

Trabajo Fin de Grado
Grado en Ingeniería de las Tecnologías de
Telecomunicación

Predistorsión digital en sistemas MIMO 2x2

Autor: María Vázquez de Górgolas

Tutor: María José Madero Ayora

Dpto. Teoría de la Señal y Comunicaciones
Escuela Técnica Superior de Ingeniería
Universidad de Sevilla

Sevilla, 2022



Trabajo Fin de Grado
Grado en Ingeniería de Tecnologías de Telecomunicación

Predistorsión digital en sistemas MIMO 2x2

Autor:

María Vázquez de Górgolas

Tutor:

María José Madero Ayora

Dpto. de Teoría de la Señal y Comunicaciones

Escuela Técnica Superior de Ingeniería

Universidad de Sevilla

Sevilla, 2022

Trabajo Fin de Grado: Predistorsión digital en sistemas MIMO 2x2

Autor: María Vázquez de Górgolas

Tutor: María José Madero Ayora

El tribunal nombrado para juzgar el Proyecto arriba indicado, compuesto por los siguientes miembros:

Presidente:

Vocales:

Secretario:

Acuerdan otorgarle la calificación de:

Sevilla, 2022

El Secretario del Tribunal

A mi familia

A mis maestros

Agradecimientos

En primer lugar, agradecer a María José por su tiempo, su dedicación y su disponibilidad como tutora de este trabajo, y por las ganas de enseñar que transmite como profesora. Gracias igualmente a todos los profesores que me han transmitido todos sus conocimientos y, con ello, han hecho posible que hoy me encuentre terminando mis estudios.

Gracias a mis abuelos, en especial a Lala, que siempre me dijo que llegaría a dónde me propusiese y me quiso incondicionalmente toda su vida. Sé que estáis muy orgullosos de mí.

Agradecer también a mis padres por el cariño, la paciencia y todos los consejos; y a mi hermana Sonia, por ser ejemplo de valentía, constancia y dedicación.

Gracias a mis amigos de la carrera, *los P.U.*, por los cafés, los bocatas de Carlos y Jose, las horas de clase, las de estudio y las de prácticas, las fiestas y los días interminables en la ETSI. Y también gracias a mis amigos de fuera, por verme crecer y acompañarme siempre.

Por último, gracias a Dani, por darme toda la paz que mi mente necesita cuando está a punto de colapsar.

María Vázquez

Sevilla, 2022

Resumen

Este trabajo intenta linealizar un sistema de tipo MIMO 2x2. Para ello, será necesario hacer un modelado del sistema, para luego aplicarle predistorsión digital. La predistorsión digital es un bloque de aprendizaje que consigue hacer el efecto inverso a un amplificador de potencia, cuyo comportamiento es no lineal. Así pues, al sumar los efectos del amplificador y del bloque de predistorsión se obtiene un sistema lineal. Los resultados experimentales se han logrado para dos amplificadores tipo AB, cuya tarea ha sido amplificar una señal MIMO 2x2 a 2 GHz.

Abstract

This work attempts to linearize a MIMO 2x2 type system. To do this, it will be necessary to model the system, and then apply digital predistortion. Digital predistortion is a learning block that manages to do the opposite effect of a power amplifier, whose behavior is non-linear. Thus, by adding the effects of the amplifier and the predistortion block, a linear system is obtained. The experimental results have been achieved for two class AB amplifiers, whose task has been to amplify a MIMO 2x2 signal at 2 GHz.

Índice

Agradecimientos	ix
Resumen	xi
Abstract	xiii
Índice	xv
Índice de Tablas	xvii
Índice de Figuras	xix
1 Introducción	1
2 5G-New Radio.....	3
2.1 OFDM.....	3
2.2 Multiple Input Multiple Output (MIMO).....	6
2.3 Agregación de portadoras (CA) y múltiples bandas concurrentes.....	7
3 Amplificador de potencia como sistema no lineal	9
3.1 Sistema lineal e invariante en el tiempo (LTI)	9
3.2 Sistemas no lineales.....	10
3.3 Series de Volterra.....	12
3.4 Expresión en forma matricial.....	14
3.5 Predistorsión digital.....	15
4 Modelos de múltiples entradas	17
4.1 Modelos bidimensionales para el caso de doble banda concurrente.....	17
4.2 Montaje experimental.....	18
4.3 GMP para transmisores MIMO propuesto por S. Amin.....	19
4.3.1 Generalized Memory Polinomial for Linear Crosstalk (GMPLC)	19
4.3.2 Generalized Memory Polinomial for NonLinear Crosstalk (GMPNLC).....	19
4.3.3 Extended Generalized Memory Polinomial for NonLinear Crosstalk (EGMPNLC)	20
4.3.4 Parallel Hammerstein for a MIMO Transmitter 2x2(PH2x2).....	20
4.3.5 Mejoras en la implementación del modelo.....	21
4.4 Procedimiento para identificar el modelo: la regresión.....	21
5. Resultados con amplificador sintético	23
5.1. Modelado.....	23
5.1.1. Prueba por el método de mínimos cuadrados	23
5.1.2. Prueba por el método DOMP.....	25
5.2. Predistorsión.....	27
6. Resultados de laboratorio.....	31
6.1. Modelado.....	31

6.2. <i>Predistorsión digital</i>	32
7. Conclusiones y líneas futuras	43
Anexo A: Datashet	45
Anexo B: Código de Matlab empleado en el trabajo	51
Anexo C: Resultados adicionales	65
Bibliografía	69

ÍNDICE DE TABLAS

Tabla 2.1: Ancho de banda y número de PRBs según separación Δf y canalización.	5
Tabla 5.1: Orden no lineal y profundidad de memoria de los diferentes modelos en la prueba con LS.	24
Tabla 5.2: NMSE para los distintos modelos y acoplamiento de 30 dB con LS.	24
Tabla 5.3: ACEPR para los distintos modelos y acoplamiento de 30 dB con LS.	24
Tabla 5.4: NMSE para los distintos modelos y acoplamiento de 10 dB con LS.	25
Tabla 5.5: ACEPR para los distintos modelos y acoplamiento de 10 dB con LS.	25
Tabla 5.6: Orden no lineal y profundidad de memoria de los diferentes modelos en la prueba con DOMP.	26
Tabla 5.7: NMSE, en dB, para los distintos modelos y acoplamiento de 30 dB con DOMP.	26
Tabla 5.8: ACEPR para los distintos modelos y acoplamiento de 30 dB con DOMP.	26
Tabla 6.1: NMSE, en dB, para niveles de potencia en el generador de -35 y -37 dBm.	32
Tabla 6.2: Configuración del modelo para la prueba 4 en predistorsión.	33
Tabla 6.3: Tabla de regresores más importantes para los cuatro modelos.	41

