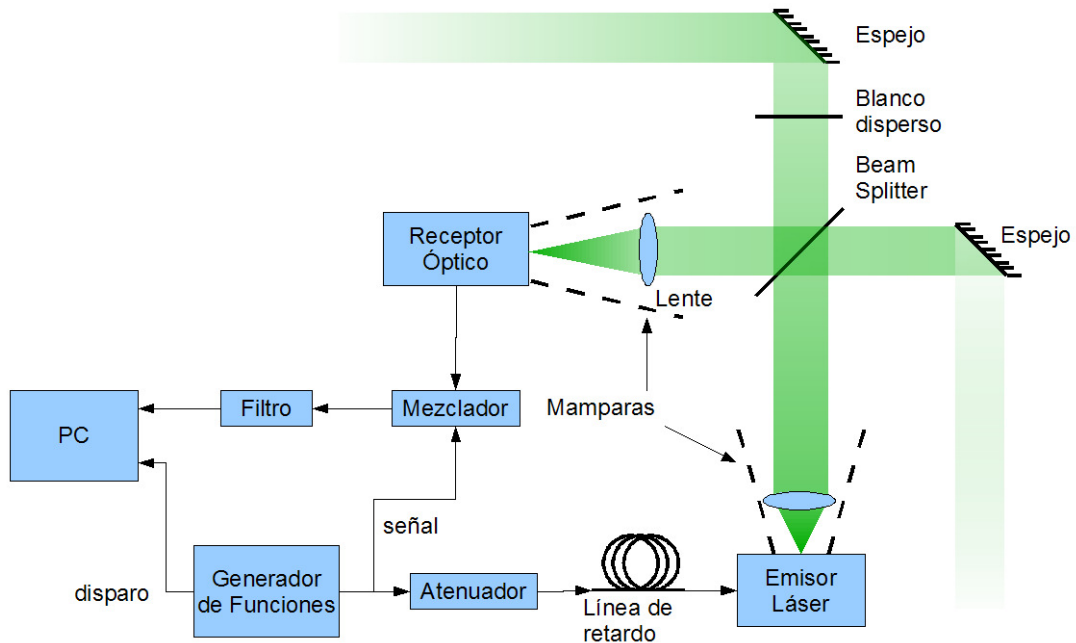


Figura 71: Montaje general modificado para reducir el background.

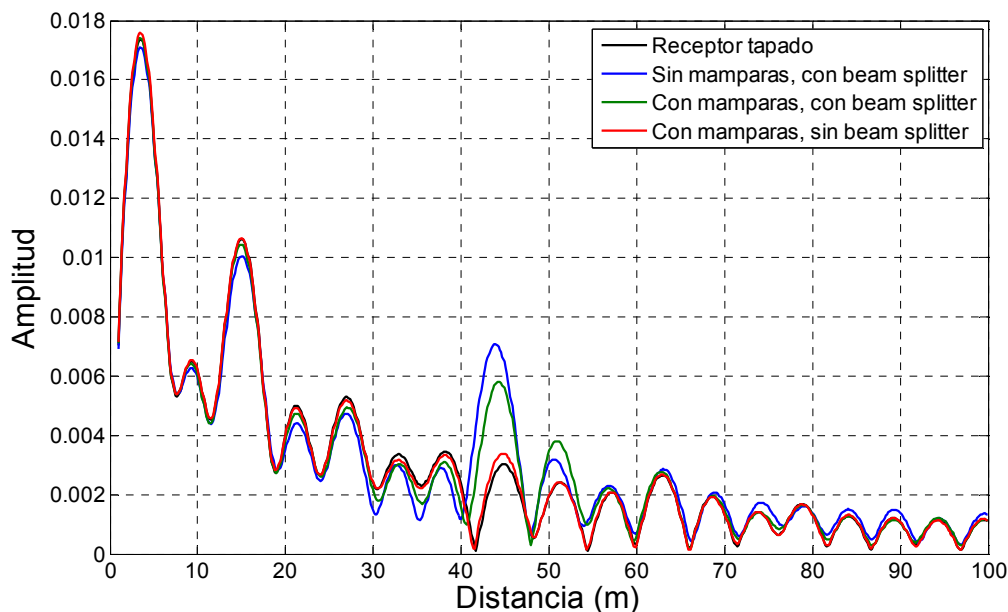


Figura 72: Optimización del background, varias opciones.

En la Figura 72 se observan varias realizaciones del background, haciendo los cambios descritos anteriormente para optimizarlo. Para todas las medidas, se han colocado los espejos que se describen para "alejar" la pared del fondo del laboratorio, y desviar el primer rebote del haz luminoso en el beam splitter para que no regrese al sistema. El mínimo background que se puede conseguir corresponde al caso en el que se tapa completamente el receptor óptico. Con el receptor óptico tapado no es posible hacer medidas, pero fija un background objetivo. Se observa que colocando las mamparas

que evitan el camino directo entre emisor y receptor se reduce el background, pero no lo suficiente. La única manera de conseguir un background parecido al objetivo es eliminar el beam splitter del montaje general.

### 5.4.2 Resta de background

Por mucho que se quiera optimizar, siempre existirá un background asociado a rebotes eléctricos u ópticos. Además, este background cambia lentamente a través del tiempo, por lo que será necesario medirlo antes (y después) de cada medida de un blanco disperso. En nuestro caso, el procedimiento a seguir será medir el background (sin el blanco) y después colocar el blanco y repetir la medida, promediando el mismo número de periodos. Las diferencias entre las medidas con o sin blanco serán las que determinarán la potencia y posición de éste, tal y como se observa en la Figura 73.

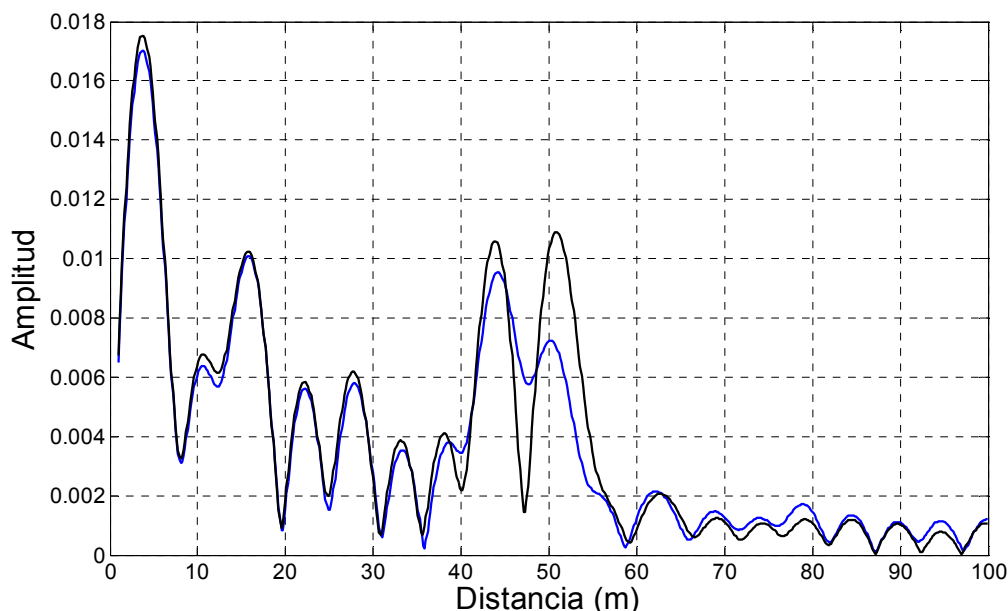


Figura 73: Respuesta a un blanco disperso a unos 50m y background asociado.

Para conseguir gráficas más precisas podemos restar el background a la medida del blanco, y así obtener gráficas parecidas al caso de blancos con reflexión especular. Después de restar el background a una medida de un blanco, se obtiene la gráfica de la Figura 74. Al contrario que en la gráfica de la medida sin restar background, en esta gráfica se observa claramente la detección de un blanco a unos 50m.

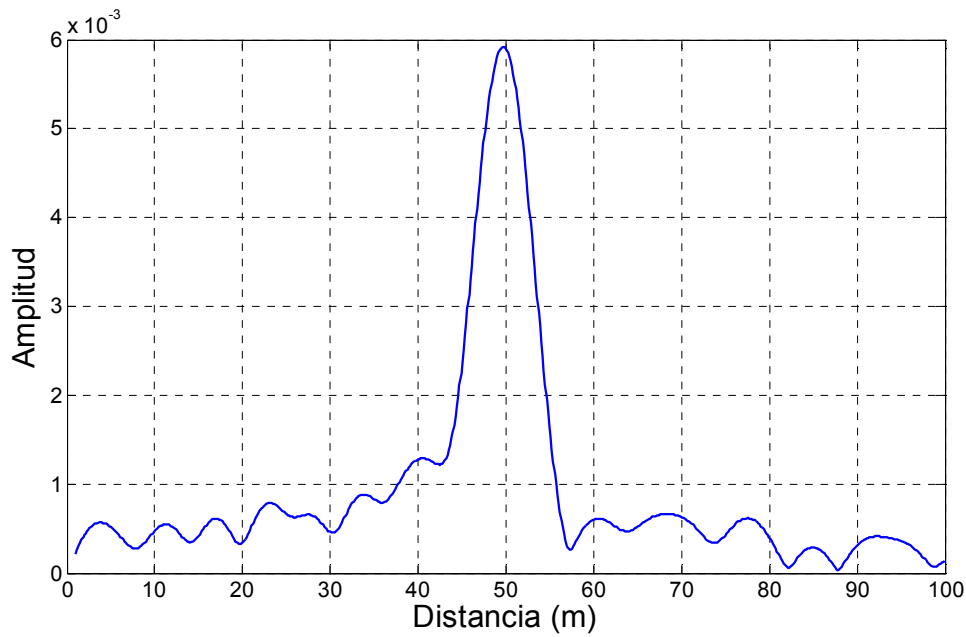


Figura 74: Respuesta a un blanco disperso a unos 50m (background restado).

### 5.4.3 Montaje alternativo

En el apartado 5.4.1 se explica cómo reducir el background añadiendo pequeñas modificaciones al sistema, pero la parte de background generada por el beam splitter no se puede optimizar, ya que este elemento es parte indispensable del montaje actual. De todas maneras, se puede definir un montaje alternativo que no necesite beam splitter.

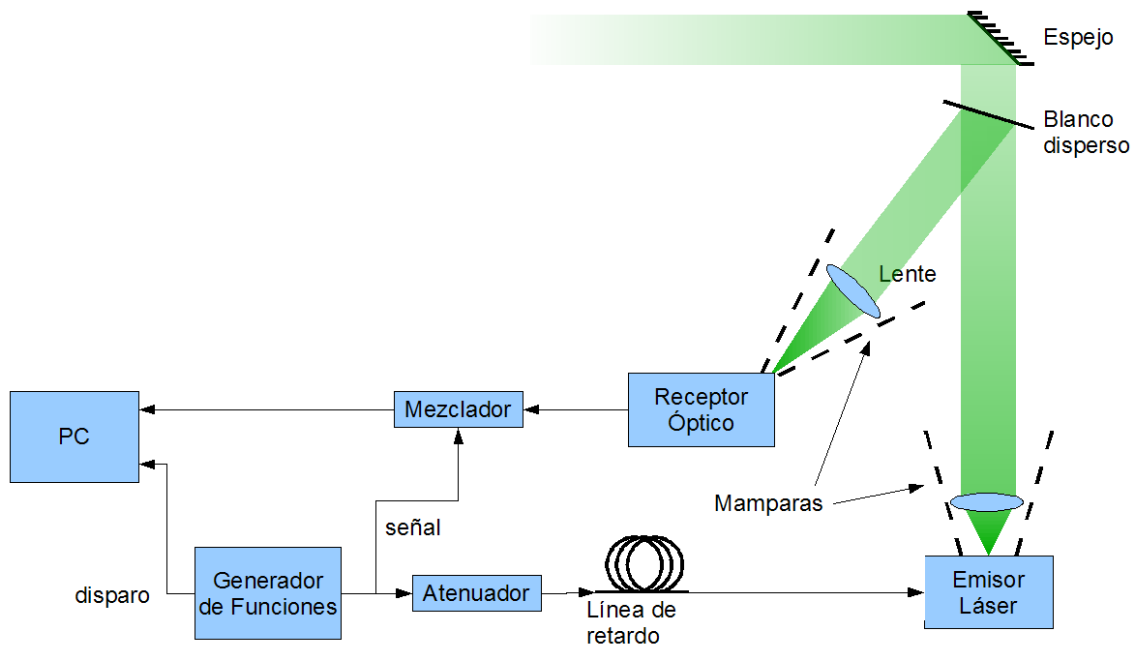


Figura 75: Montaje alternativo del laboratorio sin beam splitter.

En la Figura 75 se ha exagerado mucho la inclinación del blanco disperso. Para un blanco suficientemente separado del sistema, el haz luminoso de ida y el de vuelta son

prácticamente paralelos, pero separados una distancia suficiente para que se puedan colocar el emisor y el receptor uno al lado del otro. Además también se ha eliminado el filtro del montaje, ya que este elemento produce rebotes considerables en la parte eléctrica del background. Este cambio no afecta al sistema, ya que la tarjeta de adquisición del PC tiene un filtro paso-bajo a la entrada.

Utilizando este montaje alternativo, se puede volver a medir el background, esperando que se haya reducido considerablemente debido a la desaparición del *beam splitter*.

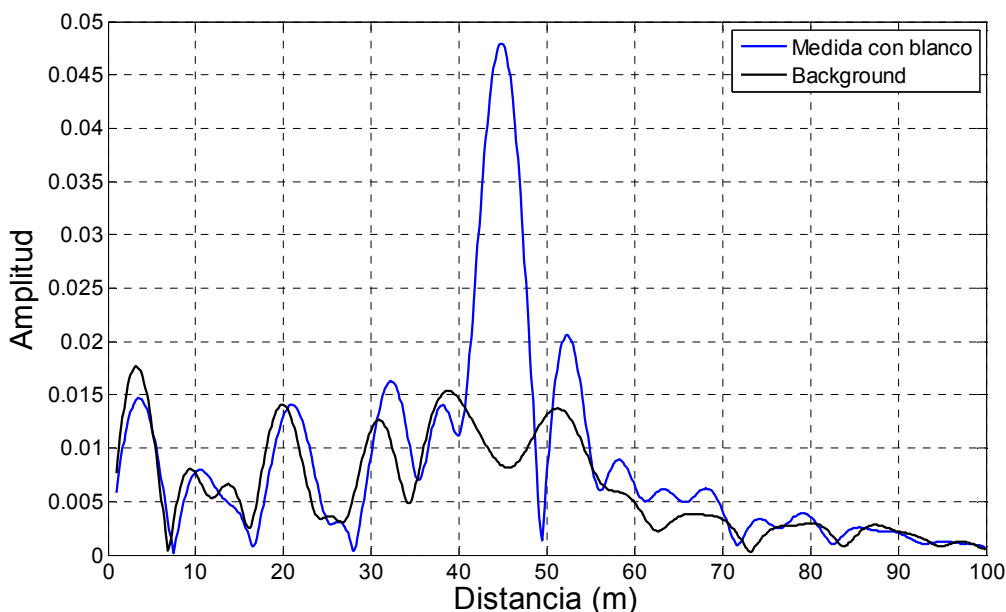


Figura 76: Medida de un blanco disperso y background usando el montaje alternativo.

En la Figura 76 se observa la medida de un blanco disperso usando el montaje alternativo, además del background asociado. Se aprecia que el background se ha reducido notablemente respecto del montaje original. El background en la zona de medida de los blancos, que en el montaje original era muy grande debido al *beam splitter*, se ha reducido de manera que se pueden incluso detectar blancos dispersos directamente sin necesidad de restar el background. Se observa también que se ha reducido el background de la parte baja del espectro, esto es debido a que también se ha eliminado el filtro del montaje original, ya que originaba una fuerte reflexión en la parte baja del espectro. En la Figura 77 se puede ver con más claridad la diferencia en el background medido con o sin el filtro.

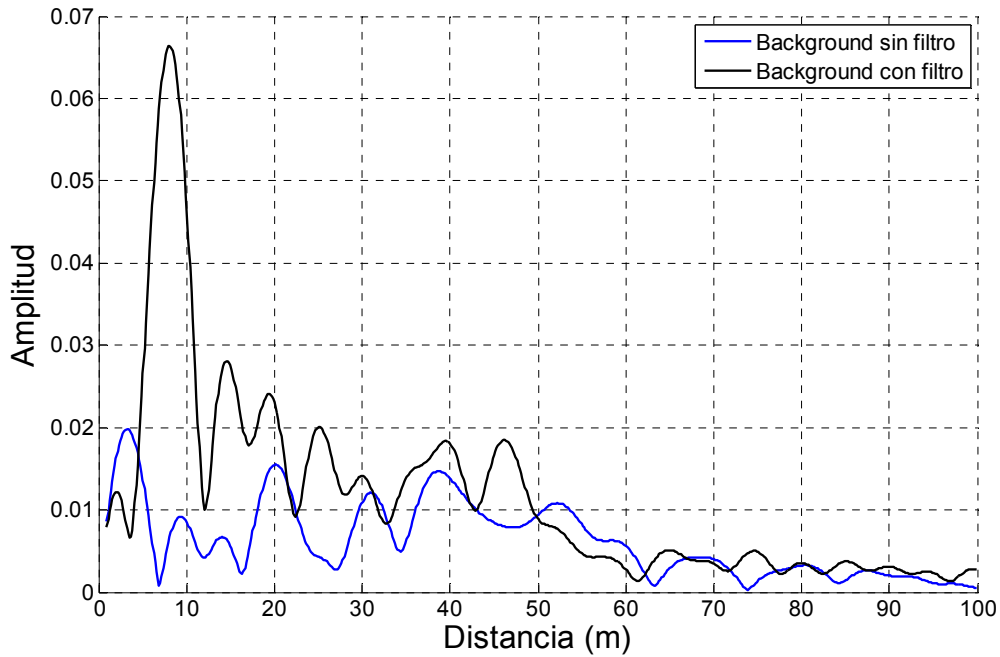


Figura 77: Medidas de background con y sin filtro.

### 5.4.4 Diagrama de reflexión del cristal

Midiendo la respuesta del vidrio con diferentes inclinaciones se puede confeccionar su diagrama de reflexión:

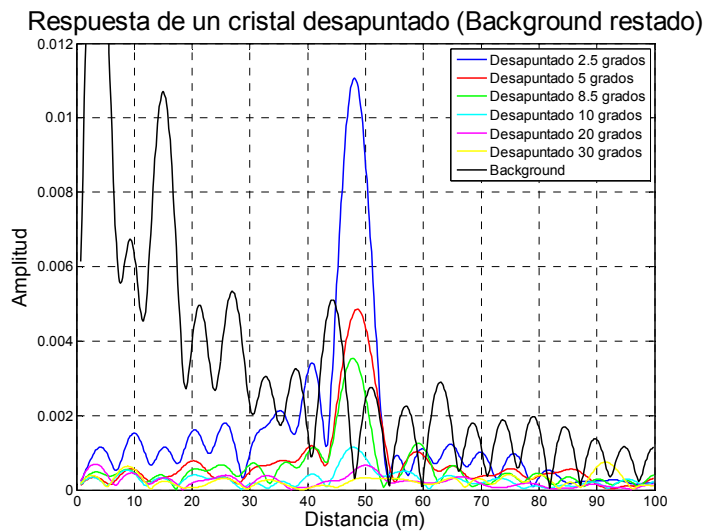


Figura 78: Respuesta de un cristal con diferentes inclinaciones (background restado).

Como se aprecia en la Figura 78, se detecta un blanco a unos 48m cuya potencia desciende a medida que se inclina el cristal. Tomando las medidas relativas de la amplitud del máximo de cada blanco, se puede representar el diagrama de reflexión del cristal:

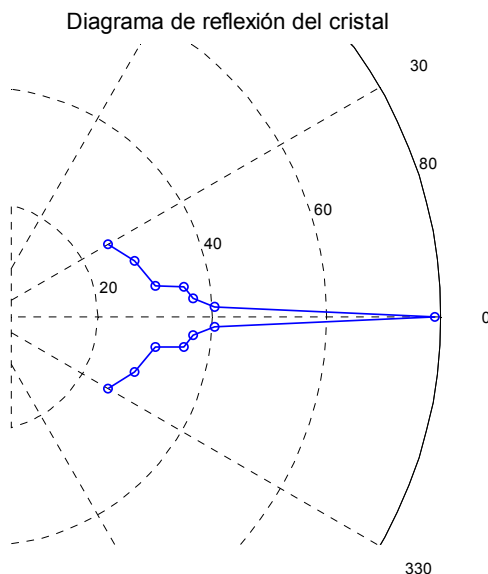


Figura 79: Respuesta Diagrama de reflexión del cristal en dB.

En la Figura 79 podemos apreciar que cuando nos alejamos de la dirección perpendicular la potencia cae drásticamente (43dB a unos 80). A partir de aproximadamente 10° la potencia reflejada se mantiene prácticamente constante, aunque en realidad, esta potencia reflejada puede ser aun más pequeña, ya que como se aprecia en la Figura 78 las medidas de estas inclinaciones dan resultados por debajo del background.

## 5.5 Medidas de coherencia del láser

En todas las medidas de más de un blanco realizadas durante el proyecto se ha observado un efecto de atenuación que no predice ni la teoría ni las simulaciones. Con el fin de caracterizar esta atenuación, se han formulado dos hipótesis:

- La atenuación se debe a cuestiones geométricas generadas por la posición de los elementos ópticos del sistema, de manera que las diferentes reflexiones de la señal luminosa dan lugar a una interferencia destructiva.
- La atenuación está introducida por la propia coherencia del láser, la potencia reflejada por dos blancos separados una distancia menor que la longitud de coherencia del láser sea diferente de la suma las potencias de cada blanco por separado.

Se han hecho una serie de medidas adicionales de dos blancos que tratan de ilustrar el efecto de estas hipótesis.

### 5.5.1 Medidas de dos blancos muy separados

Se han hecho una serie de medidas de dos blancos separados una distancia de 15m, con el fin de ver si esta distancia es menor que la longitud de coherencia del láser. Además, esta separación se ha hecho coincidir con el triple de la distancia entre ceros



de la función *sinc*, por lo que máximo de cada blanco se encuentra en la posición de un cero del otro. En estas medidas, se ha dejado un blanco fijo, mientras al otro se le ha ido incrementando su potencia, y así poder ver el efecto de atenuación en un rango amplio de amplitudes. La teoría y las simulaciones predicen que en este caso el máximo de cada blanco no se debe ver afectado por el otro.