ÍNDICE DE FIGURAS

Figura 2.1: Señal OFDM.	3
Figura 2.2: Definición de recurso tiempo-frecuencia (PRB).	4
Figura 2.3: Implementación de una modulación OFDM.	5
Figura 2.4: Esquema de un sistema MIMO $N_t \times N_r$.	6
Figura 2.5: Tipos de CA.	7
Figura 3.1: Sistema de múltiples entradas $u(t)$ y salidas $y(t)$.	9
Figura 3.3: Sistema lineal.	9
Figura 3.3: Potencia de salida frente a potencia de entrada para diferentes ganancias.	10
Figura 3.4: Símbolos transmitidos y recibidos, respectivamente.	12
Figura 3.5: Esquema de predistorsión digital.	15
Figura 3.6: Técnica de DLA.	16
Figura 3.7: Técnica de aprendizaje directo para sistema MIMO 2×2 .	16
Figura 4.1: Espectro de una señal de doble banda concurrente	17
Figura 4.2: Esquema de utilización de un transmisor MIMO	19
Figura 5.1: NMSE de x_1 para distintas velocidades de aprendizaje.	27
Figura 5.2: NMSE de x_2 para distintas velocidades de aprendizaje.	28
Figura 5.3: ACPR de x_1 para distintas velocidades de aprendizaje.	28
Figura 5.4: ACPR de x_2 para distintas velocidades de aprendizaje.	29
Figura 5.5: Ganancia en la primera y última iteración.	29
Figura 5.6: Característica AM/PM en la primera y última iteración.	30
Figura 6.1: Set-up de laboratorio.	31
Figura 6.2: NMSE de la señal x_1 para los distintos modelos GMPNLC y EGMPNLC.	34
Figura 6.3: NMSE de la señal x_2 para los modelos GMPNLC y EGMPNLC	34
Figura 6.4: Peor valor de ACPR de la señal x_1 para los modelos GMPNLC y EGMPNLC	35
Figura 6.5: Peor valor de ACPR de la señal x_2 para los modelos GMPNLC y EGMPNLC.	35
Figura 6.6: NMSE de la señal x_1 .	36
Figura 6.7: NMSE de la señal x_2 .	36
Figura 6.8: ACPR de la señal x_1 .	37
Figura 6.9: ACPR de la señal x_2 .	37
Figura 6.10: Constelación.	38
Figura 6.11: Espectro.	38

1 INTRODUCCIÓN

La comunicación es esencial para el ser humano, por lo que siempre se han intentado buscar técnicas de mejora o facilidades para que ésta se pueda desarrollar. Uno de los grandes avances para la comunicación es la posibilidad de comunicarse a distancia, que nace con la primera generación de comunicaciones móviles (1G). Sin embargo, es a partir de 2G cuando aparecen las comunicaciones digitales, y, con ello, mejoras en la comunicación como la mensajería instantánea SMS. Esto supuso sobre todo un aumento de la demanda de comunicaciones móviles, que suponen la creación de las generaciones 3G y 4G y de sus prestaciones nuevas, como la posibilidad de tener conectividad constante, poder usar todo tipo de aplicaciones y ver videos *in streaming*.

A día de hoy, las comunicaciones móviles han adquirido un papel tan fundamental en nuestra vida que se busca un escenario de conectividad total, donde todos nuestros dispositivos estén conectados entre ellos. Esto es lo que se conoce como *Internet Of Things* (IoT).

Si bien LTE, el estándar 4G, ha evolucionado favorablemente a lo largo del tiempo, no es capaz de gestionar el tráfico de datos tan heterogéneo y continuo que se espera. Por ello, nace la quinta generación de comunicaciones móviles (5G), que pretende proporcionar conectividad de banda ancha extrema, baja latencia y ultra robusta.

Una de las grandes investigaciones para intentar mejorar la eficiencia de las comunicaciones móviles se trata de estudiar el comportamiento del amplificador de potencia. Este dispositivo tiene un comportamiento no lineal, por lo que se dedican grandes esfuerzos a intentar linealizar el sistema. Además, si se trabaja con potencias de entrada en el amplificador cercanas a su punto de saturación, debido a las características de la señal modulada, el amplificador termina por saturar. Por último, otro inconveniente de los amplificadores de potencia se trata de su baja eficiencia energética, ya que gran parte de la potencia que entra en el amplificador se disipa en forma de calor.

Este trabajo se centra en una de las soluciones propuestas para los problemas que supone trabajar con un amplificador de potencia: la predistorsión digital. Este método consiste en diseñar un bloque no lineal que realice el comportamiento inverso a un amplificador de potencia, para que así la salida final del sistema sea lineal. La incorporación de este bloque requiere conocer el comportamiento del amplificador, y para ello existen técnicas como las series de Volterra.

Por tanto, el objetivo de este trabajo es modelar un bloque de predistorsión digital para linealizar un sistema MIMO 2x2, lo que supone un reto mayor pues se tendrán dos señales de entrada mezcladas mediante acopladores en cada amplificador. Para saber si el sistema se ha linealizado correctamente, se recurrirá a los conceptos de NMSE, ACEPR y ACPR, conocidas como figuras de mérito.

De manera que en el Capítulo 2 de este proyecto se explicarán los conceptos básicos necesarios para entender la capa física de 5G, 5G-NR; y en el Capítulo 3 se definirán las distintas figuras de mérito, se explicarán las series de Volterra y el concepto de predistorsión digital. Por otro lado, el Capítulo 4 estará dedicado a los modelos de múltiples entradas, donde cobra importancia el modelado de una señal MIMO 2x2; y los Capítulos 5 y 6 explican los resultados obtenidos mediante simulación en MATLAB y en laboratorio, respectivamente. Por último, el Capítulo 7 presenta las conclusiones obtenidas.

2 5G-NEW RADIO

Se entiende por 5G-New Radio (5G-NR) la capa física de la tecnología de quinta generación de comunicaciones móviles, 5G. Dicha tecnología nace por el crecimiento exponencial de la demanda de requisitos de aumento de ancho de banda y disminución de la latencia dado por el alto número de dispositivos conectados a Internet. Este crecimiento del ancho de banda ha hecho que sea necesario también la utilización de antenas de tecnología MIMO (del inglés *Multiple Input Multiple Output*), de la cual se hablará en el apartado 2.2.