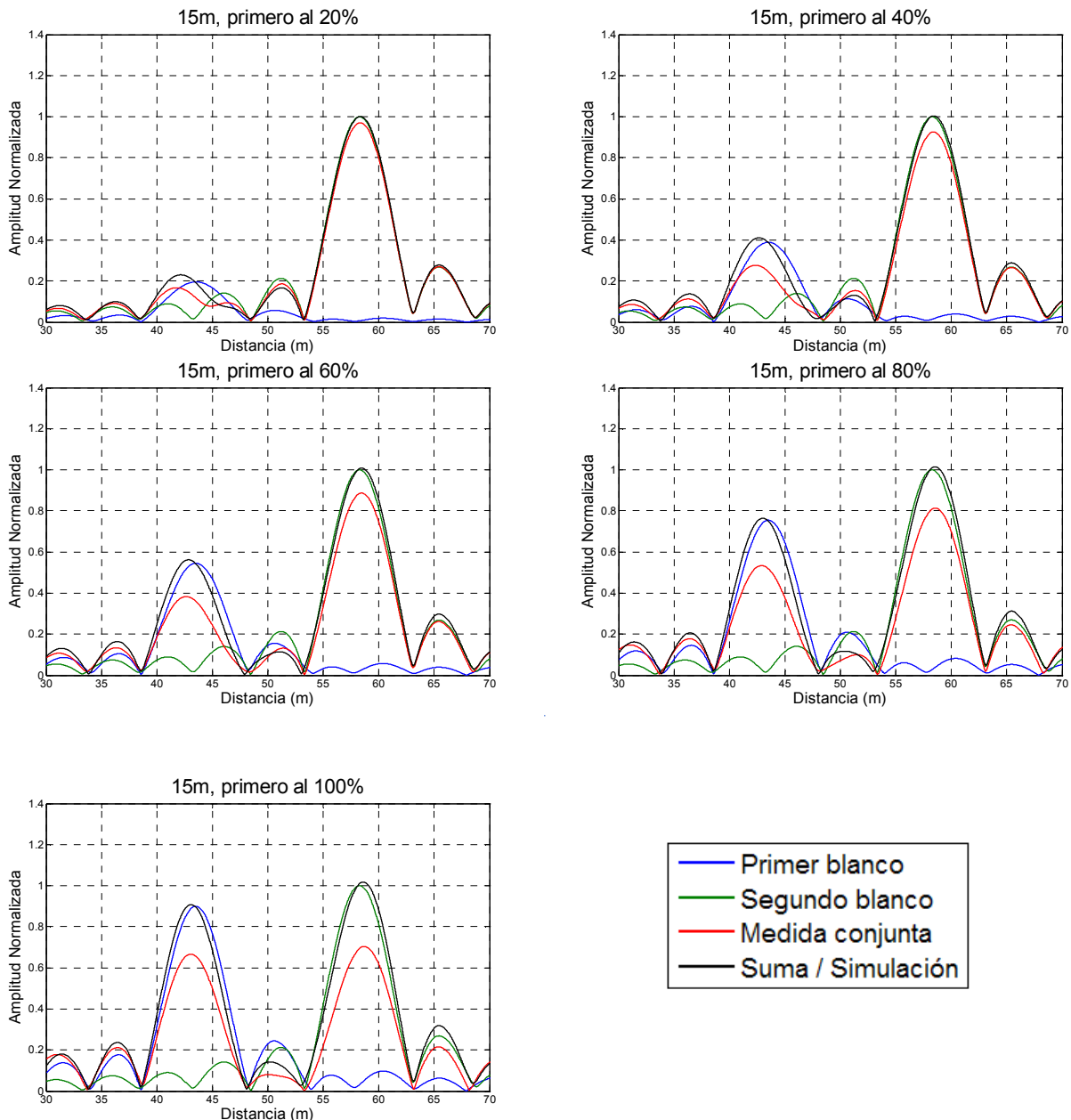


Figura 80: Varias medidas de dos blancos a 15m (50 a 80MHz), en las que el primer blanco aumenta de potencia.

En la Figura 80 están representadas las medidas de dos blancos separados un múltiplo entero de la distancia entre ceros. En cada medida se ha ido incrementando la potencia del primer blanco, siendo esta potencia un tanto por ciento aproximado de la potencia del segundo blanco. Se observa como a medida que la potencia del primer blanco crece, la medida de los dos blancos conjuntamente se atenúa. Sin embargo la suma de los dos blancos por separado (que es muy similar al resultado de simular dos blancos en las posiciones teóricas) se mantiene prácticamente igual para el segundo blanco, ya que se ha hecho coincidir el blanco en un cero del primero. Este efecto se ha reproducido de igual manera para otros rangos de frecuencia medidos:

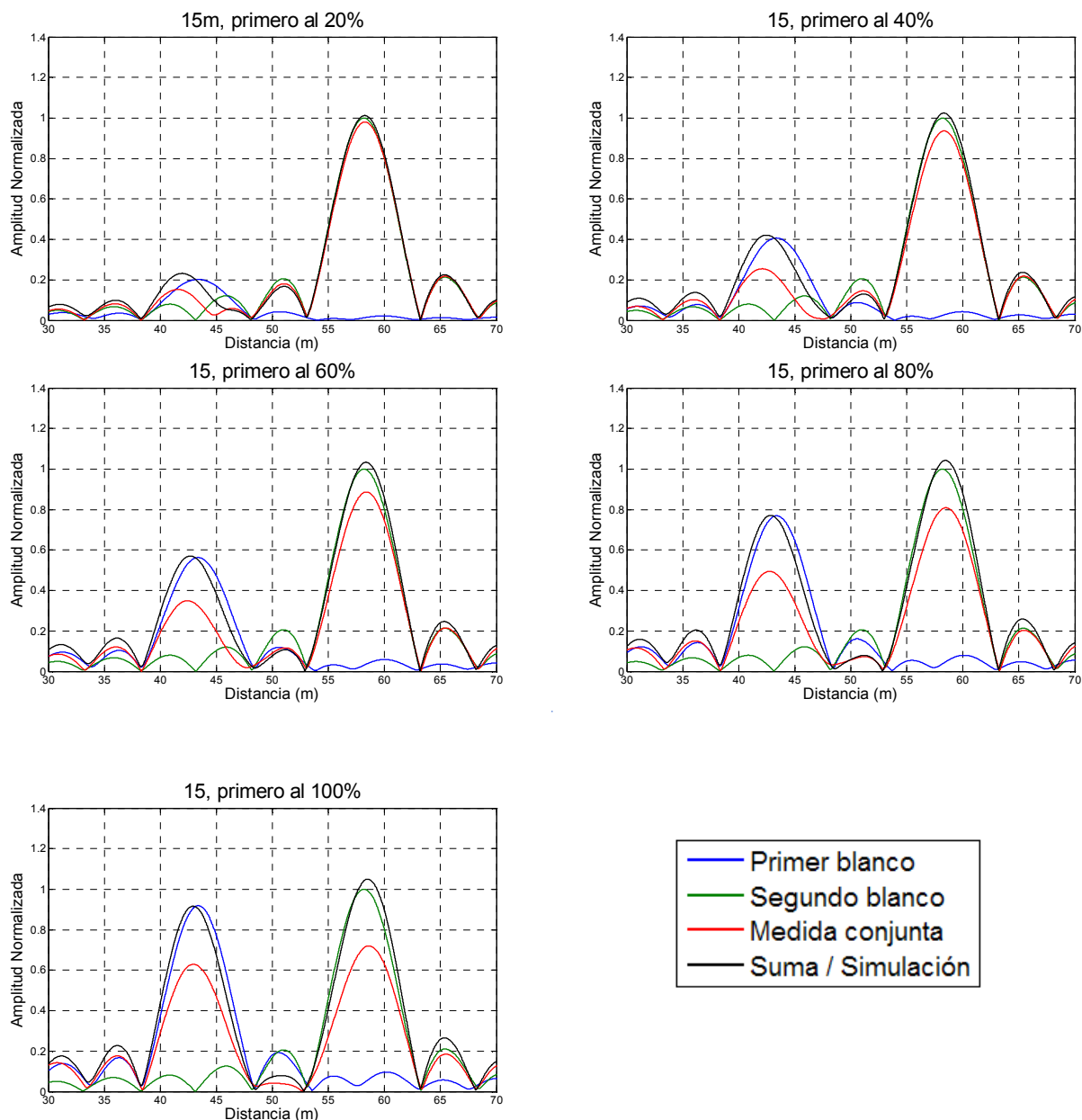


Figura 81: Varias medidas de dos blancos a 15m (10 a 40MHz), en las que el primer blanco aumenta de potencia.

En la Figura 81 se observa el mismo comportamiento que en las gráficas de la Figura 80 pero en este caso las frecuencias van de 10 a 40MHz. De nuevo, a medida que aumenta la potencia del primer blanco, se atenúa la medida conjunta, aun cuando la suma de ambos blancos por separado se mantiene prácticamente constante para el segundo blanco.

Estas medidas ponen de manifiesto que esta atenuación no es debida a la calidad de las medidas, ya que se trata de un efecto que está fuera del modelo estudiado. Tampoco puede ser debida a factores geométricos del sistema, ya que para cada potencia del primer blanco, éste tiene una inclinación diferente. Además se observa que la longitud de coherencia del láser es mayor que 15m, con lo cual no pueden hacerse medidas sin que aparezca esta atenuación, ya que el laboratorio no es suficientemente grande.

### 5.5.2 Medidas de dos blancos desplazados

Se han repetido las medidas de dos blancos a diferentes separaciones, pero colocando los blancos más alejados del origen que en el caso del apartado 5.2. Mediante estas medidas se trata de ver si la atenuación detectada es significativamente diferente para este caso, por lo que en este supuesto la atenuación podría deberse a cuestiones geométricas del sistema.

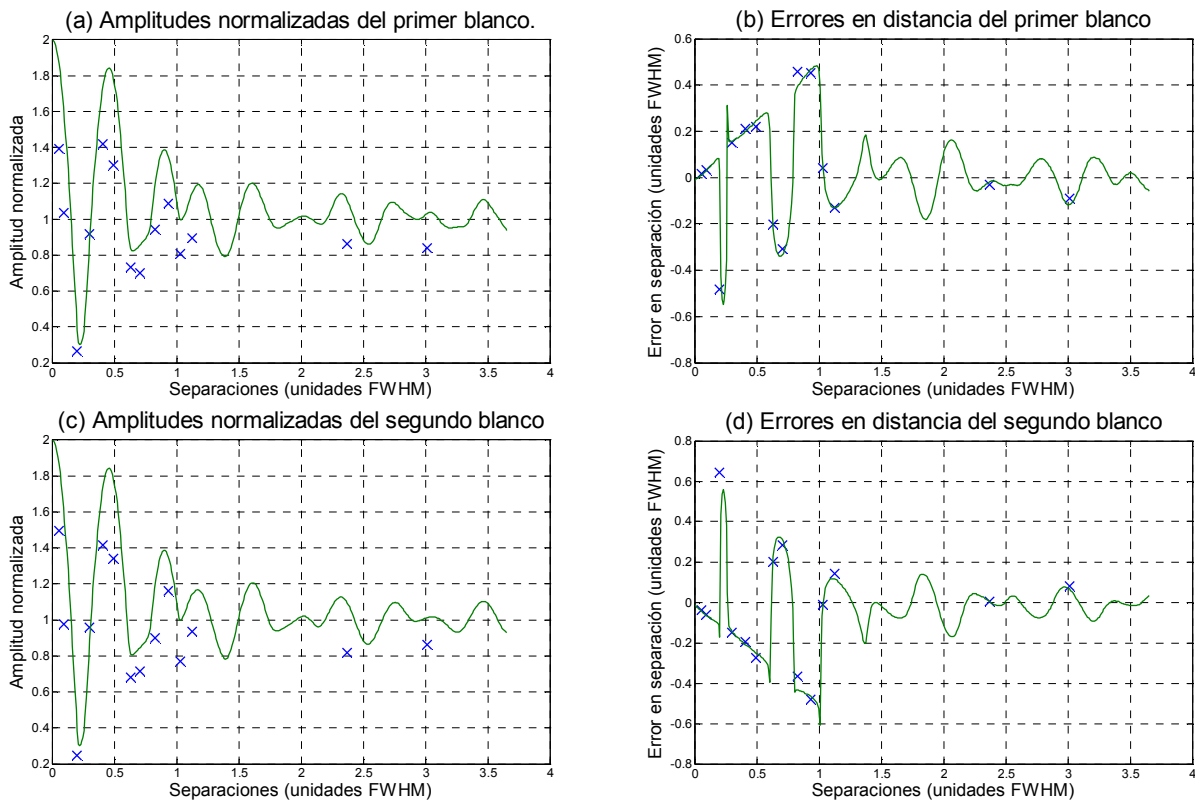


Figura 82: Medidas de dos blancos (50 a 80MHz) en posiciones más lejanas.

En la Figura 82 se ha representado el error en amplitud y en distancia para estos dos blancos a diferentes separaciones, y para el margen de frecuencias de 50 a 80MHz. Se observa que atenuación introducida en las medidas de dos blancos es parecida al caso de blancos más cercanos al origen.

En la Figura 83 se ha representado el error en amplitud del primer blanco para medidas de dos blancos que están más alejados del origen que los representados en la Figura 66. En este caso también se ha representado la versión corregida de las medidas que minimiza el error cuadrático medio de las medidas respecto de las simulaciones. Se observa que el factor de corrección aplicado 1,27, es prácticamente el mismo que en el caso de blancos más cercanos al origen representado en la Figura 66 (que era 1,29), por lo que probablemente esta atenuación no depende de la distancia al origen de los blancos. Estas medidas descartan que esta atenuación sea debida a factores geométricos del sistema, ya que cambiando de lugar los blancos, se cambian las condiciones geométricas, y por lo tanto el factor de atenuación debería de haber sido cualitativamente diferente.

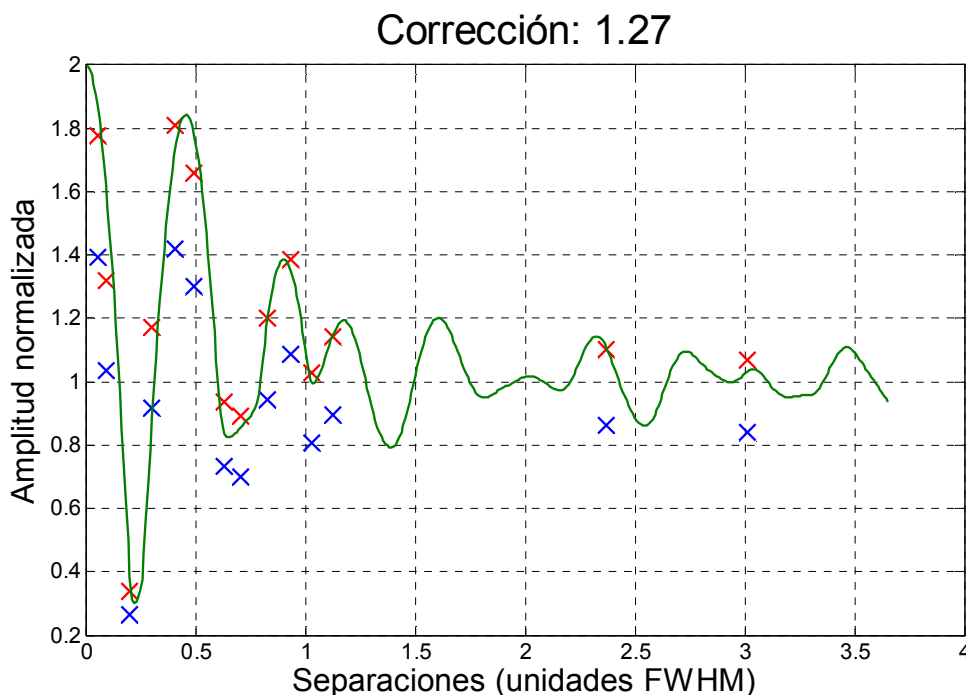


Figura 83: Error en amplitud para el primer blanco para blancos más alejados del origen. Corrección que minimiza el error cuadrático medio en rojo (50 a 80MHz).

### 5.5.3 Medidas de separaciones consecutivas

Se han realizado varias medidas cambiando la posición de uno de los blancos lo mínimo posible (en este caso han sido unos 10 cm, ya que los instrumentos del laboratorio no permiten precisar menos de esa longitud) tratando de ver interferencias constructivas y destructivas debidas a la coherencia del láser.

En la Figura 84 se han representado los errores en amplitud y distancia para dos blancos a diferentes separaciones muy parecidas entre ellas, con un margen de frecuencias de 50 a 80MHz. En las gráficas se observa que todas las medidas se comportan de igual manera, no habiendo ninguna que destaque respecto de las demás. Si alguna medida hubiese salido diferente de las demás, probablemente hubiese significado que se están dando situaciones en las que la señal interfiere consigo misma a nivel de longitud de onda del láser, debido a su coherencia. De todas maneras, las separaciones entre paso y paso tendrían que haber sido del orden de la longitud de onda para ver estas interferencias claramente, pero con los instrumentos disponibles en el laboratorio, esta medida no ha sido posible.

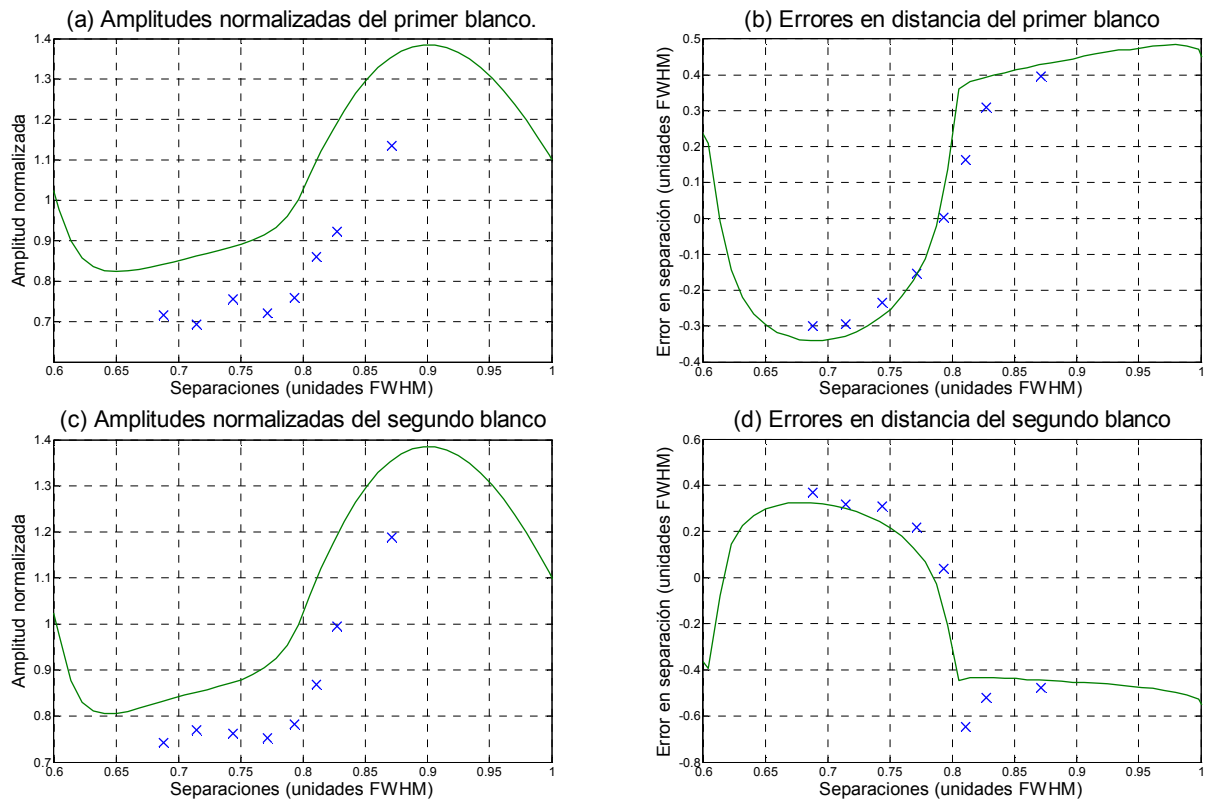


Figura 84: Medidas de dos blancos (50 a 80MHz) a separaciones consecutivas.



## **CAPÍTULO 6: CONCLUSIONES**

A lo largo de este proyecto se ha analizado la técnica FMCW en un sistema lidar, tratando de estudiar el caso concreto en el que dos o más blancos puntuales cercanos interactúan. Mediante el estudio teórico de la técnica, se ha llegado a la conclusión de que la fase relativa entre blancos depende de la separación entre estos. Esta dependencia implica que dependiendo de la separación entre blancos se pueden crear interferencias constructivas o destructivas que falseen la amplitud y la posición de los blancos detectados. Además, esta fase relativa también depende de la frecuencia inicial de la señal enviada, de forma que para valores altos de esta frecuencia, la fase cambia más rápidamente según la separación, dando lugar a fenómenos opuestos de interferencia con pequeños incrementos de separación.

La implementación de un modelo de simulación de la técnica FMCW acorde con la teoría ha permitido un mejor entendimiento de los fenómenos de interacción entre blancos, elaborándose gráficas que describan el error que introducen estas interferencias inter-blancos para dos y tres blancos.

Se ha construido un montaje de laboratorio que implementa la técnica FMCW. Este montaje ha permitido validar la teoría sobre la técnica FMCW así como el modelo de simulación salvo por una atenuación introducida en las medidas de más de un blanco que no se había predicho. Después de elaborar varias medidas tratando de explicar este comportamiento (que podría haber sido debido por ejemplo a cuestiones geométricas del sistema) no se ha conseguido hallar una explicación, aunque se sospecha que puede estar debido a la coherencia del propio láser utilizado en el laboratorio y que no esperábamos que fuese tan grande.

Se han hecho también medidas de blanco disperso (que en este proyecto se refiere a blancos que reflejan muy poca potencia). Se ha confirmado que sumando suficientes periodos de la señal recibida se puede reducir el ruido (ya que estos periodos están en fase) y así detectar blancos que reflejan potencia por debajo del nivel de ruido. Estas medidas también han puesto de manifiesto que es necesaria una optimización previa del background, ya que puede ser de amplitud considerable respecto a un blanco disperso.