5G-NR emplea un sistema de acceso múltiple que combina OFDMA (acceso múltiple por división en frecuencias ortogonales, del inglés *Orthogonal Frequency-Division Multiple Access*), TDMA (acceso múltiple por división en tiempo, del inglés *Time-Division Multiple Access*) y SDMA (acceso múltiple por división en espacio, del inglés *Space-Division Multiple Access*).

El despliegue de 5G se lleva a cabo mediante *releases* o actualizaciones del 3GPP. La primera especificación de 5G es la *Release 15*, que comenzó a finales de 2017 y quedó paralizada en junio de 2019. Es lo que se conoce como *5G Phase 1*. La *Release 16* se terminó en diciembre de 2019 y quedó congelada en junio de 2020, y se le conoce como *5G Phase 2*. Actualmente se trabaja en la *Release 17* y se planifica la *Release 18*.

2.1 OFDM

OFDM (multiplexación por división en frecuencia ortogonal, del inglés *Orthogonal Frequency Division Multiplexing*) es una técnica de transmisión de información que se basa en la división del canal de comunicación en varias bandas de frecuencia igualmente espaciadas, que constituyen subcanales de banda estrecha paralelos. Todas las subportadoras son ortogonales unas a otras en el dominio de la frecuencia, es decir, los ceros de una subportadora se producen en los máximos de sus adyacentes, tal como se muestra en la figura 2.1.

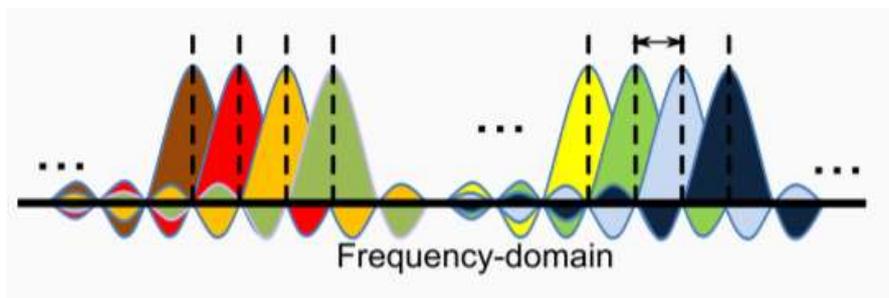


Figura 2.1: Señal OFDM.

Las señales que se emplean en 5G-NR son OFDM con prefijo cíclico (CP), esto es, se añade al principio de un símbolo OFDM una copia del final del mismo símbolo. El objetivo de esta copia es eliminar la interferencia entre portadoras (ICI), ya que si la longitud del CP es adecuada, la señal transmitida siempre es periódica y se garantiza la ortogonalidad de los símbolos OFDM después de propagarse por el canal. Hay dos tipos de prefijos cíclicos: CP normal o CP extendido. El primero de ellos es el que se utiliza por defecto, para escenarios en los que el margen de retraso es corto. Sin embargo, el CP extendido es más robusto, mejora la resistencia a la propagación del retardo, y, por tanto, se utiliza en escenarios en los que el margen de retraso es largo.

Un avance de 5G se halla en la duración de los símbolos OFDM. En LTE (4G), se emplea CP-OFDM, con una separación fija entre portadoras de 15 kHz. Para mantener la compatibilidad con LTE y a la vez tener

mayor flexibilidad, en 5G se introduce el concepto de numerología, que permite posibles valores de separación entre portadoras en función de un parámetro μ :

$$\Delta f = 2^\mu \cdot 15 \text{ kHz}, \quad \mu = 0,1,2,3,4,5.$$

Además, en esta modulación las tramas y subtramas tienen siempre la misma duración, y solo varía según se use CP extendido o normal. Sin embargo, la duración de los slots sí varía según la numerología, al igual que el número de slots que hay en cada trama.

Por otro lado, en LTE aparece el concepto de *physical resource block* (PRB), que se mantiene en 5G-NR. Este es el módulo básico de asignación de recursos, y está formado por 12 subportadoras consecutivas por cada símbolo OFDM en el dominio de la frecuencia. Además, cada PRB tiene la duración de un *slot* y está formado por 14 símbolos en CP normal o por 12 en CP extendido.

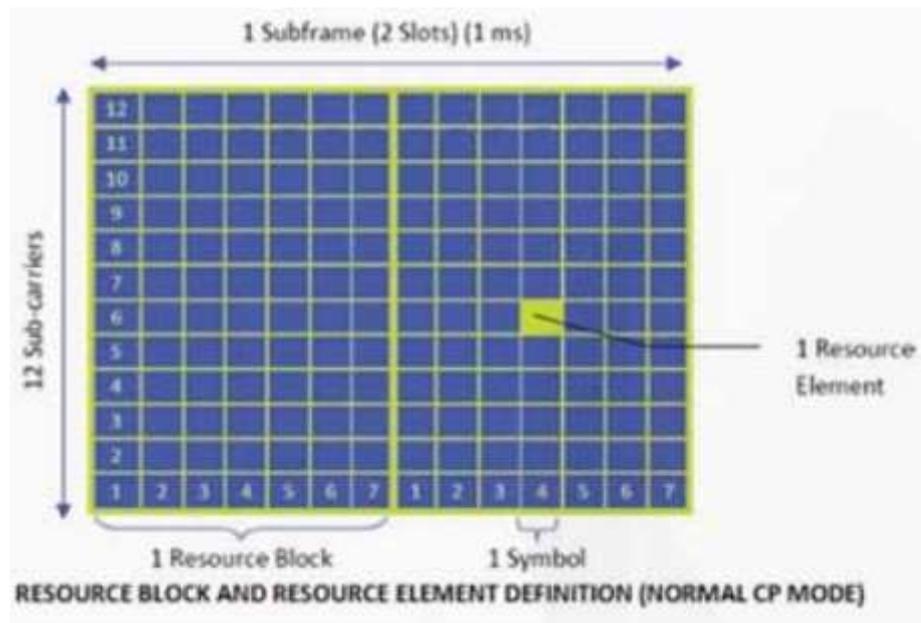


Figura 2.2: Definición de recurso tiempo-frecuencia (PRB).

El ancho de banda de transmisión ocupado dependerá del número de PRBs activos, de la canalización disponible y de la separación entre subportadoras. Las canalizaciones posibles en 5G son 5, 10, 15, 20, 25, 30, 40, 50, 60, 80 o 100 MHz para la franja FR1 (de 410 MHz a 7.125 GHz) y 50, 100, 200 o 400 MHz para la franja FR2 (de 24.25 a 52.60 GHz). En la Tabla 2.1. se muestra el número de PRBs para cada separación entre subportadoras que da lugar a esas canalizaciones y el ancho de banda ocupado por la señal transmitida.