En futuros proyectos se podría seguir investigando la coherencia del láser con el fin de explicar la procedencia de la atenuación no esperada en medidas de dos blancos. También se podrían hacer más medidas que no se han hecho durante este proyecto, como por ejemplo medidas de más de un blanco disperso, comprobando que interactúan de la manera esperada, o empezar a hacer medidas de un blanco con cierto grosor (formado por ejemplo por una acumulación de vapor de agua), tratando de estudiar el comportamiento del sistema para blancos continuos.

El siguiente paso a dar con el fin de introducir los resultados obtenidos en este proyecto en el Proyecto descrito en el apartado 1.1 puede ser medir concentraciones de gas entre dos blancos, tratando de unificar las técnicas WMS y FMCW.





## **CAPÍTULO 7: BIBLIOGRAFÍA**

- [1] R. Agishev et al. *Atmospheric CW-FM-LD-RR LADAR for trace-constituent detection: a concept development*. Appl. Phys. B 81, pp 695-703 2005.
- [2] A. Ansmann et al. *Optical Remote Measurements - Analysis of Atmospheric Processes with Optical Methods*. 2005. Disponible en <<http://lidar.tropos.de/en/index.html>>
- [3] O. Batet. *Continuous-wave Low-power Range-resolved Lidar Systems for Gas Detection and Concentration Measurement*. Ph.D. Thesis proposal 2007.
- [4] X. Chu. *Laser Remote Sensing. Fundamentals and Applications in Science and Engineering*. 2006. Disponible en <<http://superlidar.colorado.edu/Classes/Lidar2006/>>
- [5] R. M. Measures. *Laser remote sensing. Fundamental and applications*. Krieger Publishing Company 1992.
- [6] F. Rocardenbosch. *LIDAR Sensing of the Atmosphere. Receiver design and inversion Algorithms for an elastic system*. Doctor-Engineer Thesis 1996. Disponible en <<http://www.tdx.cat/TDX-0214108-152242>>



## ANEXO I: PROCEDIMIENTOS

### I.1 Separación de blancos: Automatización

Añadiremos una variable más al diseño del apartado de simulación para más de un blanco llamada `num_pasos`. Esta variable la inicializaremos con el número de pasos que separaremos los dos blancos partiendo de la separación cero. Los pasos serán de `resolucion` metros. Para cada separación es necesario simular el sistema completo. El resultado de esta simulación es el vector  $P$ , que contiene el módulo de la FFT de la señal de salida del mezclador. Mediante el algoritmo que se describe en el apartado siguiente, se estima la posición y el valor de los máximos, y por lo tanto de los blancos detectados. Podemos ver el proceso en el siguiente gráfico:

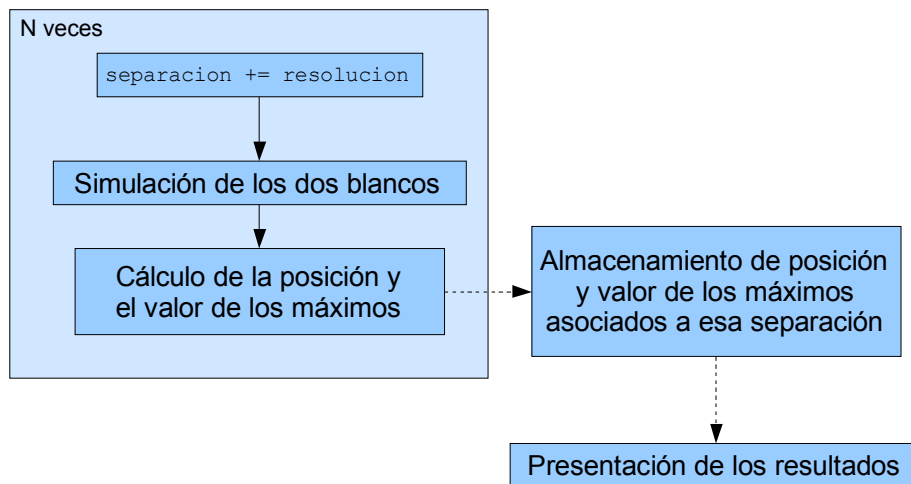


Figura 85: Procedimiento de separación y simulación de dos blancos

Una vez simulado el sistema para cada separación, obtenemos la posición y amplitud de los blancos detectados por el sistema asociados a esa separación, y por lo tanto podemos analizar el error introducido por la interacción de los dos blancos.

### I.2 Algoritmo de detección de máximos

En simulaciones de dos o más blancos se hace necesario un sistema de reconocimiento de blancos automatizado. En este apartado se propone un procedimiento que, dado un vector que contenga la simulación (o la medida) de uno o más blancos, se calculan las posiciones de los blancos detectados.

Dado el vector  $P$  que contiene el módulo de la FFT de la señal a la salida del mezclador, es necesario un sistema automático de cálculo de la posición y el valor de los máximos relativos candidatos a ser blancos detectados. Se ha optado por utilizar el algoritmo descrito en la siguiente gráfica:

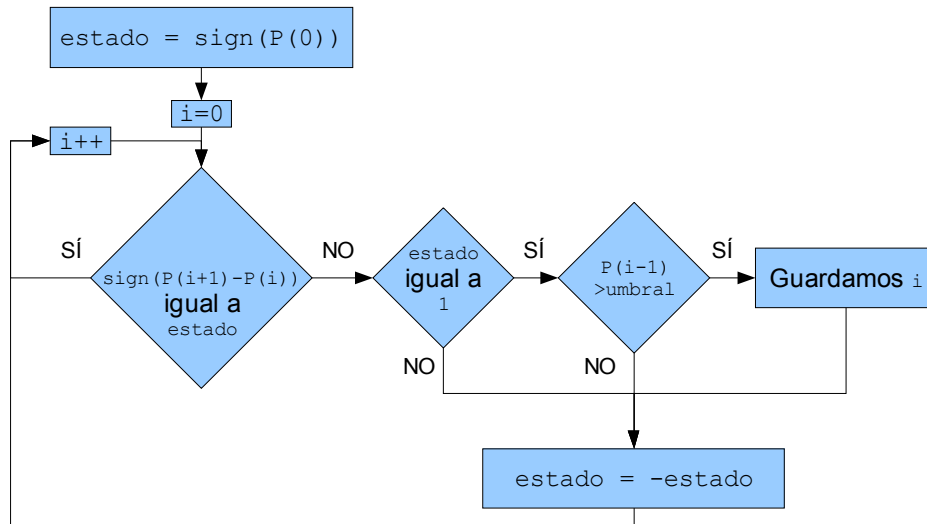


Figura 86: Método diferencial de búsqueda de máximos en un vector

En el esquema de la **Error! No se encuentra el origen de la referencia.** podemos comprobar que cada muestra del vector  $P$  se resta de la siguiente. Si el signo de la resta es diferente del signo de la resta anterior significa que estamos en un punto singular. De hecho si el valor de la resta es negativo, se puede asegurar también que se trata de un máximo relativo y por lo tanto es un candidato a ser un blanco detectado.

Hasta este punto se han detectado todos los máximos relativos del vector, pero queda por saber cuáles de ellos corresponden a blancos detectados. Para ello se compara el valor del máximo con un umbral dado. Si el valor del máximo es mayor que el umbral, decidimos que se ha detectado un blanco en ese punto.

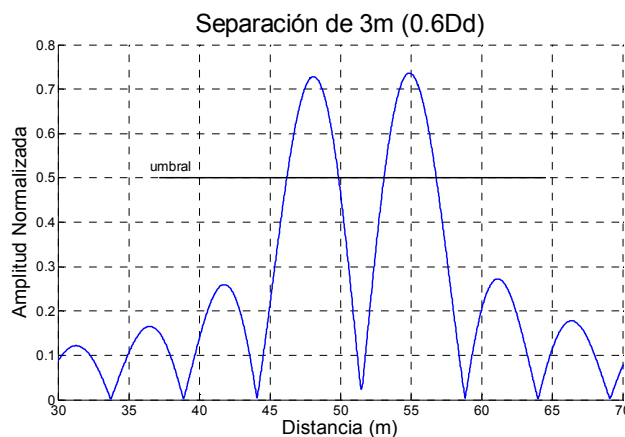


Figura 87: Posible valor del umbral a 0.5 para la correcta detección de dos blancos

La elección del umbral depende de cada caso, y en nuestro caso se ha hecho mediante el mecanismo de prueba y error. Dado que estamos simulando dos blancos, si el sistema detecta más de dos blancos, significa que el umbral es demasiado bajo, y por lo tanto hay que incrementarlo y probar de nuevo.

### I.3 Ejemplo de captura de datos: tres blancos

Para ilustrar mejor los procedimientos de igualación de potencias y captura de datos para varios blancos, se explican para el caso concreto de tres blancos. Llamaremos a

los blancos blanco 1, blanco 2 y blanco 3, siendo el blanco 1 el más cercano al sistema, y el blanco 3 el más alejado.

En primer lugar se debe alinear el sistema óptico colocando únicamente el blanco 3, maximizado lo posible la potencia detectada. Con el blanco 3 tapado, se coloca el blanco 2 y se alinea, pero sin alterar los demás componentes del sistema. Análogamente, con los blancos 2 y 3 tapados, se procede a la colocación del blanco 1, alineándolo correctamente.

Una vez alineados, se anulan los blancos 1 y 2 variando su ángulo de elevación, de forma que la potencia detectada por el sistema debido a estos blancos sea prácticamente cero. En este momento, se está detectando únicamente el blanco 3. La amplitud del blanco 3 será la amplitud objetivo para los otros 2. Con el blanco 3 tapado, se vuelve a apuntar el blanco 2, ajustando tanto como se pueda su amplitud a la objetivo. De nuevo, con los blancos 2 y 3 tapados, se vuelve a apuntar el blanco 1, también ajustando su amplitud a la objetivo.

Con los blancos debidamente alineados y las potencias igualadas, ya se puede capturar los datos de los tres blancos a la vez. Tapando los blancos 2 y 3, se puede hacer la medida del blanco 1 aislado. Opcionalmente, se puede hacer también la medida de los blancos 1 y 2 simultáneamente, tapando el blanco 3. Para hacer la medida del blanco 2 aislado, se desapunta de nuevo el blanco 1 y se tapa el blanco 3. Llegados a este punto, el blanco 1 no se puede volver a apuntar exactamente igual que estaba antes, por lo que si se quiere volver a hacer una medida de los tres blancos simultáneos, se tendría que volver a medir el blanco 1. Por último, se desapunta el blanco 2 y se mide el blanco 3 con los otros dos anulados.

Las gráficas resultantes de estas medidas se pueden normalizar todas por el mismo valor, y así poder comparar las amplitudes relativas de los blancos aislados y en conjunto.



## ANEXO II: ARCHIVOS DE MATLAB

Simulación de uno o más blancos:

```

% Constantes
c = 3e8; %velocidad de la luz en m/s
% Opciones
Fm = 100000; %Inverso del periodo de la señal sweep en Hz
Fmin = 50e6; %frecuencia inicial en Hz
Fmax = 80e6; %frecuencia final en Hz
N = 2^15; %número de muestras útiles por periodo
puntos_fft = 2^23; %número de puntos de la fft
dmax = 100; %distancia máxima de los blancos a simular en m
blancos = [50 55]; %vector de posiciones de los blancos en m
amplitudes = [1 1]; %amplitudes normalizadas de los blancos
(normalmente todo a 1)
n = 1; %numero de periodos de la señal sweep
sigma_ruido = 5e2*0; %desviacion tipica del ruido aditivo. Dejar a 0
para simular sin ruido
normaliza = 1; %1 normaliza, 0 no normaliza

% Inicializaciones
Df = Fmax - Fmin;
fs = Fmin;
Dd = c/(2*Df);

Tm = 1/Fm;
t_resolucion = Tm/N;
resolucion = t_resolucion*c/2;
tmax = 2*dmax/c;
K = Df/Tm;
r = zeros(1,N);
normalizacion = 1;

% Cálculo de la frecuencia de muestreo
Fsample = 1/t_resolucion;
muestra_tmax = floor(tmax*Fsample);

% Implementación de un periodo del coseno (2N muestras)
fs_ficticio = fs-Df*tmax/2/Tm;
t = linspace(0,2*Tm-1/Fsample,2*N);
sweep = cos(2*pi*(fs_ficticio*t+K*t.^2/2));

% Señal_enviada (N muestras útiles)
senal_enviada = sweep(muestra_tmax:muestra_tmax+N-1);

% Bucle de los periodos de la señal sweep
for i = 1 : n
    % señal_recibida: Retardo de la señal_enviada
    td = 2*blancos/c;
    senal_recibida = zeros(1,N);
    % Bucle de los blancos
    for j = 1:size(td,2)
        long_vect0 = floor(Fsample*td(j));
        vector_retardo = zeros(1,long_vect0);
        senal_blanco_actual = [vector_retardo sweep];
        senal_recibida = senal_recibida +
        amplitudes(j)*senal_blanco_actual(muestra_tmax:muestra_tmax+N-1);
        % Cálculo de la normalización, solo para el primer blanco
    end
end

```

```
        if j == 1 && normaliza == 1
            s = senal_enviada .* senal_recibida;
            S = fft(s,puntos_fft);
            normalizacion = max(abs(S));
        end
    end
    % Mezclado
    r = r + (senal_enviada .* senal_recibida) ./ normalizacion +
    sigma_ruido*randn(1,N);
end
r = r/n;

% Cálculo de la fft
R = fft(r,puntos_fft);
P = abs(R);

% Presentación
fmax = tmax*K;
pos_max = floor(fmax*(puntos_fft/Fsample)) + 1;

d = linspace(0,dmax,pos_max);

plot(d,P(1:pos_max),'k-','LineWidth',2); %módulo
%plot(d,unwrap(angle(R(1:pos_max))), 'm-'); %fase
```



Simulación de dos blancos, generando las gráficas de error en amplitud y distancia:

```

% Constantes
c = 3e8; %velocidad de la luz en m/s
% Opciones
Fm = 100000; %Inverso del periodo de la señal sweep en Hz
Fmin = 20e6; %frecuencia inicial en Hz
Fmax = 80e6; %frecuencia final en Hz
N = 2^15; %número de muestras útiles por periodo
puntos_fft = 2^23; %número de puntos de la fft
umbral = 0.8; %umbral que debe sobrepasar un máximo para ser blanco
detectado

segundo_blanco = 50; %posición del blanco fijo en m
num_puntos = 200; %número de pasos de la simulación
dmax = 100; %distancia máxima de los blancos a simular en m

%directorio de guardado y carga
principio_archivos = 'E:\PFC\Datos simulaciones\dos blancos\buenos\';

guardar = 0; %1: guarda los resultados obtenidos
cargar = 1; %1: carga el resultado de las fft de una simulación
anterior

% Inicializaciones
Df = Fmax - Fmin;
Dd = c/(2*Df);
fs = Fmin;
Tm = 1/Fm;
t_resolucion = Tm/N;
resolucion = t_resolucion*c/2;
tmax = 2*dmax/c;
beta = 2*pi*Df/Tm;
Fsample = 1/t_resolucion;
muestra_tmax = floor(tmax*Fsample);
muestra_segundo_blanco =
floor(2*segundo_blanco*Df*puntos_fft/c/Tm/Fsample) + 1;
muestras_retardo_una_muestra = Df*puntos_fft/Tm/(Fsample^2);
mitad = muestra_segundo_blanco;
td = 2*segundo_blanco/c;
retardo_segundo_blanco = floor(Fsample*td);

if cargar == 0
    tresd = [];
    % Implementacion de un periodo del coseno.
    fs_ficticio = fs-Df*tmax/2/Tm;
    t = linspace(0,2*Tm-1/Fsample,2*N);
    x = cos(2*pi*fs_ficticio*t+beta*t.^2/2);
    % Señal recibida para el segundo blanco
    v0 = zeros(1,retardo_segundo_blanco);
    xr_segundo_blanco = [v0 x];
    xr_segundo_blanco = xr_segundo_blanco(muestra_tmax:muestra_tmax+N-
1);
    % Cálculo de la normalización
    a = fft(xr_segundo_blanco .* x(muestra_tmax:muestra_tmax+N-
1),puntos_fft);
    normalizacion = max(abs(a));
    xr_segundo_blanco = xr_segundo_blanco./normalizacion;
else

```

```

    final_archivos = [' ', segundo_blanco ' a '
sprintf('%d',segundo_blanco) 'm, ' sprintf('%d',Fmin/1e6) ' a '
sprintf('%d',Fmax/1e6) 'MHz,' sprintf('%d',num_puntos)];
    final_archivos = [final_archivos 'puntos, Fm='
sprintf('%d',Fm/1e3) 'kHz,puntos_fft=2^'];
    final_archivos = [final_archivos sprintf('%d',log2(puntos_fft))
',N=2^' sprintf('%d',log2(N)) '.txt'];
    archivo_tresd = [principio_archivos 'tresd' final_archivos];
    archivo_tresd_x = [principio_archivos 'tresd_x' final_archivos];
    est = 'loading' %#ok<NOPTS>
    tresd = load(archivo_tresd);
end

% Bucle principal, en cada iteración se calcula la posición del primer
% blanco, se suma al segundo y se buscan los máximos.
d = zeros(1,num_puntos);
maxs_primer_blanco = zeros(1,num_puntos);
pos_maxs_primer_blanco = zeros(1,num_puntos);
maxs_segundo_blanco = zeros(1,num_puntos);
pos_maxs_segundo_blanco = zeros(1,num_puntos);
curvas_max_d = [];
for i = 0:num_puntos-1
%for i = 68:68
%    i=i+46; %#ok<FXSET>
    i %#ok<NOPTS>
    muestra_primer_blanco = muestra_segundo_blanco -
muestras_retardo_una_muestra*i;
    d(i+1) = (i*c)/(Fsample*2);
    if cargar == 0
        % Señal recibida para el primer blanco
        v0 = zeros(1,retardo_segundo_blanco - i);
        xr_primer_blanco = [v0 x];
        xr_primer_blanco =
xr_primer_blanco(muestra_tmax:muestra_tmax+N-1)./normalizacion;
        % Suma de blancos
        xr = xr_primer_blanco + xr_segundo_blanco;
        % Mezclado de la entrada y la salida
        r = x(muestra_tmax:muestra_tmax+N-1) .* xr;
        R = fft(r,puntos_fft);
        Prr = abs(R);
        tresd = [tresd;Prr(1:2*mitad)]; %#ok<AGROW>
    else
        Prr = tresd(i+1,:);
    end

%%50 80
%if i == 19
%    umbral = 0.2;
%end
%if i == 33
%    umbral = 0.5;
%end

%20 80
if i == 24
    umbral = 0.6;
end
if i == 40
    umbral = 0.8;
end
end

```

```

%%otro
%if i == 48
%    umbral = 0.35;
%end

%Busca de máximos que puedan ser blancos en Prr
dPrr = [Prr(1:2*mitad) 0] - [0 Prr(1:2*mitad)];
signo = sign(dPrr);
maximos_d = [];
estado = signo(1);
for k = 2 : 2*mitad
    if signo(k) ~= estado
        %Estamos en un cero en k-1, veamos si es un maximo
        if estado == 1
            %Estamos en un maximo, veamos si supera el umbral
            if Prr(k-1) > umbral
                %blanco detectado. Se añade al vector
                maximos_d = [maximos_d k-1]; %#ok<AGROW>
            end
        end
        estado = -estado;
    end
end
num_maximos = size(maximos_d,2);
if num_maximos > 2
    error('Numero de maximos mayor a 2, aumenta el umbral!!');
end
if num_maximos == 0
    warning = 'Ningun máximo!!' %#ok<NOPTS>
    maximos_d = [1 1]; %#ok<AGROW>
    num_maximos = 2;
end
if num_maximos == 1
    maximos_d = [maximos_d(1) maximos_d]; %#ok<AGROW>
    num_maximos = 2;
end
if num_maximos == 2
    %añadiendo la posición y amplitud de los blancos detectados a
los
    %vectores de las gráficas
    maxs_primer_blanco(i+1) = Prr(maximos_d(1));
    pos_maxs_primer_blanco(i+1) = (maximos_d(1) -
muestra_primer_blanco)/puntos_fft*Fsample/Df*Tm*c/2/Dd;
    maxs_segundo_blanco(i+1) = Prr(maximos_d(2));
    pos_maxs_segundo_blanco(i+1) = (maximos_d(2) -
muestra_segundo_blanco)/puntos_fft*Fsample/Df*Tm*c/2/Dd;
    end
    curvas_max_d = [curvas_max_d;maximos_d]; %#ok<AGROW>
end

% Presentación
mitad_en_metros = (mitad*Fsample*Tm*c)/(puntos_fft*Df*2);
d_normalizada = d./Dd;
tresd_x = linspace(0,2*mitad_en_metros,2*mitad);
figure(1);
plot(d_normalizada,maxs_primer_blanco,'LineWidth',2);
xlabel('Separaciones (unidades FWHM)','fontsize',20);
ylabel('Amplitud normalizada','fontsize',20);
title('Amplitudes normalizadas del primer blanco','fontsize',26);
grid;

```

```

figure(2);
plot(d_normalizada,maxs_segundo_blanco,'LineWidth',2);
xlabel('Separaciones (unidades FWHM)','fontsize',20);
ylabel('Amplitud normalizada','fontsize',20);
title('Amplitudes normalizadas del segundo blanco','fontsize',26);
grid;
figure(3);
plot(d_normalizada,pos_maxs_primer_blanco,'LineWidth',2);
xlabel('Separaciones (unidades FWHM)','fontsize',20);
ylabel('Error en separación (unidades FWHM)','fontsize',20);
title('Errores en distancia del primer blanco','fontsize',26);
grid;
figure(4);
plot(d_normalizada,pos_maxs_segundo_blanco,'LineWidth',2);
xlabel('Separaciones (unidades FWHM)','fontsize',20);
ylabel('Error en separación (unidades FWHM)','fontsize',20);
title('Errores en distancia del segundo blanco','fontsize',26);
grid;
figure(5);
plot(Prr);
plot(tresd_x,Prr(1:(2*mitad)));
%figure(8);
%%plot(tresd_x,dPrr(1:(2*mitad)));
%surf(d_normalizada,tresd_x,tresd');
%xlabel('Separaciones (unidades FWHM)','fontsize',20);
%ylabel('Distancia (m)','fontsize',20);
%title('Evolución del los máximos 3D','fontsize',26);
figure(6);
plot(d_normalizada,linspace(segundo_blanco,muestra_primer_blanco/punto
s_fft*Fsample/Df*Tm*c/2,num_puntos),'-
',d_normalizada,segundo_blanco*ones(num_puntos),'-
',d_normalizada,curvas_max_d/puntos_fft*Fsample/Df*Tm*c/2,'x','LineWid
th',2);
xlabel('Separaciones (unidades FWHM)','fontsize',20);
ylabel('Distancia (m)','fontsize',20);
title('Evolución del los máximos','fontsize',26);
grid;

% Guardado
if guardar == 1
    final_archivos = ['segundo blanco a '
sprintf('%d',segundo_blanco) 'm, ' sprintf('%d',Fmin/1e6) ' a '
sprintf('%d',Fmax/1e6) 'MHz, ' sprintf('%d',num_puntos)];
    final_archivos = [final_archivos 'puntos, Fm='
sprintf('%d',Fm/1e3) 'kHz,puntos_fft=2^'];
    final_archivos = [final_archivos sprintf('%d',log2(puntos_fft))
',N=2^' sprintf('%d',log2(N)) '.txt'];
    archivo1 = [principio_archivos 'distancias_normalizadas'
final_archivos];
    archivo2 = [principio_archivos 'pot_primer_blanco'
final_archivos];
    archivo3 = [principio_archivos 'pot_segundo_blanco'
final_archivos];
    archivo5 = [principio_archivos 'err_dist_primer_blanco'
final_archivos];
    archivo6 = [principio_archivos 'err_dist_segundo_blanco'
final_archivos];
    archivo8 = [principio_archivos 'tresd_x' final_archivos];
    archivo9 = [principio_archivos 'tresd' final_archivos];

    save(archivo1,'d_normalizada','-ascii','-double','-tabs');