FR1	SCS (kHz)	5 MHz	10 MHz	15 MHz	20 MHz	25 MHz	30 MHz	40 MHz	50 MHz	60 MHz	80 MHz	100 MHz
	15	25	52	79	106	133	[160]	216	270	N/A	N/A	N/A
	4.500 kHz	9.360 kHz	14.22 MHz	19.08 MHz	23.94 MHz	[28.8]	38.88 MHz	48.60 MHz	N/A	N/A	N/A	
30	11	24	38	51	65	[78]	106	133	162	217	273	
	3.960 kHz	8.640 kHz	13.68 MHz	18.36 MHz	23.40 MHz	[28.08]	38.16 MHz	47.88 MHz	58.32 MHz	78.12 MHz	98.28 MHz	
60	N/A	11	18	24	31	[38]	51	65	79	107	135	
	N/A	7.920 kHz	12.96 MHz	17.28 MHz	22.32 MHz	[27.36]	36.72 MHz	46.80 MHz	56.88 MHz	77.04 MHz	97.20	

FR2	SCS (kHz)	50 MHz	100 MHz	200 MHz	400 MHz
	60	66	132	264	N/A
	47.52 MHz	95.04 MHz	190.08 MHz	N/A	
120	32	66	132	264	
	46.08 MHz	95.04 MHz	190.08 MHz	380.16 MHz	

Tabla 2.1: Ancho de banda y número de PRBs según separación Δf y canalización.

Las ventajas de usar OFDM son claras, sin embargo, presenta un gran inconveniente, pues la diferencia de amplitud entre el máximo y el promedio de la señal es muy grande. Esto viene dado por la relación potencia pico a potencia promedio (PAPR), que se define como:

$$PAPR(dB) = 20 \log \left(\frac{\max[x[k]]}{\sqrt{\frac{1}{N} \sum_{n=1}^N |x[k]|^2}} \right)$$

Al ser muy grande este parámetro, habrá momentos en los que el amplificador tenga que trabajar con potencias muy altas, cercanas a la saturación. Esto hace que el amplificador trabaje en la zona de saturación también, lo que nos empeorará la calidad de la señal.

Con el fin de conseguir la ortogonalidad entre las señales, en el transmisor y receptor de un sistema con OFDM se utiliza la Transformada Inversa de Fourier, quedando el diagrama de bloques de un sistema OFDM como en la figura 2.2.

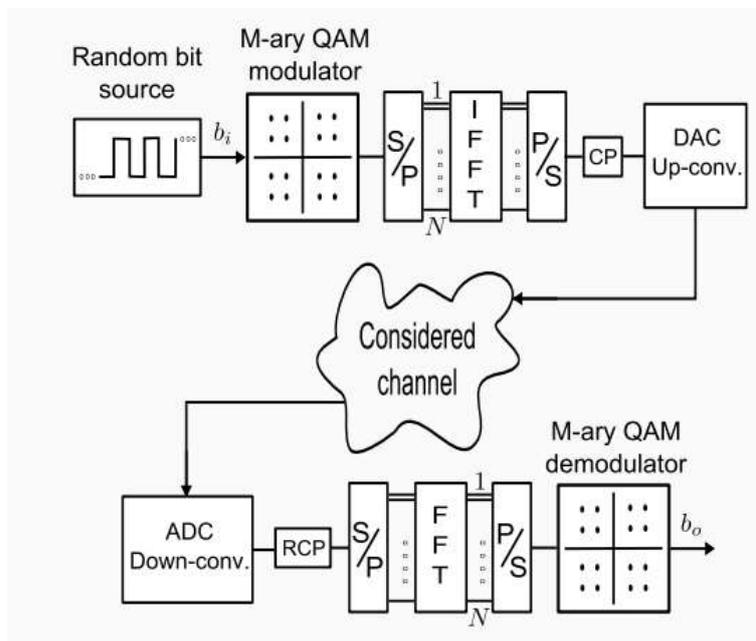


Figura 2.3: Implementación de una modulación OFDM.

2.2 Multiple Input Multiple Output (MIMO)

MIMO nace por la necesidad de aumentar la tasa binaria de un sistema, ya que éste es uno de los déficits de la transmisión de información en un sistema de comunicaciones.

La capacidad de un sistema de una única entrada y una única salida (SISO) viene determinada según el teorema de Shannon como:

$$C_{SISO} = B \cdot \log_2(1 + snr)$$

donde B es el ancho de banda de transmisión y snr la relación señal a ruido en unidades naturales.

Para aumentar dicha capacidad, lo que se planteaba antes de la existencia de MIMO era aumentar el ancho de banda, pero esto cada vez va siendo más complicado pues se consigue aumentando la frecuencia. Ampliar la frecuencia implica algunas desventajas como disminuir el alcance de la señal, lo cual no siempre es posible. Aun así, esta técnica también se utiliza en 5G, por ejemplo, para sensores de movimiento.

Otra manera de aumentar la capacidad es utilizar un sistema MIMO, con múltiples entradas y múltiples salidas, consiguiendo así que el teorema de Shannon quede como:

$$C_{MIMO} = N \cdot B \cdot \log_2(1 + snr)$$

donde N será el mínimo entre el número de entradas y de salidas. Así, por ejemplo, en el caso de un sistema MIMO 2x2, se consigue aumentar la capacidad al doble de un sistema SISO con la misma relación señal a ruido y el mismo ancho de banda.

Esta tecnología se consigue utilizando sistemas de múltiples antenas, permitiendo la creación de múltiples canales paralelos independientes en el espacio que transportan diferentes flujos de datos. Esto se conoce como multiplexado espacial. Es decir, en el mismo ancho de banda se transmiten varios flujos de forma simultánea, y estos flujos se consiguen separar casi al completo en el receptor. Para que esto último ocurra, es necesario que los distintos flujos presenten muy baja correlación, para que sean imposibles de confundir en el receptor.

Para esta transmisión, se utiliza un esquema como el de la figura 2.3.

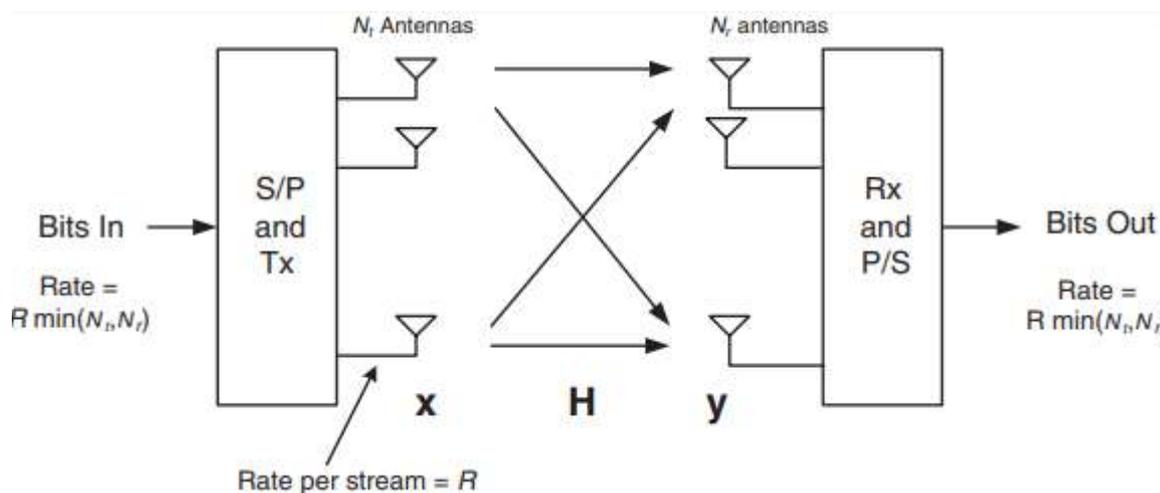


Figura 2.4: Esquema de un sistema MIMO $N_t \times N_r$.

El flujo de bits de entrada se divide en N_t flujos independientes, que se transmiten con una tasa binaria propia de un sistema SISO. Estos flujos llegan al receptor independientemente también, y mediante técnicas de demultiplexación se consigue a la salida el mismo flujo de bits que a la entrada.

El modelo matemático empleado para el multiplexado espacial se escribe matricialmente como $y=Hx+n$, donde y es el vector recibido, de tamaño $N_r \times 1$; n es el vector de ruido del sistema, de tamaño $N_r \times 1$; x es el vector transmitido, de tamaño $N_t \times 1$, y H es la matriz del canal, de tamaño $N_r \times N_t$ que tendrá la siguiente forma:

$$H = \begin{bmatrix} h_{11} & \dots & h_{1Nt} \\ \dots & \dots & \dots \\ h_{Nr1} & \dots & h_{NrNt} \end{bmatrix}$$

En nuestro caso, para el sistema MIMO 2x2, tendremos una matriz cuadrada 2x2 con los siguientes parámetros:

$$H = \begin{bmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{bmatrix}$$

Se puede demostrar que es posible decodificar los Nt flujos transmitidos siempre que:

$$\text{rango}(H) \geq Nt$$

En definitiva, los sistemas MIMO presentan múltiples ventajas y consiguen mejorar la capacidad de transmisión de un sistema sin necesidad de modificar el ancho de banda, aunque su coste sea elevado pues cada antena de transmisión y recepción tendrá su propia cadena de RF y sus propias técnicas de multiplexado.

2.3 Agregación de portadoras (CA) y múltiples bandas concurrentes

La agregación de portadoras es otra de las técnicas empleadas a partir de 4G+, que se incorporan también a 5G, que permite aumentar el ancho de banda de transmisión de un sistema.

Esta técnica permite sumar la tasa de transmisión que proporcionarían varias portadoras simultáneas con un cierto ancho de banda cada una. Cada una de estas portadoras puede usar una banda de frecuencia diferente.

CA es una técnica muy usada ya que en 5G-NR el máximo ancho de banda que puede tener una única portadora son 400 MHz, pero al poder usar múltiples portadoras, el ancho de banda de transmisión de la señal aumenta. Se pueden utilizar hasta 16 portadoras en el mismo ancho de banda, y se diferencian tres tipos de agregación de portadoras:

La primera de ellas surge cuando las portadoras están en bandas adyacentes, la segunda cuando las portadoras están en la misma banda pero no son contiguas y la tercera cuando las portadoras están en diferentes bandas. En la figura 2.4 podemos ver los tres tipos respectivamente.

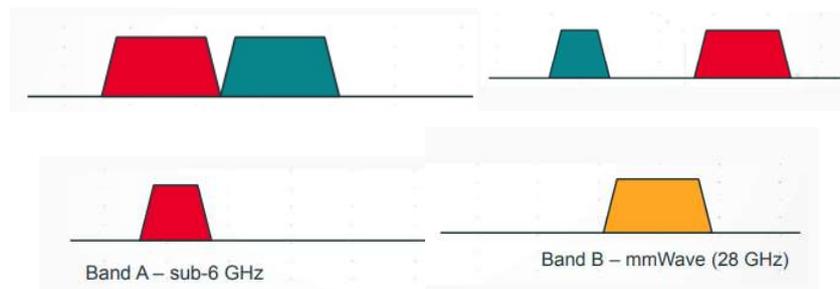


Figura 2.5: Tipos de CA.

3 AMPLIFICADOR DE POTENCIA COMO SISTEMA NO LINEAL

Un sistema se define como una “caja negra” que tiene un conjunto de entradas y salidas, estableciendo una relación entre ambas. Es decir, si tenemos una entrada a un sistema definida por $x(t)$, el sistema se encarga de transformar esta entrada en una salida $y(t)$. En nuestro caso, vamos a trabajar con sistemas de múltiples entradas y múltiples salidas, por tanto, la definición del sistema se basa en la transformación de múltiples entradas $x(t)$ en múltiples salidas $y(t)$.

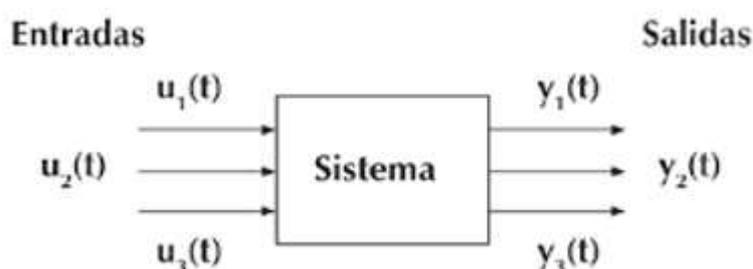


Figura 3.1: Sistema de múltiples entradas $u(t)$ y salidas $y(t)$.

3.1 Sistema lineal e invariante en el tiempo (LTI)

Se entiende que un sistema es lineal cuando cumple el principio de superposición. Este principio explica que si tenemos como salidas $y_1(t)$ e $y_2(t)$ correspondientes a las entradas $x_1(t)$ y $x_2(t)$, respectivamente, la resultante de la suma de entradas tiene que ser la suma de salidas, como se detalla en la Figura 3.2.