```

```
save(archivo2, 'maxs_primer_blanco', '-ascii', '-double', '-tabs');
save(archivo3, 'maxs_segundo_blanco', '-ascii', '-double', '-tabs');
save(archivo5, 'pos_maxs_primer_blanco', '-ascii', '-double', '-
tabs');
save(archivo6, 'pos_maxs_segundo_blanco', '-ascii', '-double', '-
tabs');
if cargar == 0
    save(archivo8, 'tresd_x', '-ascii', '-double', '-tabs');
    save(archivo9, 'tresd', '-ascii', '-double', '-tabs');
end
end
```

Simulación de tres blancos, generando las gráficas de error en amplitud y distancia:

```
% Constantes
c = 3e8; %velocidad de la luz en m/s
% Opciones
Fm = 100000; %Inverso del periodo de la señal sweep en Hz
Fmin = 10e6; %frecuencia inicial en Hz
Fmax = 40e6; %frecuencia final en Hz
N = 2^15; %número de muestras útiles por periodo
puntos_fft = 2^23; %número de puntos de la fft
umbral = 0.8; %umbral que debe sobrepasar un máximo para ser blanco
detectado

Dd = c/(2*Df);

segundo_blanco = 50;%en metros
tercer_blanco = segundo_blanco + 1*Dd;
num_puntos = 400;%en pasos
dmax = 100;

%directorio de guardado y carga
principio_archivos = 'E:\PFC\Datos simulaciones\tres blancos\buenos\';

generar = 0;
guardar = 0;
cargar = 1;

Tm = 1/Fm;
t_resolucion = Tm/N;
resolucion = t_resolucion*c/2;
tmax = 2*dmax/c;
beta = 2*pi*Df/Tm;
% Calculo de la frecuencia de muestreo y numero total de muestras.
Fsample = 1/t_resolucion;
muestra_tmax = floor(tmax*Fsample);

muestra_segundo_blanco =
floor(2*segundo_blanco*Df*puntos_fft/c/Tm/Fsample) + 1;
muestra_tercer_blanco =
floor(2*tercer_blanco*Df*puntos_fft/c/Tm/Fsample) + 1;
muestras_retardo_una_muestra = Df*puntos_fft/Tm/(Fsample^2);

td = 2*segundo_blanco/c;
retardo_segundo_blanco = floor(Fsample*td);
td = 2*tercer_blanco/c;
retardo_tercer_blanco = floor(Fsample*td);
if generar == 1
    tresd = [];
end
if generar == 1

% Implementacion de un periodo del coseno.
fs_ficticio = fs-Df*tmax/2/Tm;
t = linspace(0,2*Tm-1/Fsample,2*N);
x = cos(2*pi*fs_ficticio*t+beta*t.^2/2);

%para el segundo blanco:
```

```

%Calculo de los rebotes
v0 = zeros(1,retardo_segundo_blanco);
xr_segundo_blanco = [v0 x];
xr_segundo_blanco = xr_segundo_blanco(muestra_tmax:muestra_tmax+N-1);
a = fft(xr_segundo_blanco .* x(muestra_tmax:muestra_tmax+N-1),puntos_fft);
maximo = max(abs(a));
xr_segundo_blanco = xr_segundo_blanco./maximo;

%para el tercer blanco:
v0 = zeros(1,retardo_tercer_blanco);
xr_tercer_blanco = [v0 x];
xr_tercer_blanco = xr_tercer_blanco(muestra_tmax:muestra_tmax+N-1);

xr_segundo_blanco = xr_segundo_blanco + xr_tercer_blanco./maximo;

else
    if cargar == 1
        final_archivos = [' ', segundo_blanco ' a '
sprintf('%d',segundo_blanco) 'm, tercer blanco a '
sprintf('%d',tercer_blanco) 'm,' sprintf('%d',Fmin/1e6) ' a '
sprintf('%d',Fmax/1e6) 'MHz,' sprintf('%d',num_puntos)];
        final_archivos = [final_archivos 'puntos, Fm='
sprintf('%d',Fm/1e3) 'kHz,puntos_fft=2^'];
        final_archivos = [final_archivos
sprintf('%d',log2(puntos_fft)) ',N=2^' sprintf('%d',log2(N)) '.txt'];
        archivo_tresd = [principio_archivos 'tresd' final_archivos];
        archivo_tresd_x = [principio_archivos 'tresd_x'
final_archivos];
        est = 'loading'
        tresd = load(archivo_tresd);
    end
end
posicion_real_segundo_blanco = retardo_segundo_blanco*c/Fsample/2;

%bucle principal
d = zeros(1,num_puntos);
maxs_primer_blanco = zeros(1,num_puntos);
pos_maxs_primer_blanco = zeros(1,num_puntos);
maxs_segundo_blanco = zeros(1,num_puntos);
pos_maxs_segundo_blanco = zeros(1,num_puntos);
maxs_tercer_blanco = zeros(1,num_puntos);
pos_maxs_tercer_blanco = zeros(1,num_puntos);
curvas_max_d = [];
for i = 0:num_puntos-1
    i %#ok<NOPTS>
    muestra_primer_blanco = muestra_segundo_blanco -
muestras_retardo_una_muestra*i;
    mitad1 = floor((muestra_primer_blanco + muestra_segundo_blanco) /
2);
    mitad2 = floor((muestra_segundo_blanco + muestra_tercer_blanco) /
2);
    d(i+1) = (i*c)/(Fsample*2);
    if generar == 1
        v0 = zeros(1,retardo_segundo_blanco - i);
        xr_primer_blanco = [v0 x];
        xr_primer_blanco = xr_primer_blanco(muestra_tmax:muestra_tmax+N-1)./maximo;
    end
end

```

```

% Suma de los diferentes periodos y adiccion de ruido
xr = xr_primer_blanco + xr_segundo_blanco;

% Mezclado de la entrada y la salida
r = x(muestra_tmax:muestra_tmax+N-1) .* xr;

R = fft(r,puntos_fft);
Prr = abs(R);
tresd = [tresd;Prr(1:2*mitad2)]; %#ok<AGROW>
else
    Pr = tresd(i+1,:);
end

if i == 136
    umbral = 0.9;
end
if i == 137
    umbral = 0.8;
end

%% 20 a 80
%if i == 68
%    umbral = 0.875;
%end
%if i == 71
%    umbral = 0.8;
%end

%Metodo diferencial
dPrr = [Prr(1:2*mitad2) 0] - [0 Pr(1:2*mitad2)];
signo = sign(dPrr);
maximos_d = [];
estado = signo(1);
for k = 2 : 2*mitad2
    if signo(k) ~= estado
        %Estamos en un cero en k-1, veamos si es un maximo
        if Pr(k-1) > Pr(k)
            %Estamos en un maximo, veamos si supera el umbral
            if Pr(k-1) > umbral
                maximos_d = [maximos_d k-1]; %#ok<AGROW>
            end
        end
        estado = -estado;
    end
end
maximos_d = [maximos_d zeros(1,3-size(maximos_d,2))]; %#ok<AGROW>
num_maximos = size(maximos_d,2);
if num_maximos > 3
    error('Numero de maximos mayor a 3, aumenta el umbral!!');
end

if num_maximos == 2
    maximos_d = [maximos_d maximos_d(2)]; %#ok<AGROW>
    num_maximos = 3;
end

```



```

    if num_maximos == 1
        maxs_primer_blanco(i+1) = Prr(maximos_d);
        pos_maxs_primer_blanco(i+1) = (maximos_d -
muestra_primer_blanco)/puntos_fft*Fsample/Df*Tm*c/2/Dd;
        maxs_segundo_blanco(i+1) = Prr(maximos_d);
        pos_maxs_segundo_blanco(i+1) = (maximos_d -
muestra_segundo_blanco)/puntos_fft*Fsample/Df*Tm*c/2/Dd;
        maxs_tercer_blanco(i+1) = Prr(maximos_d);
        pos_maxs_tercer_blanco(i+1) = (maximos_d -
muestra_tercer_blanco)/puntos_fft*Fsample/Df*Tm*c/2/Dd;
        maximos_d = [maximos_d maximos_d maximos_d]; %#ok<AGROW>
    end
    if num_maximos == 2
        if abs(maximos_d(1) - muestra_primer_blanco) <
abs(maximos_d(2) - muestra_primer_blanco)
            maxs_primer_blanco(i+1) = Prr(maximos_d(1));
            pos_maxs_primer_blanco(i+1) = (maximos_d(1) -
muestra_primer_blanco)/puntos_fft*Fsample/Df*Tm*c/2/Dd;
        else
            maxs_primer_blanco(i+1) = Prr(maximos_d(2));
            pos_maxs_primer_blanco(i+1) = (maximos_d(2) -
muestra_primer_blanco)/puntos_fft*Fsample/Df*Tm*c/2/Dd;
        end
        if abs(maximos_d(1) - muestra_segundo_blanco) <
abs(maximos_d(2) - muestra_segundo_blanco)
            maxs_segundo_blanco(i+1) = Prr(maximos_d(1));
            pos_maxs_segundo_blanco(i+1) = (maximos_d(1) -
muestra_segundo_blanco)/puntos_fft*Fsample/Df*Tm*c/2/Dd;
            maximos_d_2 = [maximos_d(1) maximos_d]; %#ok<AGROW>
        else
            maxs_segundo_blanco(i+1) = Prr(maximos_d(2));
            pos_maxs_segundo_blanco(i+1) = (maximos_d(2) -
muestra_segundo_blanco)/puntos_fft*Fsample/Df*Tm*c/2/Dd;
            maximos_d_2 = [maximos_d maximos_d(2)]; %#ok<AGROW>
        end
        if abs(maximos_d(1) - muestra_tercer_blanco) <
abs(maximos_d(2) - muestra_tercer_blanco)
            maxs_tercer_blanco(i+1) = Prr(maximos_d(1));
            pos_maxs_tercer_blanco(i+1) = (maximos_d(1) -
muestra_tercer_blanco)/puntos_fft*Fsample/Df*Tm*c/2/Dd;
        else
            maxs_tercer_blanco(i+1) = Prr(maximos_d(2));
            pos_maxs_tercer_blanco(i+1) = (maximos_d(2) -
muestra_tercer_blanco)/puntos_fft*Fsample/Df*Tm*c/2/Dd;
        end
        maximos_d = maximos_d_2;
    end

    if num_maximos == 3
        maxs_primer_blanco(i+1) = Prr(maximos_d(1));
        pos_maxs_primer_blanco(i+1) = (maximos_d(1) -
muestra_primer_blanco)/puntos_fft*Fsample/Df*Tm*c/2/Dd;
        maxs_segundo_blanco(i+1) = Prr(maximos_d(2));
        pos_maxs_segundo_blanco(i+1) = (maximos_d(2) -
muestra_segundo_blanco)/puntos_fft*Fsample/Df*Tm*c/2/Dd;
        maxs_tercer_blanco(i+1) = Prr(maximos_d(3));
        pos_maxs_tercer_blanco(i+1) = (maximos_d(3) -
muestra_tercer_blanco)/puntos_fft*Fsample/Df*Tm*c/2/Dd;
    end
    curvas_max_d = [curvas_max_d;maximos_d]; %#ok<AGROW>
end

```

```

mitad2_en_metros = (mitad2*Fsample*Tm*c)/(puntos_fft*Df*2);
d_normalizada = d./Dd;
tresd_x = linspace(0,2*mitad2_en_metros,2*mitad2);
figure(1);
plot(d_normalizada,maxs_primer_blanco,'LineWidth',2);
xlabel('Separaciones (unidades FWHM)','fontsize',20);
ylabel('Amplitud normalizada','fontsize',20);
title('Amplitudes normalizadas del primer blanco','fontsize',26);
grid;
figure(2);
plot(d_normalizada,maxs_segundo_blanco,'LineWidth',2);
xlabel('Separaciones (unidades FWHM)','fontsize',20);
ylabel('Amplitud normalizada','fontsize',20);
title('Amplitudes normalizadas del segundo blanco','fontsize',26);
grid;
figure(3);
plot(d_normalizada,maxs_tercer_blanco,'LineWidth',2);
xlabel('Separaciones (unidades FWHM)','fontsize',20);
ylabel('Amplitud normalizada','fontsize',20);
title('Amplitudes normalizadas del tercer blanco','fontsize',26);
grid;
figure(4);
plot(d_normalizada,pos_maxs_primer_blanco,'LineWidth',2);
xlabel('Separaciones (unidades FWHM)','fontsize',20);
ylabel('Error en separación (unidades FWHM)','fontsize',20);
title('Errores en distancia del primer blanco','fontsize',26);
grid;
figure(5);
plot(d_normalizada,pos_maxs_segundo_blanco,'LineWidth',2);
xlabel('Separaciones (unidades FWHM)','fontsize',20);
ylabel('Error en separación (unidades FWHM)','fontsize',20);
title('Errores en distancia del segundo blanco','fontsize',26);
grid;
figure(6);
plot(d_normalizada,pos_maxs_tercer_blanco,'LineWidth',2);
xlabel('Separaciones (unidades FWHM)','fontsize',20);
ylabel('Error en separación (unidades FWHM)','fontsize',20);
title('Errores en distancia del tercer blanco','fontsize',26);
grid;
figure(7);
plot(Prr);
plot(tresd_x,Prr(1:(2*mitad2)));
%figure(8);
%%plot(tresd_x,dPrr(1:(2*mitad2)));
%surf(d_normalizada,tresd_x,tresd');
xlabel('Separaciones (unidades FWHM)','fontsize',20);
ylabel('Distancia (m)','fontsize',20);
%title('Evolución del los máximos 3D','fontsize',26);
figure(8);
plot(d_normalizada,linspace(segundo_blanco,muestra_primer_blanco/punto
s_fft*Fsample/Df*Tm*c/2,num_puntos),'-
',d_normalizada,segundo_blanco*ones(1,num_puntos),'-
',d_normalizada,tercer_blanco*ones(1,num_puntos),'-
',d_normalizada,curvas_max_d/puntos_fft*Fsample/Df*Tm*c/2,'x','LineWid
th',2);

xlabel('Separaciones (unidades FWHM)','fontsize',20);
ylabel('Distancia (m)','fontsize',20);
title('Evolución del los máximos','fontsize',26);
grid;

```

```

if guardar == 1
    principio_archivos = 'E:\PFC\Datos simulaciones\tres
blancos\buenos\';
    final_archivos = ['segundo blanco a '
sprintf('%d',segundo_blanco) 'm, tercer blanco a '
sprintf('%d',tercer_blanco) 'm,' sprintf('%d',Fmin/1e6) ' a '
sprintf('%d',Fmax/1e6) 'MHz,' sprintf('%d',num_puntos)];
    final_archivos = [final_archivos 'puntos, Fm='
sprintf('%d',Fm/1e3) 'kHz,puntos_fft=2^'];
    final_archivos = [final_archivos sprintf('%d',log2(puntos_fft))
',N=2^' sprintf('%d',log2(N)) '.txt'];
    archivo1 = [principio_archivos 'distancias_normalizadas'
final_archivos];
    archivo2 = [principio_archivos 'pot_primer_blanco'
final_archivos];
    archivo3 = [principio_archivos 'pot_segundo_blanco'
final_archivos];
    archivo4 = [principio_archivos 'pot_tercer_blanco'
final_archivos];
    archivo5 = [principio_archivos 'err_dist_primer_blanco'
final_archivos];
    archivo6 = [principio_archivos 'err_dist_segundo_blanco'
final_archivos];
    archivo7 = [principio_archivos 'err_dist_tercer_blanco'
final_archivos];
    archivo8 = [principio_archivos 'tresd_x' final_archivos];
    archivo9 = [principio_archivos 'tresd' final_archivos];

    save(archivo1,'d_normalizada','-ascii','-double','-tabs');
    save(archivo2,'maxs_primer_blanco','-ascii','-double','-tabs');
    save(archivo3,'maxs_segundo_blanco','-ascii','-double','-tabs');
    save(archivo4,'maxs_tercer_blanco','-ascii','-double','-tabs');
    save(archivo5,'pos_maxs_primer_blanco','-ascii','-double','-
tabs');
    save(archivo6,'pos_maxs_segundo_blanco','-ascii','-double','-
tabs');
    save(archivo7,'pos_maxs_tercer_blanco','-ascii','-double','-
tabs');
    if generar == 1
    save(archivo8,'tresd_x','-ascii','-double','-tabs');
    save(archivo9,'tresd','-ascii','-double','-tabs');
    end
end
end

```

Simulación de tres blancos equidistantes, generando las gráficas de error en amplitud y distancia:

```

c = 3e8; %velocidad de la luz en m/s
% Inicializacion de constantes
Fm = 100000;
Fmin = 10e6;
Fmax = 40e6;
Df = Fmax - Fmin;
fs = Fmin;

Dd = c/(2*Df);

N = 2^15;
puntos_fft = 2^23;
generar = 0;
guardar = 1;
cargar = 1;

umbral = 0.8;

segundo_blanco = 50;%en metros
num_puntos = 400;%en pasos

dmax = 100;

Tm = 1/Fm;
t_resolucion = Tm/N;
resolucion = t_resolucion*c/2;
tmax = 2*dmax/c;
beta = 2*pi*Df/Tm;
% Calculo de la frecuencia de muestreo y numero total de muestras.
Fsample = 1/t_resolucion;
muestra_tmax = floor(tmax*Fsample);

muestra_segundo_blanco =
floor(2*segundo_blanco*Df*puntos_fft/c/Tm/Fsample) + 1;
muestras_retardo_una_muestra = Df*puntos_fft/Tm/(Fsample^2);

td = 2*segundo_blanco/c;
retardo_segundo_blanco = floor(Fsample*td);
if generar == 1
    tresd = [];

    % Implementacion de un periodo del coseno.
    fs_ficticio = fs-Df*tmax/2/Tm;
    %t = 0:1/Fsample:(2*Tm-1/Fsample);
    t = linspace(0,2*Tm-1/Fsample,2*N);
    x = cos(2*pi*fs_ficticio*t+beta*t.^2/2);

    %para el segundo blanco:
    %Calculo de los rebotes
    v0 = zeros(1,retardo_segundo_blanco);
    xr_segundo_blanco = [v0 x];
    xr_segundo_blanco = xr_segundo_blanco(muestra_tmax:muestra_tmax+N-
1);
    a = fft(xr_segundo_blanco .* x(muestra_tmax:muestra_tmax+N-
1),puntos_fft);

```

```

    maximo = max(abs(a));
    xr_segundo_blanco = xr_segundo_blanco./maximo;
else
    if cargar == 1
        principio_archivos = 'E:\PFC\Datos simulaciones\tres
blancos\buenos\';
        final_archivos = [' ', segundo_blanco ' '
sprintf('%d',segundo_blanco) 'm, blancos simétricos,'
sprintf('%d',Fmin/1e6) ' a ' sprintf('%d',Fmax/1e6) 'MHz,'
sprintf('%d',num_puntos)];
        final_archivos = [final_archivos 'puntos, Fm='
sprintf('%d',Fm/1e3) 'kHz,puntos_fft=2^'];
        final_archivos = [final_archivos
sprintf('%d',log2(puntos_fft)) ',N=2^' sprintf('%d',log2(N)) '.txt'];
        archivo_tresd = [principio_archivos 'tresd' final_archivos];
        archivo_tresd_x = [principio_archivos 'tresd_x'
final_archivos];
        est = 'loading'
        tresd = load(archivo_tresd);
        %tresd = tresd';
    end
end
posicion_real_segundo_blanco = retardo_segundo_blanco*c/Fsample/2;

%bucle principal
d = zeros(1,num_puntos);
maxs_primer_blanco = zeros(1,num_puntos);
pos_maxs_primer_blanco = zeros(1,num_puntos);
maxs_segundo_blanco = zeros(1,num_puntos);
pos_maxs_segundo_blanco = zeros(1,num_puntos);
maxs_tercer_blanco = zeros(1,num_puntos);
pos_maxs_tercer_blanco = zeros(1,num_puntos);
curvas_max_d = [];
for i = 0:num_puntos-1
%for i = 68:68
%    i=i+46; %#ok<FXSET>
    i %#ok<NOPTS>
        muestra_primer_blanco = muestra_segundo_blanco -
muestras_retardo_una_muestra*i;
        muestra_tercer_blanco = muestra_segundo_blanco +
muestras_retardo_una_muestra*i;
        mitad1 = floor((muestra_primer_blanco + muestra_segundo_blanco) /
2);
        mitad2 = muestra_segundo_blanco;
        d(i+1) = (i*c)/(Fsample*2);
        if generar == 1
            v0 = zeros(1,retardo_segundo_blanco - i);
            xr_primer_blanco = [v0 x];
            xr_primer_blanco = xr_primer_blanco(muestra_tmax:muestra_tmax+N-
1)./maximo;

            v0 = zeros(1,retardo_segundo_blanco + i);
            xr_tercer_blanco = [v0 x];
            xr_tercer_blanco = xr_tercer_blanco(muestra_tmax:muestra_tmax+N-
1)./maximo;

            % Suma de los diferentes periodos y adiccion de ruido
            xr = xr_primer_blanco + xr_segundo_blanco + xr_tercer_blanco;

            % Mezclado de la entrada y la salida
            r = x(muestra_tmax:muestra_tmax+N-1) .* xr;