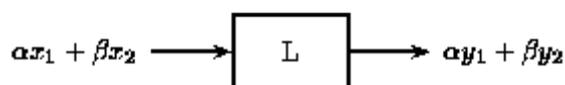


Figura 3.2: Sistema lineal.

Además, un sistema es invariante en el tiempo cuando, si se retrasa la entrada un tiempo t_0 , la salida sufre ese mismo retardo t_0 .

Los sistemas que cumplen estas dos propiedades son los sistemas lineales e invariantes en el tiempo (LTI), que son los más sencillos de manejar de una forma cuantitativa, pues el sistema viene definido por su respuesta impulsiva $h(t)$. Esta respuesta impulsiva se haya fácilmente, pues es la salida de un sistema que tiene como entrada la función $\delta(t)$. Así, la salida de cualquier sistema LTI $y(t)$ se encuentra realizando la convolución entre la entrada $x(t)$ y su respuesta al impulso $h(t)$.

El tipo de comportamiento que se espera en un amplificador de potencia es un comportamiento lineal, en concreto de escalado, es decir, que la potencia de entrada P_{in} (W) y de salida P_{out} (W) del amplificador estén relacionadas de la forma $P_{in} = g \cdot P_{out}$, donde g es una constante que se denominará ganancia del amplificador.

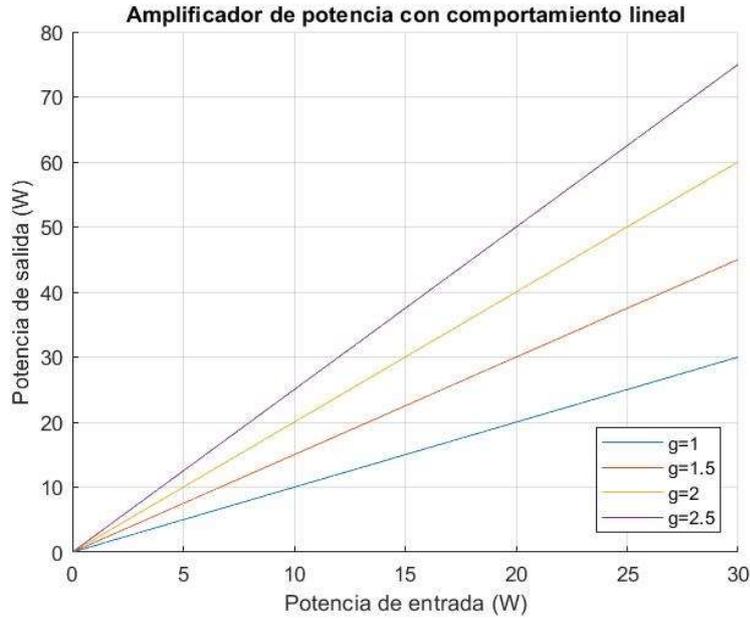


Figura 3.3: Potencia de salida frente a potencia de entrada para diferentes ganancias.

Aunque este tipo de sistemas tienen claras ventajas para trabajar con ellos por su fácil manejo, la realidad es que la mayoría de los amplificadores con los que trabajamos tienen un comportamiento no lineal, es por ello por lo que el siguiente apartado estará centrado en los sistemas no lineales.

3.2 Sistemas no lineales

Se define un sistema no lineal como aquel que no cumple el principio de superposición. El hecho de que no lo cumplan hace que aparezcan nuevas componentes en la señal, es decir, que ésta no sea una copia retardada y atenuada de la entrada. Esto no siempre es algo que queremos evitar, ya que muchas veces son necesarios estos comportamientos. Un sistema no lineal se define como:

$$x(t) \rightarrow y(t) = a_1x(t - \tau) + a_2x(t - \tau)^2 + \dots + a_nx(t - \tau)^n$$

donde n sería el orden del sistema no lineal. En el caso de que $x(t)$ fuese muy pequeña, las componentes polinómicas serían despreciables en comparación con el primer término. Sin embargo, esto no es compatible con la idea de que la potencia de salida sea grande, ya que para ello habría que hacer la potencia de entrada lo más grande posible, y, por tanto, $x(t)$ no podría ser pequeña.

Algunos ejemplos de elementos no lineales son los amplificadores o los mezcladores. Para medir cuán no lineal es un dispositivo, hay que hacer una caracterización experimental, tomando medidas de los efectos no lineales de un sistema. A continuación, se describen distintas pruebas.

3.2.1. Prueba de un tono

Para este caso, se describe la entrada del dispositivo bajo prueba (DUT) como:

$$x(t) = A_{in} \cos(2\pi f_c t) \quad (3.1)$$

En el caso que el DUT fuese un dispositivo lineal, según el apartado 3.1, su salida debería quedar como

$$y_L(t) = A_{out} \cos(2\pi f_c t + \phi_{out}(f)) \quad (3.2)$$

Y si el DUT fuese no lineal, según el apartado 3.2, su salida sería

$$y_{NL}(t) = \sum_{k=1}^{\infty} A_{out,k}(f, A_{in}) \cos[2\pi k f_c t + \phi_{out,k}(f, A_{in})] \quad (3.3)$$

En este último caso, aparecen armónicos en distintas frecuencias múltiplos de la frecuencia f_c de entrada.

Por tanto, esta prueba nos permite saber si un dispositivo es lineal o no. En el caso que lo fuese, la salida del sistema sería una copia de la entrada amplificada o atenuada y desplazada en fase. Si aparece más de un tono a la salida, el sistema sería no lineal.

3.2.2. Prueba de dos tonos

Aunque la prueba de un tono es capaz de determinar si un sistema es lineal o no, aporta poca información sobre las componentes no lineales que aparecen alrededor de la frecuencia fundamental. En este caso, la señal de entrada al DUT en el dominio del tiempo sería la suma de dos señales coseno con diferente frecuencia, que en el dominio de la frecuencia quedaría como dos tonos a las frecuencias f_1 y f_2 .

$$x(t) = A_1 \cos(2\pi f_1 t) + A_2 \cos(2\pi f_2 t) \quad (3.4)$$

Por tanto, en el caso que el sistema fuese no lineal, se distorsionan ambas sinusoides, quedando la salida como

$$y_{NL}(t) = \sum_{k=1}^{\infty} A_{out,k}(f, A_1, A_2) \cos [2\pi f_n t + \phi_{out,k}(f, A_1, A_2)], \text{ con } f_n = m f_1 + k f_2 \quad (3.5)$$

siendo m y k valores enteros. Los términos que se generan alrededor de las frecuencias fundamentales f_1 y f_2 se denominan productos de intermodulación de orden n , siendo $n = |m| + |k|$. En particular, los productos de intermodulación de orden tres provocan que la señal de salida tenga términos muy cercanos a las frecuencias de interés, lo que hará que sean más difíciles de eliminar. Por ello, se define como figura de mérito el punto de intercepto de tercer orden.