```

```

R = fft(r,puntos_fft);
Prr = abs(R);
tresd = [tresd;Prr(1:2*mitad2)]; %#ok<AGROW>
else
    Pr = tresd(i+1,:);
end

%10 a 40
if i == 40
    umbral = 0.5;
end
if i == 80
    umbral = 0.8;
end

%% 50 a 80
%if i == 14
%    umbral = 0.3;
%end
%if i == 35
%    umbral = 0.5;
%end
%if i == 40
%    umbral = 0.8;
%end
%if i == 70
%    umbral = 0.5;
%end
%if i == 85
%    umbral = 0.65;
%end

%Metodo diferencial
dPrr = [Prr(1:2*mitad2) 0] - [0 Pr(1:2*mitad2)];
signo = sign(dPrr);
maximos_d = [];
estado = signo(1);
for k = 2 : 2*mitad2
    if signo(k) ~= estado
        %Estamos en un cero en k-1, veamos si es un maximo
        if Pr(k-1) > Pr(k)
            %Estamos en un maximo, veamos si supera el umbral
            if Pr(k-1) > umbral
                maximos_d = [maximos_d k-1]; %#ok<AGROW>
            end
        end
        estado = -estado;
    end
end
%maximos_d = [maximos_d zeros(1,3-size(maximos_d,2))]; %#ok<AGROW>
num_maximos = size(maximos_d,2);
if num_maximos > 3
    error('Numero de maximos mayor a 3, aumenta el umbral!!!');
end

if num_maximos == 0
    warning = 'Ningun máximo!!!' %#ok<NOPTS>
    maximos_d = [1 1 1]; %#ok<AGROW>
    num_maximos = 3;

```

```

end
if num_maximos == 1
    maximos_d = [maximos_d(1) maximos_d(1) maximos_d(1)];
%#ok<AGROW>
    num_maximos = 3;
end

if num_maximos == 2
    maximos_d = [maximos_d maximos_d(2)]; %#ok<AGROW>
    num_maximos = 3;
end

if num_maximos == 3
    maxs_primer_blanco(i+1) = Prr(maximos_d(1));
    pos_maxs_primer_blanco(i+1) = (maximos_d(1) -
muestra_primer_blanco)/puntos_fft*Fsample/Df*Tm*c/2/Dd;
    maxs_segundo_blanco(i+1) = Prr(maximos_d(2));
    pos_maxs_segundo_blanco(i+1) = (maximos_d(2) -
muestra_segundo_blanco)/puntos_fft*Fsample/Df*Tm*c/2/Dd;
    maxs_tercer_blanco(i+1) = Prr(maximos_d(3));
    pos_maxs_tercer_blanco(i+1) = (maximos_d(3) -
muestra_tercer_blanco)/puntos_fft*Fsample/Df*Tm*c/2/Dd;
end
curvas_max_d = [curvas_max_d;maximos_d]; %#ok<AGROW>
end

d = (0:num_puntos-1)/muestras_Dd;
mitad2_en_metros = (mitad2*Fsample*Tm*c)/(puntos_fft*Df*2);
d_normalizada = d./Dd;
tresd_x = linspace(0,2*mitad2_en_metros,2*mitad2);
figure(1);
plot(d_normalizada,maxs_primer_blanco,'LineWidth',2);
xlabel('Separaciones (unidades FWHM)','fontsize',20);
ylabel('Amplitud normalizada','fontsize',20);
title('Amplitudes normalizadas del primer blanco','fontsize',26);
grid;
figure(2);
plot(d_normalizada,maxs_segundo_blanco,'LineWidth',2);
xlabel('Separaciones (unidades FWHM)','fontsize',20);
ylabel('Amplitud normalizada','fontsize',20);
title('Amplitudes normalizadas del segundo blanco','fontsize',26);
grid;
figure(3);
plot(d_normalizada,maxs_tercer_blanco,'LineWidth',2);
xlabel('Separaciones (unidades FWHM)','fontsize',20);
ylabel('Amplitud normalizada','fontsize',20);
title('Amplitudes normalizadas del tercer blanco','fontsize',26);
grid;
figure(4);
plot(d_normalizada,pos_maxs_primer_blanco,'LineWidth',2);
xlabel('Separaciones (unidades FWHM)','fontsize',20);
ylabel('Error en separación (unidades FWHM)','fontsize',20);
title('Errores en distancia del primer blanco','fontsize',26);
grid;
figure(5);
plot(d_normalizada,pos_maxs_segundo_blanco,'LineWidth',2);
xlabel('Separaciones (unidades FWHM)','fontsize',20);
ylabel('Error en separación (unidades FWHM)','fontsize',20);
title('Errores en distancia del segundo blanco','fontsize',26);
grid;
figure(6);

```

```

plot(d_normalizada,pos_maxs_tercer_blanco,'LineWidth',2);
xlabel('Separaciones (unidades FWHM)','fontsize',20);
ylabel('Error en separación (unidades FWHM)','fontsize',20);
title('Errores en distancia del tercer blanco','fontsize',26);
grid;
figure(7);
plot(Prr);
plot(tresd_x,Prr(1:(2*mitad2)));
%figure(8);
%plot(tresd_x,dPrr(1:(2*mitad2)));
%surf(d_normalizada,tresd_x,tresd');
%xlabel('Separaciones (unidades FWHM)','fontsize',20);
%ylabel('Distancia (m)','fontsize',20);
%title('Evolución del los máximos 3D','fontsize',26);
figure(8);
plot(d_normalizada,linspace(segundo_blanco,muestra_primer_blanco/punto
s_fft*Fsample/Df*Tm*c/2,num_puntos),'-
',d_normalizada,segundo_blanco*ones(1,num_puntos),'-
',d_normalizada,linspace(segundo_blanco,muestra_tercer_blanco/puntos_f
ft*Fsample/Df*Tm*c/2,num_puntos),'-
',d_normalizada,curvas_max_d/puntos_fft*Fsample/Df*Tm*c/2,'x','LineWid
th',2);

xlabel('Separaciones (unidades FWHM)','fontsize',20);
ylabel('Distancia (m)','fontsize',20);
title('Evolución del los máximos','fontsize',26);
grid;

if guardar == 1
    principio_archivos = 'E:\PFC\Datos simulaciones\tres
blancos\buenos\';
    final_archivos = [' ', segundo_blanco a '
sprintf('%d',segundo_blanco) 'm, blancos simétricos,'
sprintf('%d',Fmin/1e6) ' a ' sprintf('%d',Fmax/1e6) 'MHz,'
sprintf('%d',num_puntos)];
    final_archivos = [final_archivos 'puntos, Fm='
sprintf('%d',Fm/1e3) 'kHz,puntos_fft=2^'];
    final_archivos = [final_archivos sprintf('%d',log2(puntos_fft))
',N=2^' sprintf('%d',log2(N)) '.txt'];
    archivo1 = [principio_archivos 'distancias_normalizadas'
final_archivos];
    archivo2 = [principio_archivos 'pot_primer_blanco'
final_archivos];
    archivo3 = [principio_archivos 'pot_segundo_blanco'
final_archivos];
    archivo4 = [principio_archivos 'pot_tercer_blanco'
final_archivos];
    archivo5 = [principio_archivos 'err_dist_primer_blanco'
final_archivos];
    archivo6 = [principio_archivos 'err_dist_segundo_blanco'
final_archivos];
    archivo7 = [principio_archivos 'err_dist_tercer_blanco'
final_archivos];
    archivo8 = [principio_archivos 'tresd_x' final_archivos];
    archivo9 = [principio_archivos 'tresd' final_archivos];

    save(archivo1,'d_normalizada','-ascii','-double','-tabs');
    save(archivo2,'maxs_primer_blanco','-ascii','-double','-tabs');
    save(archivo3,'maxs_segundo_blanco','-ascii','-double','-tabs');
    save(archivo4,'maxs_tercer_blanco','-ascii','-double','-tabs');

```



```
save(archivo5, 'pos_maxs_primer_blanco', '-ascii', '-double', '-  
tabs');  
save(archivo6, 'pos_maxs_segundo_blanco', '-ascii', '-double', '-  
tabs');  
save(archivo7, 'pos_maxs_tercer_blanco', '-ascii', '-double', '-  
tabs');  
if generar == 1  
save(archivo8, 'tresd_x', '-ascii', '-double', '-tabs');  
save(archivo9, 'tresd', '-ascii', '-double', '-tabs');  
end  
end
```

Programa de lectura de datos de laboratorio. También para blanco disperso:

```
% Constantes
c = 3e8;

% Opciones
Fsample = 250e3; %frecuencia de muestreo de los datos
puntos_fft = 2^13; %número de puntos de la fft
%directorio que contiene los datos
principio_archivo = 'E:\PFC\Datos\Medidas de 1 blanco\algunas medidas
con un blanco\';
%archivo de datos 1 (normalmente señal)
final_archivo = '20 a 80MHz, 100 periodos, lms.txt';
%archivo de datos 2 (normalmente background)
final_archivo_background = '10 a 40MHz, 100 periodos, lms.txt';

Fmin = 20e6; %frecuencia inicial en Hz
Fmax = 80e6; %frecuencia final en Hz
solo_background = 0; %0: representa la señal; 1: representa el
background
resta = 0; % 1: representa la resta (señal - background)
normaliza = 1;% 1: normaliza respecto del máximo

% Opciones de representación
estilo_grafica = 'b-';
pos_min = 1;
pos_max = 100;

% Inicializaciones
Df = Fmax - Fmin;
archivo_datos = [principio_archivo final_archivo];
archivo_datos_background = [principio_archivo
final_archivo_background];

% Lectura de archivos
datos = load(archivo_datos)';
datos_background = load(archivo_datos_background)';

[N,num_T] = size(datos);
Tm = N/Fsample;
dmax = c*Fsample*Tm/Df/2;
pos_max = floor(pos_max*puntos_fft/dmax) + 1;
pos_min = floor(pos_min*puntos_fft/dmax) + 1;

if resta == 1
    datos = datos - datos_background;
else
    if solo_background == 1
        datos = datos_background;
    end
end

d = linspace(0,dmax,puntos_fft);

% Suma de los periodos de la señal
resultado = datos * ones(num_T,1);

% Resta de la media
```

```
media = ones(1,N) * resultado/N;
resultado = resultado - media;

% Cálculo de la fft
R = fft(resultado,puntos_fft);

%Normalización
if normaliza == 1
    R = R/max(abs(R));
end

P = abs(R);
%P = P./max(P);

% Presentación
hold on;
plot(d(pos_min:pos_max),P(pos_min:pos_max),estilo_grafica,'LineWidth',
2); % modulo
%plot(d(pos_min:pos_max),angle(R(pos_min:pos_max)),estilo_grafica,'Lin
eWidth',2); % fase
hold off;

xlabel('Distancia (m)','fontsize',20);
if normaliza == 1
    ylabel('Amplitud normalizada','fontsize',20);
else
    ylabel('Amplitud','fontsize',20);
end
grid;
```

## Estimación del eventanado para medidas de laboratorio:

```
% Constantes
c = 3e8;
% Opciones
Fsample = 250e3; %frecuencia de muestreo de los datos
puntos_fft = 2^13; %número de puntos de la fft
%directorio que contiene los datos
principio_archivo = 'E:\PFC\Datos\Medidas de 1 blanco\algunas medidas
con un blanco\';
%archivo de datos 1 (normalmente señal)
final_archivo = '20 a 80MHz, 100 periodos, lms.txt';
%archivo de datos 2 (normalmente background)
final_archivo_background = '10 a 40MHz, 100 periodos, lms.txt';

Fmin = 20e6; %frecuencia inicial en Hz
Fmax = 80e6; %frecuencia final en Hz
solo_background = 0; %0: representa la señal; 1: representa el
background
resta = 0; % 1: representa la resta (señal - background)
normaliza = 0;% 1: normaliza respecto del máximo

% Opciones de representación
estilo_grafica = 'b-';
pos_min = 1;
pos_max = 100;

% Inicializaciones
Df = Fmax - Fmin;
archivo_datos = [principio_archivo final_archivo];
archivo_datos_background = [principio_archivo
final_archivo_background];

% Lectura de archivos
datos = load(archivo_datos)';
datos_background = load(archivo_datos_background)';

[N,num_T] = size(datos);
Tm = N/Fsample;
dmax = c*Fsample*Tm/Df/2;
pos_max = floor(pos_max*puntos_fft/dmax) + 1;
pos_min = floor(pos_min*puntos_fft/dmax) + 1;

if resta == 1
    datos = datos - datos_background;
else
    if solo_background == 1
        datos = datos_background;
    end
end

d = linspace(0,dmax,puntos_fft);

% Suma de los periodos de la señal
resultado = datos * ones(num_T,1);

% Resta de la media
media = ones(1,N) * resultado/N;
```

```

resultado = resultado - media;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%ENVENTANADO%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

N = size(resultado,1);
signo = sign(resultado);
estado = signo(1);
pos_ceros = 1;
for i = 1 : N
    if signo(i) ~= estado
        pos_ceros = [pos_ceros i]; %#ok<AGROW>
        estado = -estado;
    end
end
estado = signo(1);
maximos = [];
pos_max = [];
minimos = [];
pos_min = [];
for i = 1:size(pos_ceros,2)-1
    if estado == 1
        [valor,posicion] =
max(resultado(pos_ceros(i):pos_ceros(i+1)));
        pos_max = [pos_max posicion+pos_ceros(i)-1];%#ok<AGROW>
        maximos = [maximos valor];%#ok<AGROW>
    else
        [valor,posicion] =
min(resultado(pos_ceros(i):pos_ceros(i+1)));
        pos_min = [pos_min posicion+pos_ceros(i)-1];%#ok<AGROW>
        minimos = [minimos valor];%#ok<AGROW>
    end
    estado = -estado;
end
maximos = [maximos(1) maximos maximos(size(maximos,2))];
pos_max = [1 pos_max N];
minimos = [minimos(1) minimos minimos(size(minimos,2))];
pos_min = [1 pos_min N];
curva_maximos = [];
curva_minimos = [];
for i = 1:size(pos_max,2)-1
    trozo = linspace(maximos(i),maximos(i+1),pos_max(i+1)-
pos_max(i));
    curva_maximos = [curva_maximos trozo];%#ok<AGROW>
end
curva_maximos = [curva_maximos maximos(size(maximos,2))];
for i = 1:size(pos_min,2)-1
    trozo = linspace(minimos(i),minimos(i+1),pos_min(i+1)-
pos_min(i));
    curva_minimos = [curva_minimos trozo];%#ok<AGROW>
end
curva_minimos = [curva_minimos minimos(size(minimos,2))];
t = 1:N;
curva_minimos = curva_minimos(1:N);
ventana = (curva_maximos-curva_minimos)/2;
ventana = ventana./max(ventana);
ventana = ventana';
figure(1);

```

```
plot(t, resultado, t, curva_maximos, t, curva_minimos, t, ventana.*max(resultado), 'LineWidth', 2);
figure(2);
plot(t, resultado, t, resultado./ventana, 'LineWidth', 2);
figure(3);
R1 = fft(resultado, puntos_fft);
R2 = fft(resultado./ventana, puntos_fft);
P1 = abs(R1);
P2 = abs(R2);
plot(d, P1./max(P1), estilo_grafica, d, P2./max(P2), 'r-', 'LineWidth', 2);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%FIN ENVENTANADO%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

## Comparación de medidas de laboratorio con simulaciones para dos blancos:

```

%NOTA: La numeración de los blancos es según se los encuentra el
laser.
%El primer blanco es el primer cristal que se encuentra, el segundo el
%segundo cristal, y el tercero el espejo

%Constantes
c = 3e8;
Fsample = 250e3;
puntos_fft = 2^20;
directorio_datos = 'E:\PFC\Datos\Medidas de 2 blancos\2 blancos varias
medias varias frecuencias\';
umbral = 0.6;%largos
%umbral = 0.8;
extension = 'txt';

principio_archivo = 'E:\PFC\Datos simulaciones\dos blancos\buenos\';
final_archivo = ', segundo blanco a 50m, 50 a 80MHz,400puntos,
Fm=100kHz,puntos_fft=2^23,N=2^15.txt';

%50 a 80 => 5.480410, 5.508000e+001
%20 a 80 => 5.485090

sep_norm_sim = load([principio_archivo 'distancias_normalizadas'
final_archivo]);
maxs_primer_blanco_sim = load([principio_archivo 'pot_primer_blanco'
final_archivo]);
maxs_segundo_blanco_sim = load([principio_archivo 'pot_segundo_blanco'
final_archivo]);
err_dist_primer_blanco_sim = load([principio_archivo
'err_dist_primer_blanco' final_archivo]);
err_dist_segundo_blanco_sim = load([principio_archivo
'err_dist_segundo_blanco' final_archivo]);

%Opciones
Fm = 1000;
Fmin = 50e6;
Fmax = 80e6;
pos_min = 40;
pos_max = 60;
distancias = ['0,5';'1,2';'2  '; '3,6';'4,4';'5,5';'6,5';'7,5';'15  '];
%#ok<NBRAK> cortos
%distancias = ['0,3  '; '0,6  '; '1    '; '1,5  '; '2    '; '2,5  '; '3    '; '3,5
'; '4    '; '4,5  '; '5    '; '5,5  '; '11,8'; '15  ']; %#ok<NBRAK> 2 blancos
largo

%Inicializaciones
Tm = 1/Fm;
Df = Fmax - Fmin;
Dd = c/(2*Df);
dmax = c*Fsample*Tm/Df/2;

%Al lio
num_puntos = size(distancias,1);
separaciones = zeros(1,num_puntos);
maxs_1_blanco = zeros(1,num_puntos);
pos_maxs_1_blanco = zeros(1,num_puntos);
maxs_2_blanco = zeros(1,num_puntos);

```

```

pos_maxs_2_blanco = zeros(1,num_puntos);
err_amp_1 = zeros(1,num_puntos);
err_amp_2 = zeros(1,num_puntos);
grafica_1_blanco = [];
grafica_2_blanco = [];
grafica_total = [];
grafica_suma = [];

for i = 1:num_puntos
    dist2 = distancias(i,:);

    %filtramos espacios
    dist = [];
    for k = 1:size(dist2,2)
        if dist2(k) ~= ' '
            dist = [dist dist2(k)]; %#ok<AGROW>
        end
    end

    archivo_1_blanco = [directorio_datos dist ', '
    sprintf('%d',Fmin/1e6) ' a ' sprintf('%d',Fmax/1e6) ' MHz, blanco 1
ON, blanco 2 OFF, blanco 3 OFF.' extension];
    archivo_2_blanco = [directorio_datos dist ', '
    sprintf('%d',Fmin/1e6) ' a ' sprintf('%d',Fmax/1e6) ' MHz, blanco 1
OFF, blanco 2 ON, blanco 3 OFF.' extension];
    archivo_total = [directorio_datos dist ', ' sprintf('%d',Fmin/1e6)
' a ' sprintf('%d',Fmax/1e6) ' MHz, blanco 1 ON, blanco 2 ON, blanco 3
OFF.' extension];
%
    datos_1_blanco = load(archivo_1_blanco)';
    if isstruct(datos_1_blanco)
        datos_1_blanco = datos_1_blanco.blanco_1;
    end
    datos_2_blanco = load(archivo_2_blanco)';
    if isstruct(datos_2_blanco)
        datos_2_blanco = datos_2_blanco.blanco_2;
    end
    datos_total = load(archivo_total)';
    if isstruct(datos_total)
        datos_total = datos_total.suma_total;
    end

    [N,num_T] = size(datos_total);

    Tm = N/Fsample;

    resultado_1_blanco = datos_1_blanco * ones(num_T,1);
    R_1_blanco = fft(resultado_1_blanco,puntos_fft);
    P_1_blanco = abs(R_1_blanco);

    [max_1_blanco pos_max_1_blanco] = max(P_1_blanco(1:puntos_fft/2));

    resultado_2_blanco = datos_2_blanco * ones(num_T,1);
    R_2_blanco = fft(resultado_2_blanco,puntos_fft);
    P_2_blanco = abs(R_2_blanco);

    [max_2_blanco pos_max_2_blanco] = max(P_2_blanco);

    normalizacion = (max_1_blanco + max_2_blanco)/2;

```



```

R_1_blanco = R_1_blanco./normalizacion;
P_1_blanco = abs(R_1_blanco);

R_2_blanco = R_2_blanco./normalizacion;
P_2_blanco = abs(R_2_blanco);

err_amp_1(i) = -(normalizacion - max_1_blanco)/normalizacion;
err_amp_2(i) = -(normalizacion - max_2_blanco)/normalizacion;

separacion = pos_max_2_blanco - pos_max_1_blanco;
separaciones(i) = separacion*dmax/puntos_fft %#ok<NOPTS>
if i == 1
    mitad = (pos_max_2_blanco + pos_max_1_blanco)/2;
end
mitad_en_metros = mitad*dmax/puntos_fft;
grafica_1_blanco = [grafica_1_blanco';P_1_blanco(1:2*mitad)'];
%#ok<AGROW>
grafica_2_blanco = [grafica_2_blanco';P_2_blanco(1:2*mitad)'];
%#ok<AGROW>
R_suma = R_1_blanco + R_2_blanco;
P_suma = abs(R_suma);
grafica_suma = [grafica_suma';P_suma(1:2*mitad)']; %#ok<AGROW>

resultado_total = datos_total * ones(num_T,1);
R_total = fft(resultado_total,puntos_fft)./normalizacion;
P_total = abs(R_total);
%P_total = P_suma;
grafica_total = [grafica_total';P_total(1:2*mitad)']; %#ok<AGROW>

amp_norm_1_blanco = max_1_blanco / normalizacion;
amp_norm_2_blanco = max_2_blanco / normalizacion;

%%%% largos
%%50 a 80
%if i == 3
%    umbral = 0.2;
%end
%if i == 4
%    umbral = 0.6;
%end
%%-15 a 15
%if i == 7
%    umbral = 0.5;
%end

%%%% cortos
%50 a 80
if i == 2
    umbral = 0.2;
end
if i == 3
    umbral = 0.6;
end
%10 a 40
%if i == 4
%    umbral = 0.6;

```

```

%end

%Busqueda de maximos
dPrr = [P_total(1:2*mitad)' 0] - [0 P_total(1:2*mitad)'];
signo = sign(dPrr);
maximos_d = [];
estado = signo(1);
for k = 2 : 2*mitad
    if signo(k) ~= estado
        %Estamos en un cero en k-1, veamos si es un maximo
        if P_total(k-1) > P_total(k)
            %Estamos en un maximo, veamos si supera el umbral
            if P_total(k-1) > umbral
                maximos_d = [maximos_d k-1]; %#ok<AGROW>
            end
        end
        estado = -estado;
    end
end
num_maximos = size(maximos_d,2);
if num_maximos > 2
    error('Numero de maximos mayor a 2, aumenta el umbral!!!');
end

if num_maximos == 0
    warning = 'Ningun máximo!!!'
    maximos_d = [1 1]; %#ok<AGROW>
    num_maximos = 2;
end

if num_maximos == 1
    %error('Numero de maximos igual a 2. ');
    maximos_d = [maximos_d(1) maximos_d(1)]; %#ok<UNRCH,AGROW>
    num_maximos = 2;
end

if num_maximos == 2
    maxs_1_blanco(i) = P_total(maximos_d(1))/amp_norm_1_blanco;
    pos_maxs_1_blanco(i) = (maximos_d(1) -
pos_max_1_blanco)*dmax/puntos_fft;
    maxs_2_blanco(i) = P_total(maximos_d(2))/amp_norm_2_blanco;
    pos_maxs_2_blanco(i) = (maximos_d(2) -
pos_max_2_blanco)*dmax/puntos_fft;
end

end

d = linspace(0,dmax,puntos_fft);

%CORRECCIÓN%

corr_actual_por_defecto = 0.4;
error_max_por_defecto = 1000;

paso = 0.0001;
corr_max = 2;
corr_actual = corr_actual_por_defecto;

```

```
corr_ant = corr_actual_por_defecto;
error = error_max_por_defecto;
error_ant = error_max_por_defecto + 1;

while error < error_ant
    corr_ant = corr_actual;
    corr_actual = corr_actual + paso;
    error_ant = error;
    error = 0;
    for k = 1 : size(maxs_1_blanco,2)
        valor_sim =
maxs_primer_blanco_sim(round((separaciones(k)/Dd)/sep_norm_sim(size(se
p_norm_sim,2))*size(sep_norm_sim,2))+1);
        valor_lab = maxs_1_blanco(k);
        error = error + (corr_actual*valor_lab - valor_sim)^2;
    end
end
correccion1 = corr_ant;

corr_actual = corr_actual_por_defecto;
corr_ant = corr_actual_por_defecto;
error = error_max_por_defecto;
error_ant = error_max_por_defecto + 1;

while error < error_ant
    corr_ant = corr_actual;
    corr_actual = corr_actual + paso;
    error_ant = error;
    error = 0;
    for k = 1 : size(maxs_2_blanco,2)
        valor_sim =
maxs_segundo_blanco_sim(round((separaciones(k)/Dd)/sep_norm_sim(size(s
ep_norm_sim,2))*size(sep_norm_sim,2))+1);
        valor_lab = maxs_2_blanco(k);
        error = error + (corr_actual*valor_lab - valor_sim)^2;
    end
end
correccion2 = corr_ant;

corr_actual = corr_actual_por_defecto;
corr_ant = corr_actual_por_defecto;
error = error_max_por_defecto;
error_ant = error_max_por_defecto + 1;

while error < error_ant
    corr_ant = corr_actual;
    corr_actual = corr_actual + paso;
    error_ant = error;
    error = 0;
    for k = 1 : size(maxs_1_blanco,2)
        valor_sim =
maxs_primer_blanco_sim(round((separaciones(k)/Dd)/sep_norm_sim(size(se
p_norm_sim,2))*size(sep_norm_sim,2))+1);
        valor_lab = maxs_1_blanco(k);
        error = error + (corr_actual*valor_lab - valor_sim)^2;
        valor_sim =
maxs_segundo_blanco_sim(round((separaciones(k)/Dd)/sep_norm_sim(size(s
ep_norm_sim,2))*size(sep_norm_sim,2))+1);
        valor_lab = maxs_2_blanco(k);
        error = error + (corr_actual*valor_lab - valor_sim)^2;
```

```

    end
end
correccion4 = corr_ant;

correccion4
media = (correccion1 + correccion2)/2

maxs_1_blanco2 = maxs_1_blanco * correccion1;
maxs_2_blanco2 = maxs_2_blanco * correccion2;

%%%%%%%%%%%%%%

figure(1);
%plot(separaciones./Dd,maxs_1_blanco,'x',separaciones./Dd,maxs_1_blanco2,'x',sep_norm_sim,maxs_primer_blanco_sim,'MarkerSize',15,'LineWidth',2);
plot(separaciones./Dd,maxs_1_blanco,'x',sep_norm_sim,maxs_primer_blanco_sim,'MarkerSize',15,'LineWidth',2);
xlabel('Separaciones (unidades FWHM)','fontsize',20);
ylabel('Amplitud normalizada','fontsize',20);
%title(['Amplitudes normalizadas del primer blanco. Corrección: ' sprintf('%0.3g',correccion1)],'fontsize',26);
title('Amplitudes normalizadas del primer blanco.','fontsize',26);
grid;
figure(2);
%plot(separaciones./Dd,maxs_2_blanco,'x',separaciones./Dd,maxs_2_blanco2,'x',sep_norm_sim,maxs_segundo_blanco_sim,'MarkerSize',15,'LineWidth',2);
plot(separaciones./Dd,maxs_2_blanco,'x',sep_norm_sim,maxs_segundo_blanco_sim,'MarkerSize',15,'LineWidth',2);
xlabel('Separaciones (unidades FWHM)','fontsize',20);
ylabel('Amplitud normalizada','fontsize',20);
%title(['Amplitudes normalizadas del segundo blanco. Corrección: ' sprintf('%0.3g',correccion2)],'fontsize',26);
title('Amplitudes normalizadas del segundo blanco','fontsize',26);
grid;
figure(3);
%plot(separaciones./Dd,pos_maxs_cristal./Dd,'x',sep_norm_sim,err_dist_primer_blanco_sim,'MarkerSize',15);
plot(separaciones./Dd,pos_maxs_1_blanco./Dd,'x',sep_norm_sim,err_dist_primer_blanco_sim,'MarkerSize',15,'LineWidth',2);
xlabel('Separaciones (unidades FWHM)','fontsize',20);
ylabel('Error en separación (unidades FWHM)','fontsize',20);
title('Errores en distancia del primer blanco','fontsize',26);
grid;
figure(4);
%plot(separaciones./Dd,pos_maxs_espejo./Dd,'x',sep_norm_sim,err_dist_segundo_blanco_sim,'MarkerSize',15);
plot(separaciones./Dd,pos_maxs_2_blanco./Dd,'x',sep_norm_sim,err_dist_segundo_blanco_sim,'MarkerSize',15,'LineWidth',2);
xlabel('Separaciones (unidades FWHM)','fontsize',20);
ylabel('Error en separación (unidades FWHM)','fontsize',20);
title('Errores en distancia del segundo blanco','fontsize',26);
grid;
for i = 1:num_puntos
    figure(i+4);
    P_1_blanco = grafica_1_blanco(:,i);
    P_2_blanco = grafica_2_blanco(:,i);
    P_total = grafica_total(:,i);

```

```
P_suma = grafica_suma(:,i);

plot(d(1:2*mitad),P_1_blanco(1:2*mitad),d(1:2*mitad),P_2_blanco(1:2*mitad),d(1:2*mitad),P_total(1:2*mitad),d(1:2*mitad),P_suma(1:2*mitad),'-k','LineWidth',2);
xlabel('Distancia (m)','fontsize',20);
ylabel('Amplitud Normalizada','fontsize',20);
dist2 = distancias(i,:);

%filtramos espacios
dist = [];
for k = 1:size(dist2,2)
    if dist2(k) ~= ' '
        dist = [dist dist2(k)]; %#ok<AGROW>
    end
end
title([dist '(' sprintf('%0.3g',separaciones(i)) 'm). Errores: '
sprintf('%0.3g',err_amp_1(i)*100) '%, '
sprintf('%0.3g',err_amp_2(i)*100) '%'],'fontsize',26);
grid;
end
%figure(1);
%plot(d(1:2*mitad),P_1_blanco(1:2*mitad),d(1:2*mitad),P_2_blanco(1:2*mitad),d(1:2*mitad),P_3_blanco(1:2*mitad),d(1:2*mitad),P_total(1:2*mitad));
```

## Comparación de medidas de laboratorio con simulaciones para tres blancos:

```

%NOTA: La numeración de los blancos es según se los encuentra el
laser.
%El primer blanco es el primer cristal que se encuentra, el segundo el
%segundo cristal, y el tercero el espejo

%Constantes
c = 3e8;
Fsample = 250e3;
puntos_fft = 2^20;
directorio_datos = 'E:\PFC\Datos\Medidas de 3 blancos\3 blancos,
buenos\';
umbral = 0.4;
extension = 'txt';

principio_archivo = 'E:\PFC\Datos simulaciones\tres blancos\buenos\';
final_archivo = ', segundo blanco a 50m, tercer blanco a 55m, 20 a
80MHz, 150puntos, Fm=100kHz, puntos_fft=2^23, N=2^15.txt';

sep_norm_sim = load([principio_archivo 'distancias_normalizadas'
final_archivo]);
maxs_primer_blanco_sim = load([principio_archivo 'pot_primer_blanco'
final_archivo]);
maxs_segundo_blanco_sim = load([principio_archivo 'pot_segundo_blanco'
final_archivo]);
maxs_tercer_blanco_sim = load([principio_archivo 'pot_tercer_blanco'
final_archivo]);
err_dist_primer_blanco_sim = load([principio_archivo
'err_dist_primer_blanco' final_archivo]);
err_dist_segundo_blanco_sim = load([principio_archivo
'err_dist_segundo_blanco' final_archivo]);
err_dist_tercer_blanco_sim = load([principio_archivo
'err_dist_tercer_blanco' final_archivo]);

%Opciones
Fm = 1000;
Fmin = 20e6;
Fmax = 80e6;
pos_min = 40;
pos_max = 60;
%distancias = ['0.2'; '0.3'; '0.5'; '1.2'; '2.2'; '3
'; '3.7'; '4.5'; '5.5'; '6.3']; 3 blancos septiembre
%distancias = ['0,2'; '0,5'; '1,5'; '2
'; '3,5
'; '4,5bis'; '4,25bis'; '5
'; '6,3
']; %#ok<NBRAK>3 blancos
septiembre
distancias = ['0,3'; '1,5'; '2,5'; '3
'; '4,5']; %#ok<NBRAK>3 blancos,
buenos

%Inicializaciones
Tm = 1/Fm;
Df = Fmax - Fmin;
Dd = c/(2*Df);
dmax = c*Fsample*Tm/Df/2;

%Al lio
num_puntos = size(distancias,1);
separaciones = zeros(1,num_puntos);
separaciones2 = zeros(1,num_puntos);

```

```

maxs_1_blanco = zeros(1,num_puntos);
pos_maxs_1_blanco = zeros(1,num_puntos);
maxs_2_blanco = zeros(1,num_puntos);
pos_maxs_2_blanco = zeros(1,num_puntos);
maxs_3_blanco = zeros(1,num_puntos);
pos_maxs_3_blanco = zeros(1,num_puntos);
err_amp_1 = zeros(1,num_puntos);
err_amp_2 = zeros(1,num_puntos);
err_amp_3 = zeros(1,num_puntos);
grafica_1_blanco = [];
grafica_2_blanco = [];
grafica_3_blanco = [];
grafica_total = [];
grafica_suma = [];

for i = 1:num_puntos
    dist2 = distancias(i,:);

    %filtramos espacios
    dist = [];
    for k = 1:size(dist2,2)
        if dist2(k) ~= ' '
            dist = [dist dist2(k)]; %#ok<AGROW>
        end
    end

    archivo_1_blanco = [directorio_datos dist ' '
sprintf('%d',Fmin/1e6) ' a ' sprintf('%d',Fmax/1e6) ' MHz, blanco 1
ON, blanco 2 OFF, blanco 3 OFF.' extension];
    archivo_2_blanco = [directorio_datos dist ' '
sprintf('%d',Fmin/1e6) ' a ' sprintf('%d',Fmax/1e6) ' MHz, blanco 1
OFF, blanco 2 ON, blanco 3 OFF.' extension];
    archivo_3_blanco = [directorio_datos dist ' '
sprintf('%d',Fmin/1e6) ' a ' sprintf('%d',Fmax/1e6) ' MHz, blanco 1
OFF, blanco 2 OFF, blanco 3 ON.' extension];
    archivo_total = [directorio_datos dist ' ' sprintf('%d',Fmin/1e6)
' a ' sprintf('%d',Fmax/1e6) ' MHz, blanco 1 ON, blanco 2 ON, blanco 3
ON.' extension];
%
    datos_1_blanco = load(archivo_1_blanco)';
    if isstruct(datos_1_blanco)
        datos_1_blanco = datos_1_blanco.blanco_1;
    end
    datos_2_blanco = load(archivo_2_blanco)';
    if isstruct(datos_2_blanco)
        datos_2_blanco = datos_2_blanco.blanco_2;
    end
    datos_3_blanco = load(archivo_3_blanco)';
    if isstruct(datos_3_blanco)
        datos_3_blanco = datos_3_blanco.blanco_3;
    end
    datos_total = load(archivo_total)';
    if isstruct(datos_total)
        datos_total = datos_total.suma_total;
    end

    [N,num_T] = size(datos_total);

    Tm = N/Fsample;

```

```

resultado_1_blanco = datos_1_blanco * ones(num_T,1);
R_1_blanco = fft(resultado_1_blanco,puntos_fft);
P_1_blanco = abs(R_1_blanco);

[max_1_blanco pos_max_1_blanco] = max(P_1_blanco(1:puntos_fft/2));

resultado_2_blanco = datos_2_blanco * ones(num_T,1);
R_2_blanco = fft(resultado_2_blanco,puntos_fft);
P_2_blanco = abs(R_2_blanco);

[max_2_blanco pos_max_2_blanco] = max(P_2_blanco);

resultado_3_blanco = datos_3_blanco * ones(num_T,1);
R_3_blanco = fft(resultado_3_blanco,puntos_fft);
P_3_blanco = abs(R_3_blanco);

[max_3_blanco pos_max_3_blanco] = max(P_3_blanco);

normalizacion = (max_1_blanco + max_2_blanco + max_3_blanco)/3;

R_1_blanco = R_1_blanco./normalizacion;
P_1_blanco = abs(R_1_blanco);

R_2_blanco = R_2_blanco./normalizacion;
P_2_blanco = abs(R_2_blanco);

R_3_blanco = R_3_blanco./normalizacion;
P_3_blanco = abs(R_3_blanco);

err_amp_1(i) = -(normalizacion - max_1_blanco)/normalizacion;
err_amp_2(i) = -(normalizacion - max_2_blanco)/normalizacion;
err_amp_3(i) = -(normalizacion - max_3_blanco)/normalizacion;

separacion = pos_max_2_blanco - pos_max_1_blanco;
separacion2 = pos_max_3_blanco - pos_max_2_blanco;
separaciones(i) = separacion*dmax/puntos_fft %#ok<NOPTS>
separaciones2(i) = separacion2*dmax/puntos_fft %#ok<NOPTS>
if i == 1
    mitad = pos_max_2_blanco;
end
mitad_en_metros = mitad*dmax/puntos_fft;
grafica_1_blanco = [grafica_1_blanco';P_1_blanco(1:2*mitad)'];
%#ok<AGROW>
grafica_2_blanco = [grafica_2_blanco';P_2_blanco(1:2*mitad)'];
%#ok<AGROW>
grafica_3_blanco = [grafica_3_blanco';P_3_blanco(1:2*mitad)'];
%#ok<AGROW>

R_suma = R_1_blanco + R_2_blanco + R_3_blanco;
P_suma = abs(R_suma);
grafica_suma = [grafica_suma';P_suma(1:2*mitad)']; %#ok<AGROW>

resultado_total = datos_total * ones(num_T,1);
R_total = fft(resultado_total,puntos_fft)./normalizacion;
P_total = abs(R_total);
grafica_total = [grafica_total';P_total(1:2*mitad)']; %#ok<AGROW>

```



```

amp_norm_1_blanco = max_1_blanco / normalizacion;
amp_norm_2_blanco = max_2_blanco / normalizacion;
amp_norm_3_blanco = max_3_blanco / normalizacion;

%Busqueda de maximos
dPrr = [P_total(1:2*mitad)' 0] - [0 P_total(1:2*mitad)'];
signo = sign(dPrr);
maximos_d = [];
estado = signo(1);
for k = 2 : 2*mitad
    if signo(k) ~= estado
        %Estamos en un cero en k-1, veamos si es un maximo
        if P_total(k-1) > P_total(k)
            %Estamos en un maximo, veamos si supera el umbral
            if P_total(k-1) > umbral
                maximos_d = [maximos_d k-1]; %#ok<AGROW>
            end
        end
        estado = -estado;
    end
end
%maximos_d = [maximos_d zeros(1,3-size(maximos_d,2))]; %#ok<AGROW>
num_maximos = size(maximos_d,2);
if num_maximos > 3
    error('Numero de maximos mayor a 3, aumenta el umbral!!');
end

if num_maximos == 1
    %error('Numero de maximos igual a 2. ');
    maximos_d = [maximos_d(1) maximos_d(1) maximos_d(1)];
%#ok<UNRCH,AGROW>
    num_maximos = 3;
end

if num_maximos == 2
    %error('Numero de maximos igual a 2. ');
    maximos_d = [maximos_d(1) maximos_d]; %#ok<UNRCH,AGROW>
    num_maximos = 3;
end

if num_maximos == 3
    maxs_1_blanco(i) = P_total(maximos_d(1))/amp_norm_1_blanco;
    pos_maxs_1_blanco(i) = (maximos_d(1) -
pos_max_1_blanco)*dmax/puntos_fft;
    maxs_2_blanco(i) = P_total(maximos_d(2))/amp_norm_2_blanco;
    pos_maxs_2_blanco(i) = (maximos_d(2) -
pos_max_2_blanco)*dmax/puntos_fft;
    maxs_3_blanco(i) = P_total(maximos_d(3))/amp_norm_3_blanco;
    pos_maxs_3_blanco(i) = (maximos_d(3) -
pos_max_3_blanco)*dmax/puntos_fft;
end

end

d = linspace(0,dmax,puntos_fft);

%CORRECCIÓN%

```

```

corr_actual_por_defecto = 0.7;
error_max_por_defecto = 1000;

paso = 0.0001;
corr_max = 2;
corr_actual = corr_actual_por_defecto;
corr_ant = corr_actual_por_defecto;
error = error_max_por_defecto;
error_ant = error_max_por_defecto + 1;

while error < error_ant
    corr_ant = corr_actual;
    corr_actual = corr_actual + paso;
    error_ant = error;
    error = 0;
    for k = 1 : size(maxs_1_blanco,2)
        valor_sim =
maxs_primer_blanco_sim(round((separaciones(k)/Dd)/sep_norm_sim(size(se
p_norm_sim,2))*size(sep_norm_sim,2))+1);
        valor_lab = maxs_1_blanco(k);
        error = error + (corr_actual*valor_lab - valor_sim)^2;
    end
end
correccion1 = corr_ant;

corr_actual = corr_actual_por_defecto;
corr_ant = corr_actual_por_defecto;
error = error_max_por_defecto;
error_ant = error_max_por_defecto + 1;

while error < error_ant
    corr_ant = corr_actual;
    corr_actual = corr_actual + paso;
    error_ant = error;
    error = 0;
    for k = 1 : size(maxs_2_blanco,2)
        valor_sim =
maxs_segundo_blanco_sim(round((separaciones(k)/Dd)/sep_norm_sim(size(s
ep_norm_sim,2))*size(sep_norm_sim,2))+1);
        valor_lab = maxs_2_blanco(k);
        error = error + (corr_actual*valor_lab - valor_sim)^2;
    end
end
correccion2 = corr_ant;

corr_actual = corr_actual_por_defecto;
corr_ant = corr_actual_por_defecto;
error = error_max_por_defecto;
error_ant = error_max_por_defecto + 1;

while error < error_ant
    corr_ant = corr_actual;
    corr_actual = corr_actual + paso;
    error_ant = error;
    error = 0;
    for k = 1 : size(maxs_3_blanco,2)
        valor_sim =
maxs_tercer_blanco_sim(round((separaciones(k)/Dd)/sep_norm_sim(size(se
p_norm_sim,2))*size(sep_norm_sim,2))+1);

```

```

        valor_lab = maxs_3_blanco(k);
        error = error + (corr_actual*valor_lab - valor_sim)^2;
    end
end
correccion3 = corr_ant;

corr_actual = corr_actual_por_defecto;
corr_ant = corr_actual_por_defecto;
error = error_max_por_defecto;
error_ant = error_max_por_defecto + 1;

while error < error_ant
    corr_ant = corr_actual;
    corr_actual = corr_actual + paso;
    error_ant = error;
    error = 0;
    for k = 1 : size(maxs_1_blanco,2)
        valor_sim =
maxs_primer_blanco_sim(round((separaciones(k)/Dd)/sep_norm_sim(size(se
p_norm_sim,2))*size(sep_norm_sim,2))+1);
        valor_lab = maxs_1_blanco(k);
        error = error + (corr_actual*valor_lab - valor_sim)^2;
        valor_sim =
maxs_segundo_blanco_sim(round((separaciones(k)/Dd)/sep_norm_sim(size(s
ep_norm_sim,2))*size(sep_norm_sim,2))+1);
        valor_lab = maxs_2_blanco(k);
        error = error + (corr_actual*valor_lab - valor_sim)^2;
        valor_sim =
maxs_tercer_blanco_sim(round((separaciones(k)/Dd)/sep_norm_sim(size(se
p_norm_sim,2))*size(sep_norm_sim,2))+1);
        valor_lab = maxs_3_blanco(k);
        error = error + (corr_actual*valor_lab - valor_sim)^2;
    end
end
correccion4 = corr_ant;

correccion4
media = (correccion1 + correccion2 + correccion3) /3

maxs_1_blanco2 = maxs_1_blanco * correccion1;
maxs_2_blanco2 = maxs_2_blanco * correccion2;
maxs_3_blanco2 = maxs_3_blanco * correccion3;