3.2.3. Figuras de mérito

Si bien ya se ha clasificado la no linealidad de un DUT, es necesario también saber la diferencia entre la salida de un sistema no lineal y la salida lineal que se busca. Además, estas pruebas son poco realistas, ya que normalmente tendremos una señal multitono con mayor ancho de banda. Por ello, nace la obligación de definir distintas figuras de mérito como son la relación de potencia en el canal adyacente (ACPR), la magnitud del vector error (EVM) y el error cuadrático medio normalizado (NMSE).

ACPR

Esta figura nos da una medida de cuánto afecta la potencia generada en los canales adyacentes con respecto al canal de interés. En este caso, la entrada no será una señal de dos tonos sino una señal modulada, y su valor determinará cuánta potencia emite el DUT fuera del ancho de banda de entrada. Por tanto, es una medida de la distorsión fuera de banda.

Este valor viene dado para el canal adyacente inferior y para el superior, respectivamente, por

$$ACPR_{-1}(dB) = 10 \log_{10} \frac{\int_{f_{L1}}^{f_{L2}} s_y(f) df}{\int_{f_{C1}}^{f_{C2}} s_y(f) df}, \quad ACPR_{+1}(dB) = 10 \log_{10} \frac{\int_{f_{U1}}^{f_{U2}} s_y(f) df}{\int_{f_{C1}}^{f_{C2}} s_y(f) df}, \quad (3.6)$$

donde $[f_{L1}, f_{L2}]$ y $[f_{U1}, f_{U2}]$ determinan el ancho de banda de los canales adyacentes inferior y superior, respectivamente, y $[f_{C1}, f_{C2}]$ es el ancho de banda de canalización.

ACEPR

Esta figura es parecida a la anterior, pero su medida relaciona cuánto afecta la potencia señal error $S_e(f)$ entre la señal medida y la señal modulada con respecto al canal de interés. Se define por tanto como:

$$ACEPR_{-1}(dB) = 10 \log_{10} \frac{\int_{f_{L1}}^{f_{L2}} S_e(f) df}{\int_{f_{C1}}^{f_{C2}} s_y(f) df}, \quad ACEPR_{+1}(dB) = 10 \log_{10} \frac{\int_{f_{U1}}^{f_{U2}} S_e(f) df}{\int_{f_{C1}}^{f_{C2}} s_y(f) df}, \quad (3.7)$$

EVM

En este caso, se representan los símbolos recibidos en la constelación de la señal, y se comparan con los símbolos transmitidos. La medida de error será la diferencia entre el símbolo recibido \hat{x}_i y el transmitido x_i . Esta diferencia viene caracterizada por la distorsión en banda que sufre la señal en recepción. Por tanto, el

EVM será una medida de la distorsión en banda de la señal. Su expresión será, para N símbolos transmitidos y recibidos

$$EVM[\%] = \sqrt{\frac{\sum_{i=1}^N |\hat{x}_i - x_i|^2}{\sum_{i=1}^N |x_i|^2}} \times 100 \quad (3.8)$$

En la práctica, se intentará que el EVM sea lo más cercano a cero posible, ya que cuánto más pequeño es el EVM más cercanos son los símbolos transmitidos y recibidos.

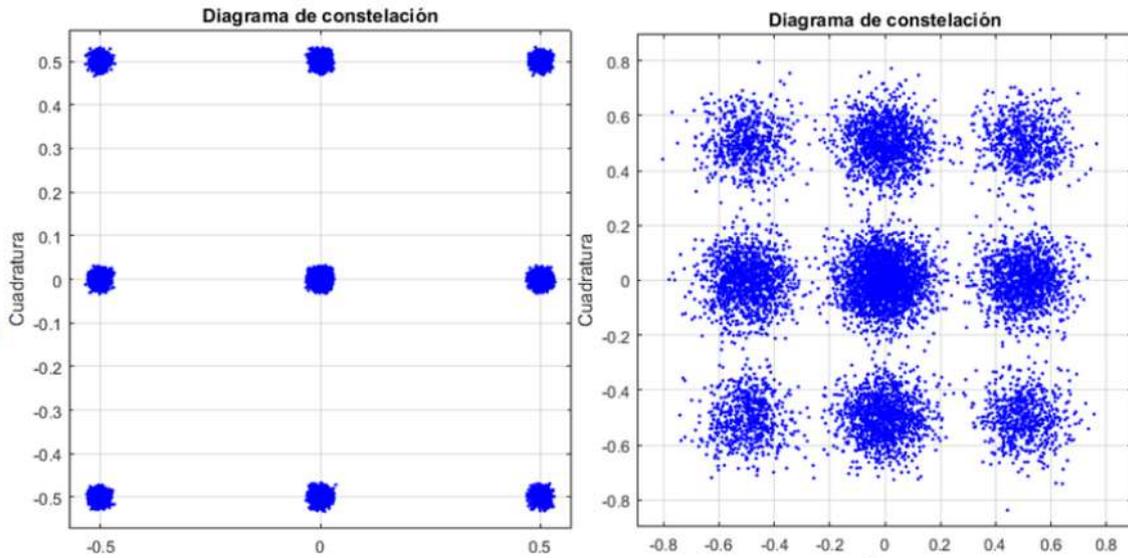


Figura 3.4: Símbolos transmitidos y recibidos, respectivamente.

NMSE

Con esta medida se pretende calcular la diferencia entre la señal real que se obtiene a la salida con la que se obtendría si el sistema fuese ideal y, por consiguiente, lineal. Esta medida no diferencia, como las anteriores, entre distorsión fuera de banda y distorsión en banda, sino que pretende medir la calidad de la señal en términos absolutos. Su expresión es, para N muestras

$$NMSE[dB] = 20 \log \frac{\sum_{k=1}^N |y_{lineal}(k) - y_{real}(k)|}{\sum_{k=1}^N |y_{lineal}(k)|} \quad (3.9)$$

3.3 Series de Volterra

Para entender las series de Volterra, es conveniente primero hablar de las aproximaciones de Taylor, cuyo objetivo es descomponer una salida no lineal de un sistema $y(t)$ como suma de polinomios de la entrada escalados por unos coeficientes a_n . Así, la función de Taylor de una entrada $x(t)$ se desarrollaría:

$$y(t) = f(x(t)) = \sum_{n=0}^{\infty} a_n \cdot x^n(t) = a_0 + a_1 x(t) + \dots + a_n x^n(t) \quad (3.10)$$

Si además, para hacer el modelo más realista, contamos con que existen unos ciertos retrasos τ en la función, es decir, que la salida del sistema depende de la entrada con un retraso τ , la salida del sistema quedaría como:

$$y(t) = f(x(t)) = \sum_{n=0}^{\infty} a_n \cdot x^n(t - \tau) = a_0 + a_1 x(t - \tau) + \dots + a_n x^n(t - \tau) \quad (3.11)$$

Sin embargo, en el caso de un amplificador de potencia, este modelo no es realista pues el amplificador presenta memoria, es decir, que en un instante t_1 la salida del sistema no solo depende de $x(t_1)$, sino también de muestras pasadas, y esto no es abordable con una aproximación de Taylor.