%%%%%%%%%%%%%%

figure(1);
%plot(separaciones./Dd,maxs_1_blanco,'x',separaciones./Dd,maxs_1_blanco
o2,'x',sep_norm_sim,maxs_primer_blanco_sim,'MarkerSize',15,'LineWidth'
,2);
plot(separaciones./Dd,maxs_1_blanco,'x',sep_norm_sim,maxs_primer_blanco
o_sim,'MarkerSize',15,'LineWidth',2);
xlabel('Separaciones (unidades FWHM)','fontsize',20);
ylabel('Amplitud normalizada','fontsize',20);
%title(['Amplitudes normalizadas del primer blanco. Corrección: '
sprintf('%0.3g',correccion1)],'fontsize',26);
title('Amplitudes normalizadas del primer blanco.','fontsize',26);
grid;
figure(2);

```

```

%plot(separaciones./Dd,maxs_2_blanco,'x',separaciones./Dd,maxs_2_blanco2,'x',sep_norm_sim,maxs_segundo_blanco_sim,'MarkerSize',15,'LineWidth',2);
plot(separaciones./Dd,maxs_2_blanco,'x',sep_norm_sim,maxs_segundo_blanco_sim,'MarkerSize',15,'LineWidth',2);
xlabel('Separaciones (unidades FWHM)','fontsize',20);
ylabel('Amplitud normalizada','fontsize',20);
%title(['Amplitudes normalizadas del segundo blanco. Corrección: '
sprintf('%0.3g',correccion2)],'fontsize',26);
title('Amplitudes normalizadas del segundo blanco.','fontsize',26);
grid;
figure(3);
%plot(separaciones./Dd,maxs_3_blanco,'x',separaciones./Dd,maxs_3_blanco2,'x',sep_norm_sim,maxs_tercer_blanco_sim,'MarkerSize',15,'LineWidth',2);
plot(separaciones./Dd,maxs_3_blanco,'x',sep_norm_sim,maxs_tercer_blanco_sim,'MarkerSize',15,'LineWidth',2);
xlabel('Separaciones (unidades FWHM)','fontsize',20);
ylabel('Amplitud normalizada','fontsize',20);
%title(['Amplitudes normalizadas del tercer blanco. Corrección: '
sprintf('%0.3g',correccion3)],'fontsize',26);
title('Amplitudes normalizadas del tercer blanco.','fontsize',26);
grid;
figure(4);
%plot(separaciones./Dd,pos_maxs_cristal./Dd,'x',sep_norm_sim,err_dist_primer_blanco_sim,'MarkerSize',15);
plot(separaciones./Dd,pos_maxs_1_blanco./Dd,'x',sep_norm_sim,err_dist_primer_blanco_sim,'MarkerSize',15,'LineWidth',2);
xlabel('Separaciones (unidades FWHM)','fontsize',20);
ylabel('Error en separación (unidades FWHM)','fontsize',20);
title('Errores en distancia del primer blanco','fontsize',26);
grid;
figure(5);
%plot(separaciones./Dd,pos_maxs_espejo./Dd,'x',sep_norm_sim,err_dist_segundo_blanco_sim,'MarkerSize',15);
plot(separaciones./Dd,pos_maxs_2_blanco./Dd,'x',sep_norm_sim,err_dist_segundo_blanco_sim,'MarkerSize',15,'LineWidth',2);
xlabel('Separaciones (unidades FWHM)','fontsize',20);
ylabel('Error en separación (unidades FWHM)','fontsize',20);
title('Errores en distancia del segundo blanco','fontsize',26);
grid;
figure(6);
%plot(separaciones./Dd,pos_maxs_espejo./Dd,'x',sep_norm_sim,err_dist_segundo_blanco_sim,'MarkerSize',15);
plot(separaciones./Dd,pos_maxs_3_blanco./Dd,'x',sep_norm_sim,err_dist_tercer_blanco_sim,'MarkerSize',15,'LineWidth',2);
xlabel('Separaciones (unidades FWHM)','fontsize',20);
ylabel('Error en separación (unidades FWHM)','fontsize',20);
title('Errores en distancia del tercer blanco','fontsize',26);
grid;
for i = 1:num_puntos
    figure(i+6);
    P_1_blanco = grafica_1_blanco(:,i);
    P_2_blanco = grafica_2_blanco(:,i);
    P_3_blanco = grafica_3_blanco(:,i);
    P_total = grafica_total(:,i);
    P_suma = grafica_suma(:,i);

plot(d(1:2*mitad),P_1_blanco(1:2*mitad),d(1:2*mitad),P_2_blanco(1:2*mitad),d(1:2*mitad),P_3_blanco(1:2*mitad),d(1:2*mitad),P_total(1:2*mitad),d(1:2*mitad),P_suma(1:2*mitad),'LineWidth',2);

```

```
xlabel('Distancia (m)', 'fontsize', 20);
ylabel('Amplitud Normalizada', 'fontsize', 20);
    dist2 = distancias(i,:);

%filtramos espacios
dist = [];
for k = 1:size(dist2,2)
    if dist2(k) ~= ' '
        dist = [dist dist2(k)]; %#ok<AGROW>
    end
end
title([dist '(' sprintf('%0.3g', separaciones(i)) 'm). Errores: '
sprintf('%0.3g', err_amp_1(i)*100) '%, '
sprintf('%0.3g', err_amp_2(i)*100) '%, '
sprintf('%0.3g', err_amp_3(i)*100) '%'], 'fontsize', 26);
grid;
end
%figure(1);
%plot(d(1:2*mitad), P_1_blanco(1:2*mitad), d(1:2*mitad), P_2_blanco(1:2*mitad), d(1:2*mitad), P_3_blanco(1:2*mitad), d(1:2*mitad), P_total(1:2*mitad));
```

### Estimación de la atenuación en medidas de dos blancos:

```

%NOTA: La numeración de los blancos es según se los encuentra el
laser.
%El primer blanco es el primer cristal que se encuentra, el segundo el
%segundo cristal, y el tercero el espejo

%Constantes
c = 3e8;
Fsample = 250e3;
puntos_fft = 2^20;
directorio_datos = 'E:\PFC\Datos\Medidas de 2 blancos\Caracterización
de la atenuación\';
umbral = 0.7;
extension = 'txt';

%Opciones
Fm = 1000;
Fmin = 10e6;
Fmax = 40e6;
pos_min = 40;
pos_max = 60;
distancias = ['15_100%'; '15_80% '; '15_60% '; '15_40% '; '15_20% '];
%#ok<NBRAK>

%Inicializaciones
Tm = 1/Fm;
Df = Fmax - Fmin;
Dd = c/(2*Df);
dmax = c*Fsample*Tm/Df/2;

%Al lio
num_puntos = size(distancias,1);
separaciones = zeros(1,num_puntos);
at_blanco_1 = zeros(1,num_puntos);
pos_maxs_1_blanco = zeros(1,num_puntos);
at_blanco_2 = zeros(1,num_puntos);
pos_maxs_2_blanco = zeros(1,num_puntos);
err_amp_1 = zeros(1,num_puntos);
err_amp_2 = zeros(1,num_puntos);
grafica_1_blanco = [];
grafica_2_blanco = [];
grafica_total = [];
grafica_suma = [];

for i = 1:num_puntos
    dist2 = distancias(i,:);

    %filtramos espacios
    dist = [];
    for k = 1:size(dist2,2)
        if dist2(k) ~= ' '
            if dist2(k) == '_'
                dist = [dist ' ']; %#ok<AGROW>
            else
                dist = [dist dist2(k)]; %#ok<AGROW>
            end
        end
    end
end
end

```

```

    archivo_1_blanco = [directorio_datos dist ' '
sprintf('%d',Fmin/1e6) ' a ' sprintf('%d',Fmax/1e6) ' MHz, blanco 1
ON, blanco 2 OFF, blanco 3 OFF.' extension];
    archivo_2_blanco = [directorio_datos dist ' '
sprintf('%d',Fmin/1e6) ' a ' sprintf('%d',Fmax/1e6) ' MHz, blanco 1
OFF, blanco 2 ON, blanco 3 OFF.' extension];
    archivo_total = [directorio_datos dist ' ' sprintf('%d',Fmin/1e6)
' a ' sprintf('%d',Fmax/1e6) ' MHz, blanco 1 ON, blanco 2 ON, blanco 3
OFF.' extension];
%
    datos_1_blanco = load(archivo_1_blanco)';
    datos_2_blanco = load(archivo_2_blanco)';
    datos_total = load(archivo_total)';

    [N,num_T] = size(datos_total);

    Tm = N/Fsample;

    resultado_1_blanco = datos_1_blanco * ones(num_T,1);
    R_1_blanco = fft(resultado_1_blanco,puntos_fft);
    P_1_blanco = abs(R_1_blanco);

    [max_1_blanco pos_max_1_blanco] = max(P_1_blanco(1:puntos_fft/2));

    resultado_2_blanco = datos_2_blanco * ones(num_T,1);
    R_2_blanco = fft(resultado_2_blanco,puntos_fft);
    P_2_blanco = abs(R_2_blanco);

    [max_2_blanco pos_max_2_blanco] = max(P_2_blanco);

    normalizacion = max_2_blanco;

    R_1_blanco = R_1_blanco./normalizacion;
    P_1_blanco = abs(R_1_blanco);

    R_2_blanco = R_2_blanco./normalizacion;
    P_2_blanco = abs(R_2_blanco);

    err_amp_1(i) =-(normalizacion - max_1_blanco)/normalizacion;
    err_amp_2(i) =-(normalizacion - max_2_blanco)/normalizacion;

    separacion = pos_max_2_blanco - pos_max_1_blanco;
    separaciones(i) = separacion*dmax/puntos_fft %#ok<NOPTS>
    if i == 1
        mitad = pos_max_2_blanco;
    end
    mitad_en_metros = mitad*dmax/puntos_fft;
    grafica_1_blanco = [grafica_1_blanco';P_1_blanco(1:2*mitad)']';
%#ok<AGROW>
    grafica_2_blanco = [grafica_2_blanco';P_2_blanco(1:2*mitad)']';
%#ok<AGROW>

    R_suma = R_1_blanco + R_2_blanco;
    P_suma = abs(R_suma);

```

```

grafica_suma = [grafica_suma';P_suma(1:2*mitad)']; %#ok<AGROW>

resultado_total = datos_total * ones(num_T,1);
R_total = fft(resultado_total,puntos_fft)./normalizacion;
P_total = abs(R_total);
grafica_total = [grafica_total';P_total(1:2*mitad)']; %#ok<AGROW>

amp_norm_1_blanco = max_1_blanco / normalizacion;
amp_norm_2_blanco = max_2_blanco / normalizacion;

%%50 a 80
%if i == 3
%    umbral = 0.2;
%end
%if i == 4
%    umbral = 0.5;
%end
%%10 a 40
%if i == 5
%    umbral = 0.5;
%end
%%-15 a 15
%if i == 12
%    umbral = 0.6;
%end

%Busqueda de maximos SUMA
dPrr = [P_suma(1:2*mitad)' 0] - [0 P_suma(1:2*mitad)'];
signo = sign(dPrr);
maximos_d_suma = [];
estado = signo(1);
for k = 2 : 2*mitad
    if signo(k) ~= estado
        %Estamos en un cero en k-1, veamos si es un maximo
        if P_suma(k-1) > P_suma(k)
            %Estamos en un maximo, veamos si supera el umbral
            if P_suma(k-1) > umbral
                maximos_d_suma = [maximos_d_suma k-1]; %#ok<AGROW>
            end
        end
        estado = -estado;
    end
end
%maximos_d = [maximos_d zeros(1,3-size(maximos_d,2))]; %#ok<AGROW>
num_maximos = size(maximos_d_suma,2);
if num_maximos > 2
    error('Numero de maximos mayor a 3, aumenta el umbral!!');
end

if num_maximos == 1
    maximos_d_suma = [maximos_d_suma(1) maximos_d_suma(1)];
%#ok<UNRCH,AGROW>
end

%Busqueda de maximos
dPrr = [P_total(1:2*mitad)' 0] - [0 P_total(1:2*mitad)'];
signo = sign(dPrr);
maximos_d = [];
estado = signo(1);
for k = 2 : 2*mitad

```



```

    if signo(k) ~= estado
        %Estamos en un cero en k-1, veamos si es un maximo
        if P_total(k-1) > P_total(k)
            %Estamos en un maximo, veamos si supera el umbral
            if P_total(k-1) > umbral
                maximos_d = [maximos_d k-1]; %#ok<AGROW>
            end
        end
        estado = -estado;
    end
end
num_maximos = size(maximos_d,2);
if num_maximos > 2
    error('Numero de maximos mayor a 3, aumenta el umbral!!');
end

if num_maximos == 1
    %error('Numero de maximos igual a 2. ');
    maximos_d = [maximos_d(1) maximos_d(1)]; %#ok<UNRCH,AGROW>
    num_maximos = 2;
end

if num_maximos == 2
    at_blanco_1(i) = (P_suma(maximos_d_suma(1)) -
P_total(maximos_d(1)))/P_suma(maximos_d_suma(1));
    pos_maxs_1_blanco(i) = (maximos_d(1) -
maximos_d_suma(1))*dmax/puntos_fft;
    at_blanco_2(i) = (P_suma(maximos_d_suma(2)) -
P_total(maximos_d(2)))/P_suma(maximos_d_suma(2));
    pos_maxs_2_blanco(i) = (maximos_d(2) -
maximos_d_suma(2))*dmax/puntos_fft;
end

end

d = linspace(0,dmax,puntos_fft);

figure(1);
plot(separaciones./Dd,at_blanco_1,'x','MarkerSize',15,'LineWidth',2);
xlabel('Separaciones (unidades FWHM)','fontsize',20);
ylabel('Amplitud normalizada','fontsize',20);
title('Atenuación primer blanco','fontsize',26);
grid;
figure(2);
plot(separaciones./Dd,at_blanco_2,'x','MarkerSize',15,'LineWidth',2);
xlabel('Separaciones (unidades FWHM)','fontsize',20);
ylabel('Amplitud normalizada','fontsize',20);
title('Atenuacion segundo blanco','fontsize',26);
grid;
figure(3);
plot(separaciones./Dd,pos_maxs_1_blanco./Dd,'x','MarkerSize',15,'LineW
idth',2);
xlabel('Separaciones (unidades FWHM)','fontsize',20);
ylabel('Error en separación (unidades FWHM)','fontsize',20);
title('Errores en distancia del primer blanco','fontsize',26);
grid;
figure(4);
plot(separaciones./Dd,pos_maxs_2_blanco./Dd,'x','MarkerSize',15,'LineW
idth',2);
xlabel('Separaciones (unidades FWHM)','fontsize',20);

```

```

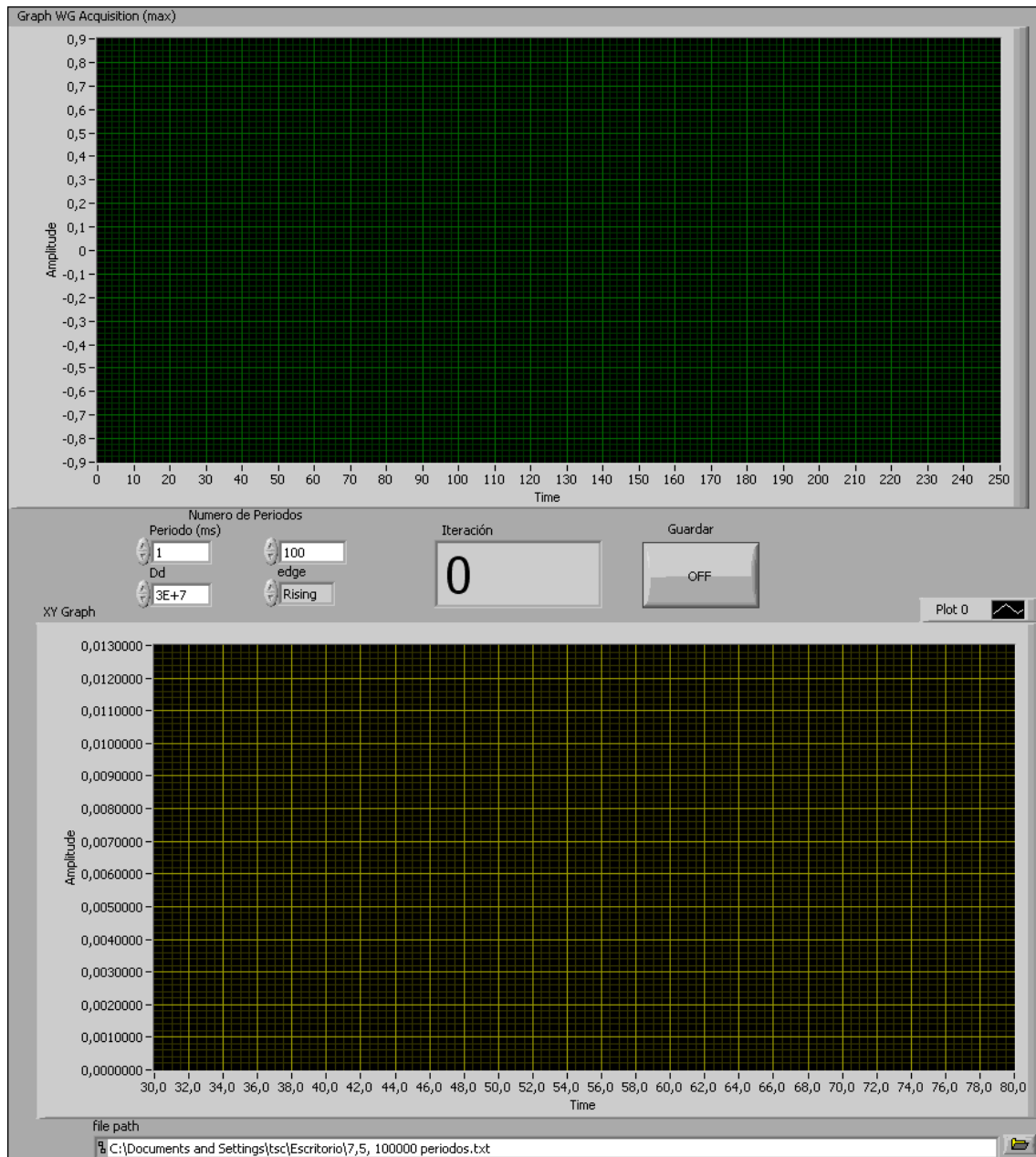
ylabel('Error en separación (unidades FWHM)', 'fontsize', 20);
title('Errores en distancia del segundo blanco', 'fontsize', 26);
grid;
for i = 1:num_puntos
    figure(i+4);
    P_1_blanco = grafica_1_blanco(:,i);
    P_2_blanco = grafica_2_blanco(:,i);
    P_total = grafica_total(:,i);
    P_suma = grafica_suma(:,i);

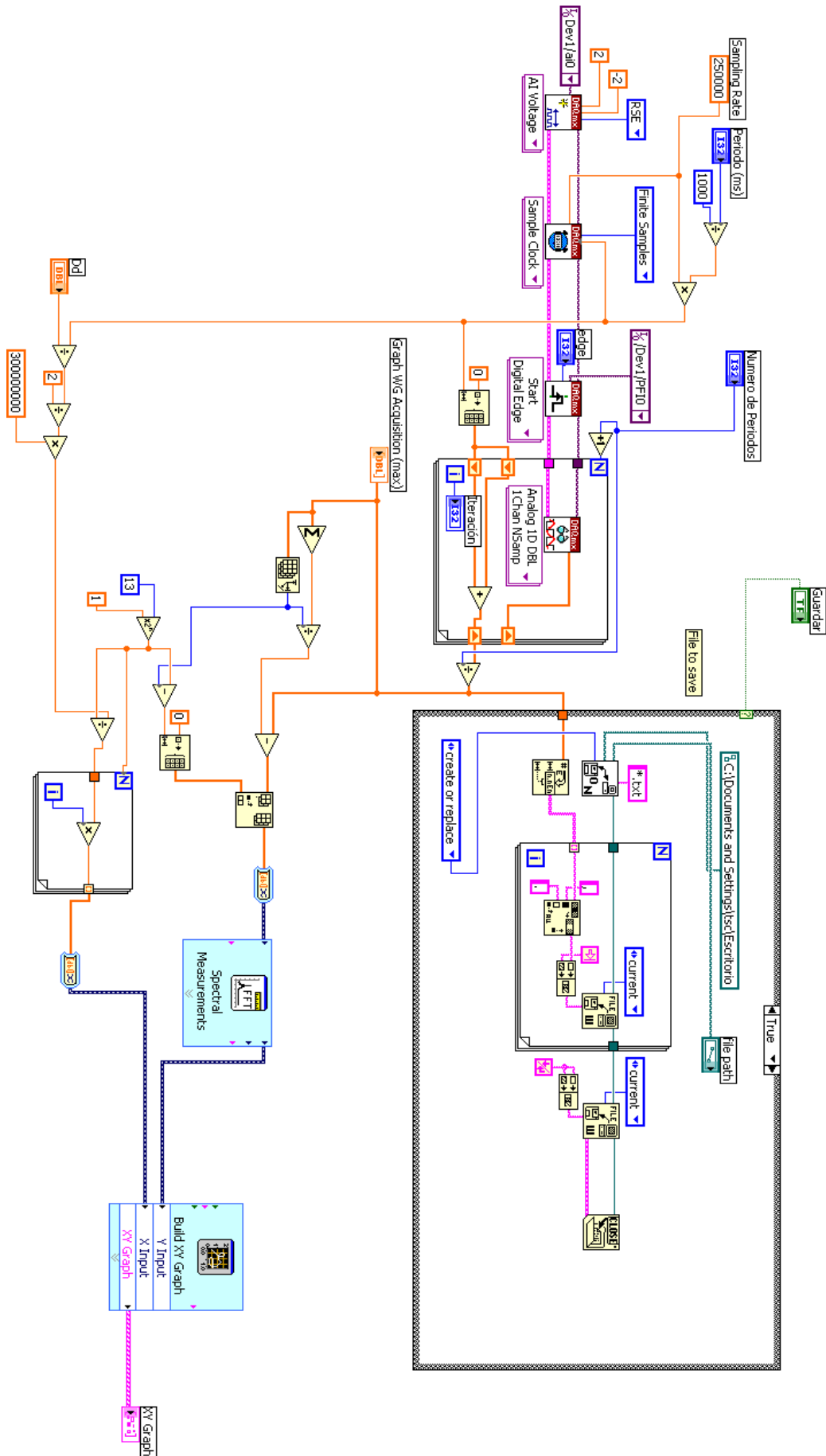
    plot(d(1:2*mitad), P_1_blanco(1:2*mitad), d(1:2*mitad), P_2_blanco(1:2*mi
tad), d(1:2*mitad), P_total(1:2*mitad), d(1:2*mitad), P_suma(1:2*mitad), '-
k', 'LineWidth', 2);
    xlabel('Distancia (m)', 'fontsize', 20);
    ylabel('Amplitud Normalizada', 'fontsize', 20);
    dist2 = distancias(i,:);

    %filtramos espacios
    dist = [];
    for k = 1:size(dist2,2)
        if dist2(k) ~= ' '
            if dist2(k) == '_'
                dist = [dist ' ']; %#ok<AGROW>
            else
                dist = [dist dist2(k)]; %#ok<AGROW>
            end
        end
    end
    title([dist '(' sprintf('%0.3g', separaciones(i)) 'm). Errores: '
sprintf('%0.3g', err_amp_1(i)*100) '%, '
sprintf('%0.3g', err_amp_2(i)*100) '%'], 'fontsize', 26);
grid;
end

```

Programa en LabView de captura de datos del laboratorio:





## **ANEXO III: APARATOS DEL LABORATORIO**

### Montaje



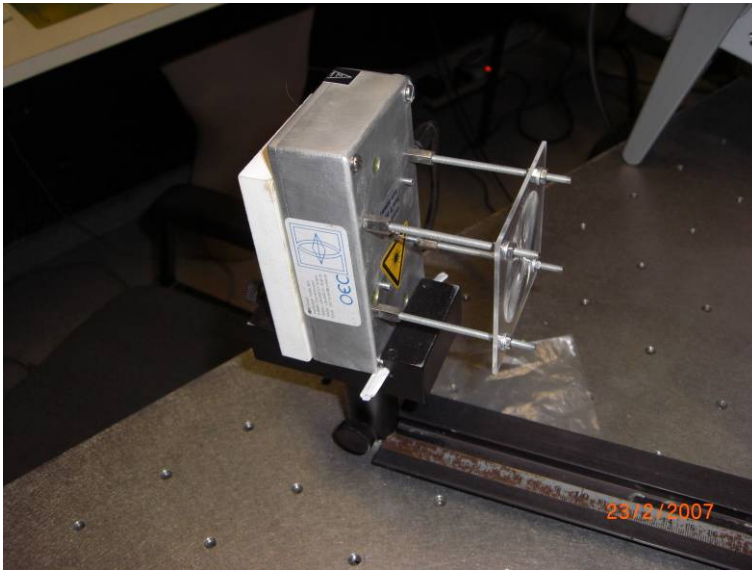




Generador de funciones Agilent 33250A

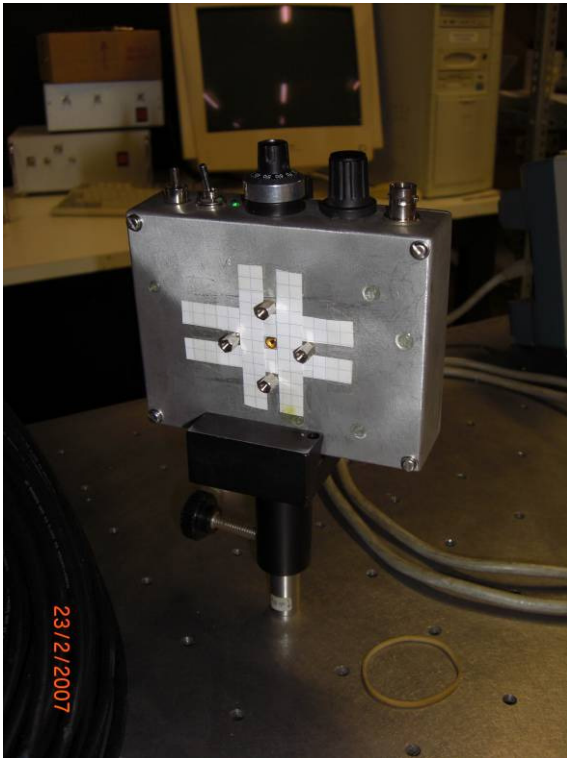


Módulo láser más lente de fresnell



Receptor Óptico





Lente biconvexa



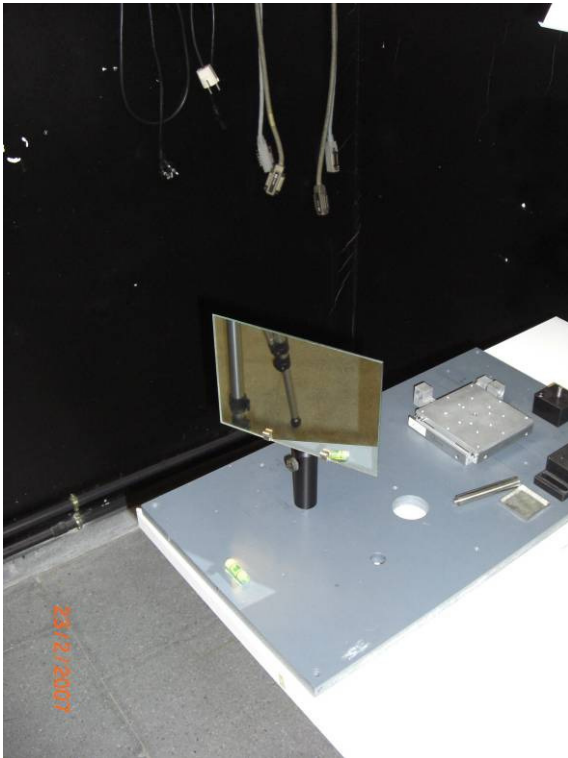
Frontal de conexiones



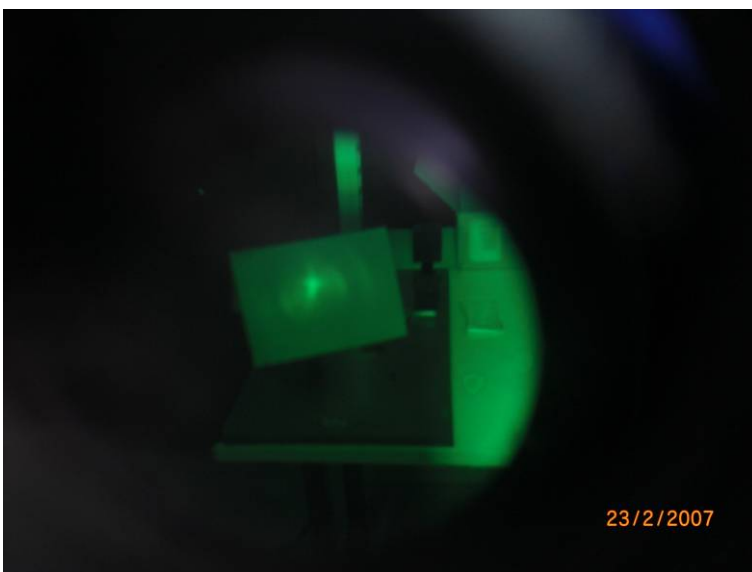
Beam splitter

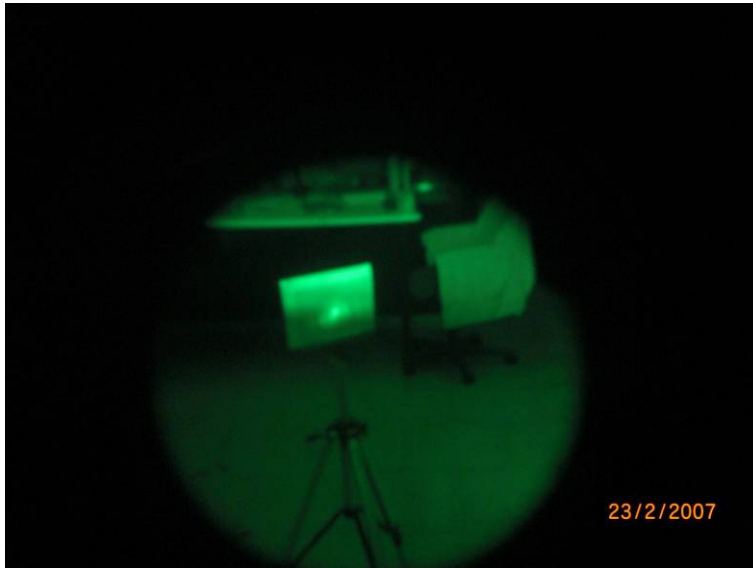


Blancos



Visor de infrarrojos







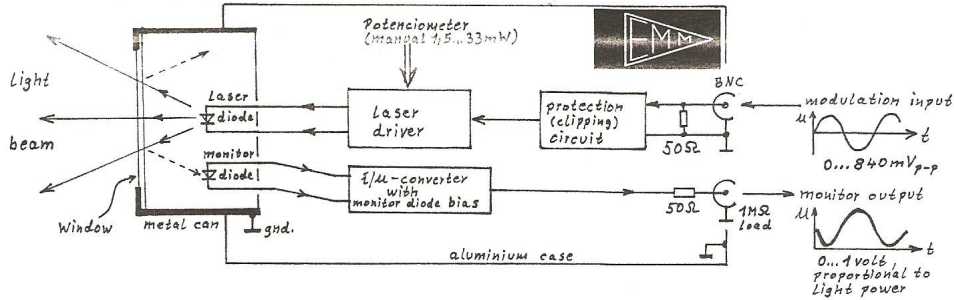
# ANEXO IV: CARACTERÍSTICAS DE LOS COMPONENTES

## Láser

Laser modul 904 nm / DC...>100 MHz / 0...50 mW

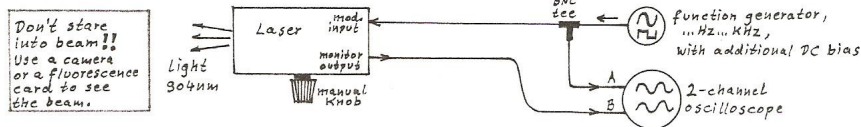
1/2007

This device, built into an aluminium case 11x6x3 cm, contains a single-mode laser diode, driving circuit with diode-protection and a monitor-diode amplifier. For collecting the divergent lightbeam, there are two lenses and mounting material.



how to use:

- To get a feeling for the modulation characteristic, try the following scheme:

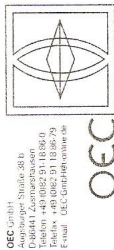


For modulation frequencies DC...about 70 kHz, the monitor signal is a good representation of light intensity. You can see the additional effect of manual knob adjust, output DC-bias of generator and AC-modulation. The laser diode is always protected against input overdriving due to the clipping function. (Although, the modulation signal don't should be greater than +/-4 volts in order to save the input resistors and diodes.)

The monitor diode works with the part of light reflected by the window. Dust on the surface or other reflections (mirrors or parallel glass plates in front of the window) will increase the monitor signal. External light from lamps entering the monitor diode, will make an additional part in the signal.

Note the phase reverse (180 degree) between modulation signal on the one hand and light modulation and monitor signal on the other hand.

- The frequency response of the monitor output decreases for frequencies greater than about 100 kHz. Regardless of the fast (>100 MHz) i/u-converter, the monitor channel is slow due to internal diffusion or travelling effects within the photodiode. The main utility of this output is adjusting the best DC- and AC-levels for the modulation, to adjust at low frequencies.
- For make a collimated beam, you can use the large fresnel lens or the small glass lens. Both lenses have an aspheric characteristic. See enclosed mounting kits. About 66% of laser output intensity will pass the lens. For eye safety, using of the large fresnel lens is the better way because the light power is divided to a broader beam. Use a white paper and a CCD-camera for visualizing. If needed, the intensity can be decreased easily by smaller DC- and AC-modulation. Control the purity of the modulated light (no clipping near zero) at low modulating frequency (several kHz) with the scope, see above.
- After the work, it's sufficient to switch off the device. The standby-losses of the small power adapter are negligible.

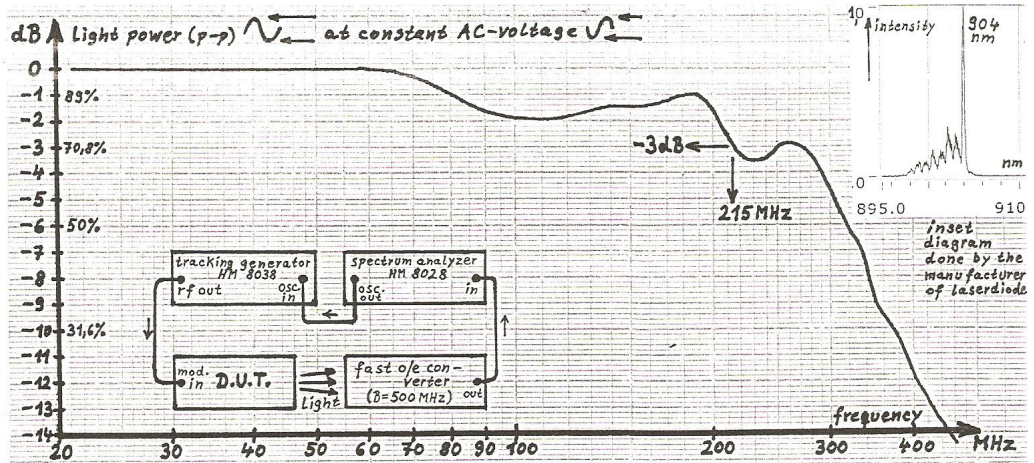


OEC GmbH  
 Optische Elektronik  
 Digitaler Zonenstrahltrieb  
 Telefon +49 030 852 01 14300-0  
 Telefax +49 030 852 01 14300-1  
 E-Mail: OEC-Cont@oec.de



specifications:

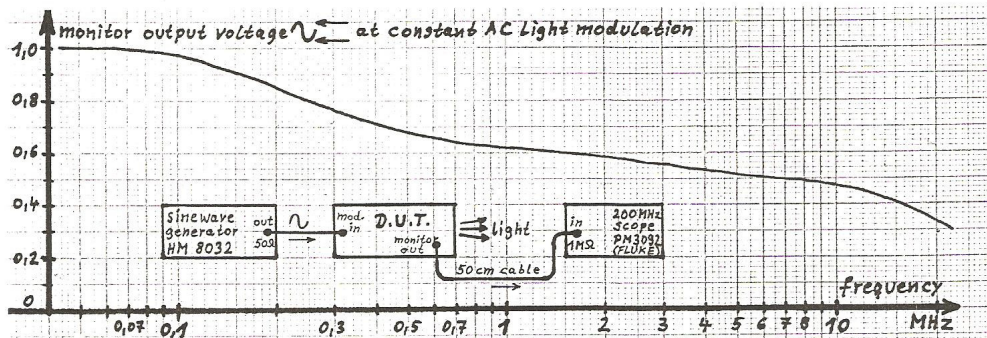
- modulation input:
  - 50 ohms input impedance.
  - input voltage 0...840 mvolts for light power variation 0...50 mwatts (p-p).
  - max. input voltage +/-4 volts !



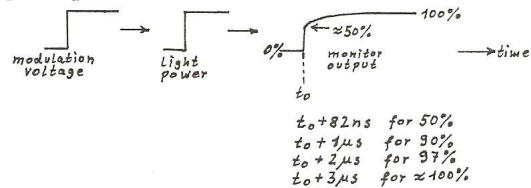
- clipped output at  $\geq 1$  volt(p-p) modulation:

- monitor output:

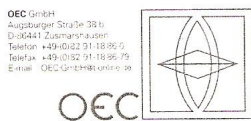
- 50 ohms output impedance.
- 0...+1 volt at 0...50 mwatt light power (with 1 MΩ load at monitor output).



• step response:



Decrease in frequency response and long tail in step response are caused by the characteristics of the monitor photodiode.



lead-free soldered  
 with Sn95,5Ag3,8Cu0,7



## Receptor óptico

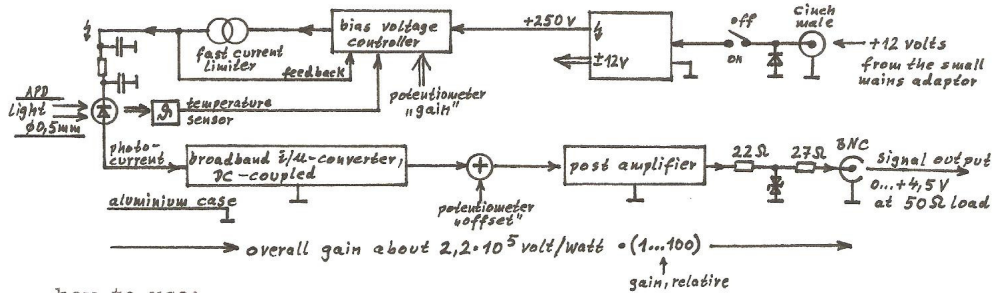
Silicon - APD -photoreceiver

3/2008

This device, built into an aluminium case 120x95x34 mm and supplied by an AC/DC-adaptor +12V/270mA, is useful for measuring light:

- wavelength 400...1000 nm ( $\lambda_{peak}$  at 820 nm)
- light power: few photons per impuls...20  $\mu$ W (clipping)
- modulation frequency: DC...185 MHz (-3 dB)

The light power entering the photodiode will be converted to a proportional output voltage with a good linearity at low and medium gain settings.



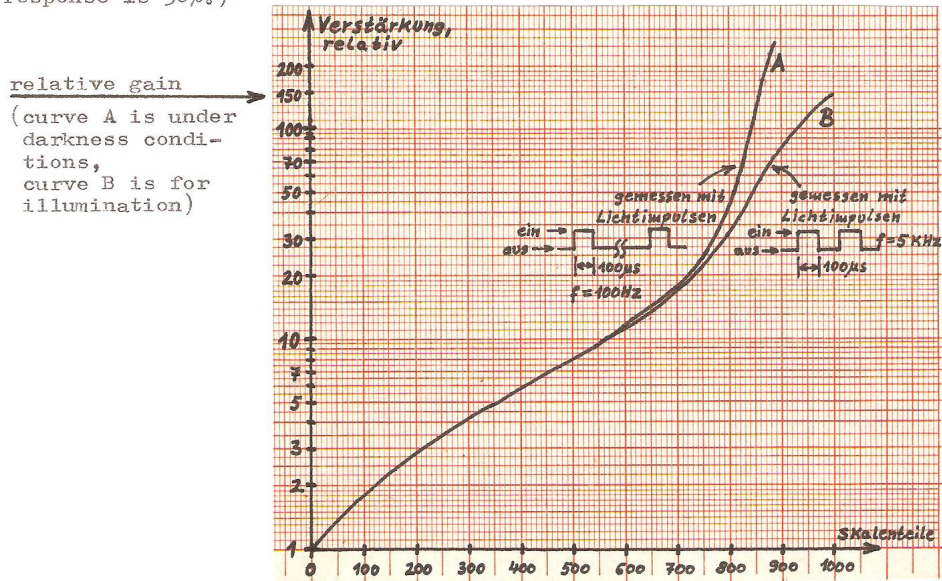
how to use:

- Mount the detector into the optical assembly. The four screwholes on the corners (110x85 mm) can be used for mounting the whole detector. Use metric M3 screws only !  
The four hexagonal bolts around the photodiode are for mounting a lens or another optics.
- This APD-detector can detect very weak light intensities. Due to the typical excess-noise of the avalanche effect, the device is very sensitive to disturbing light. Every light, even the light to measure, makes additional noise in the signal and decreases the SNR. This effect is low at low relative gains (potentiometer "gain" near to left) and increases at higher gains. In darkness and at very low light levels, gain and sensitivity can be turned to high values. Light impulses consisting of few photons can be seen. But under daylight conditions, a better SNR can be reached by using a smaller relative gain. Turn the potentiometer counterclockwise.  
If the device is used for free-space optical link, the optics in front of the photodiode must be designed with care. The receiving angle should be as small as possible. Every disturbing daylight must be avoided.
- When the detector is mounted into the experimental setup, then
  - turn the potentiometer "gain" counterclockwise
  - Now, switch the toggle to "on". The green LED is lighting. The device is ready to work now. With a scope, the output BNC connector can be controlled. In darkness, the output DC voltage can be nulled by the potentiometer "offset". The span is about -1..0...+1 V. During the first few minutes, dark offset drifts a little bit until the electronics is warmed up.
- Only at very very weak light intensities, it can be useful to turn the potentiometer "gain" clockwise to the region at 900 points. See the above diagram on the reverse. Turn with care ! On the scope, large output impulses (several volts high) can be seen. This is the beginning "geiger mode" which is used purely in photon counting systems. There is a dark count rate (about 10000 counts/sec., depending on "gain") and there is a count probability (photon  $\rightarrow$  impuls) in the order of few percents. Of course, this device without internal cooler is not optimized for photon counting purposes, but it may be enough for some experiments.
- Cable length / DC stress / using of lock-in-techniques: see dokumentation paper for the silicon-pin-photoreceiver (last 14 lines).



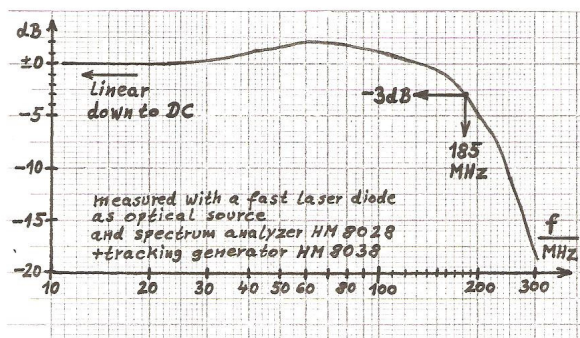
specifications (measured with 50 ohms load at output):

- gain at  $\lambda_{peak}$  (820nm):  $2,2 \cdot 10^5$  volts/watt (relative gain = 1)  
 (At 590 nm & 940 nm, response is 50%,  
 When potentiometer "gain" will be turned clockwise, gain increases up to 200-fold)



- output parameters:
  - source impedance 50 ohms.
  - signal voltage 0...+4,5 volts before saturation.
  - dark offset: depends on temperature, can be nullled by potentiometer "offset" (2 volts span).
  - broadband noise: 15 mvolts,pp  $\hat{=}$  3 mvolt RMS (with an oscilloscope 200 MHz),  
 3 mvolts,pp  $\hat{=}$  0,6 mvolt RMS (with an oscilloscope 20 MHz).
- input referred noise density:
  - NEP =  $0,9 \text{ pW}/\sqrt{\text{Hz}}$  (potentiometer at left - relative gain = 1)
  - =  $9 \text{ fW}/\sqrt{\text{Hz}}$  (potentiometer at 840 points - relative gain = 100)
  - only possible at very weak illumination !

- gain over frequency:  
 (measured with sinusoidal signals)



- power supply:
  - +12 volts, stabilized (from the AC/DC-adaptor), current consumption about 80 mA in darkness, rising at strong illumination.
- leadfree soldered.