Por ello nacen las series de Volterra, que, partiendo de las series de Taylor, muestran la salida de un sistema como:

$$y(t) = \sum_{p=1}^{\infty} y_p(t), \quad (3.12)$$

donde cada término p sigue la siguiente forma:

$$y_p(t) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \dots \int_{-\infty}^{\infty} h_p(\tau_1, \tau_2, \dots, \tau_n) \cdot \prod_{s=1}^n x(t - \tau_s) \cdot \partial \tau_s \quad (3.13)$$

De esta forma, los efectos de memoria se quedan contemplados en los retrasos τ . La expresión (3.13) contiene infinitas combinaciones entre las versiones retrasadas de la señal de entrada $x(t)$, llamados regresores.

Sin embargo, a nivel de computación, para que este sistema sea útil y realizable es necesario que las variables sean discretas, y, como la memoria de un ordenador no es infinita, también es imprescindible truncar la serie. Así, definiendo N como el orden máximo del sistema no lineal y Q como la longitud de la memoria o profundidad, es decir, la cantidad de retrasos de la señal a tener en cuenta, se reescribe la serie de Volterra como:

$$y[k] = \sum_{n=1}^N \sum_{q_1=0}^Q \dots \sum_{q_n=0}^Q h_n(q_1, \dots, q_n) \cdot \prod_{s=1}^n x(k - q_s) \quad (3.14)$$

Esta aproximación a tiempo discreto sirve para señales reales, por lo que se denomina comúnmente serie de Volterra en RF (radiofrecuencia). Sin embargo, para trabajar es más útil hacerlo con muestras de la envolvente compleja de la señal en banda base, por lo que es deseable reescribir la ecuación (3.14) como:

$$\tilde{y}[k] = \sum_{n \text{ impar}}^N \sum_{q_1=0}^Q \dots \sum_{q_{2p+1}=0}^{Q_{2p+1}} h_{2p+1}(q_1, \dots, q_{2p+1}) \cdot \prod_{s=1}^{p+1} \tilde{x}[k - q_s] \cdot \prod_{r=p+2}^{2p+1} \tilde{x}^*[k - q_r], \quad (3.15)$$

donde $n=2p+1$ es el orden n -ésimo de no linealidad. Este modelo se denomina Full Volterra (FV), ya que incorpora todos los coeficientes posibles, con las únicas aproximaciones de que la señal es discreta y truncada.

En la práctica, trabajar con este modelo se hace muy arduo ya que la carga computacional que conlleva la cantidad de coeficientes a tener en cuenta es excesiva. De manera que, a lo largo del tiempo, se han ido obteniendo simplificaciones de este modelo, más útiles para aplicarlos a sistemas no lineales.

3.3.1. Memory Polinomial (MP)

Esta simplificación omite la contribución de los términos cruzados a la señal de salida $y(t)$, es decir, supone que todas las versiones retrasadas de la entrada tienen un retraso común q . Esto supone que el vector de memorias propio del modelo FV quede reducido a un único valor $[q]$. Así, esta aproximación quedaría de la siguiente forma:

$$y_{MP}[k] = \sum_{p=1}^P \sum_{q=0}^Q h_p[q] x[k - q] |x[k - q]|^{p-1}, \quad (3.16)$$

donde se ha omitido la notación $\tilde{y}[k]$ o $\tilde{x}[k]$ para las envolventes complejas por simplicidad, de igual manera que se hará para el caso siguiente.

3.3.2. Generalized Memory Polinomial (GMP)

Este modelo es una ampliación del MP, pero teniendo también en cuenta los términos cruzados. Podemos dividir este modelo en tres partes, donde la primera quedará igual que el MP, y la segunda y la tercera son combinaciones de la envolvente retrasada y adelantada, respectivamente, m muestras con respecto a la señal. El modelo GMP quedará entonces como:

$$\begin{aligned}
y_{GMP}[k] = & \sum_{p=1}^P \sum_{q=0}^Q h_p[q]x[k-q]|x[k-q]|^{2p} \\
& + \sum_{p=1}^P \sum_{q=0}^Q \sum_{m=0}^M h_p[q,m]x[k-q]|x[k-q-m]|^{2p} \\
& + \sum_{p=1}^P \sum_{q=0}^Q \sum_{m=0}^M h_p[q,m]x[k-q]|x[k-q+m]|^{2p} \quad (3.17)
\end{aligned}$$

3.4 Expresión en forma matricial

A nivel computacional, es más conveniente la definición de los modelos anteriores mediante matrices. Además, de esta forma no se pierden las propiedades algebraicas de las ecuaciones que explican el modelo.

Como ejemplo, se va a reescribir en forma matricial el modelo GMP de la ecuación (3.17), que es el que se utiliza en el modelaje de MIMO posteriormente.

Lo primero que se necesita para ello es definir P, Q y M, y para que no quede demasiado amplio, se tomará P=2, Q=1 y M=1. Así, la ecuación (3.17) quedaría como:

$$\begin{aligned}
y_{GMP}[k] = & h_1[0]x[k]|x[k]|^2 + \\
& h_1[1]x[k-1]|x[k-1]|^2 + \\
& h_2[0]x[k]|x[k]|^2 + \\
& h_2[1]x[k-1]|x[k-1]|^2 + \\
& h_1[0,0]x[k]|x[k]|^2 + h_1[0,1]x[k]|x[k-1]|^2 + h_1[1,0]x[k]|x[k-1]|^2 + h_1[1,1]x[k-1]|x[k-2]|^2 \\
& + h_2[0,0]x[k]|x[k]|^4 + h_2[0,1]x[k]|x[k-1]|^4 + h_2[1,0]x[k]|x[k-1]|^4 \\
& + h_2[1,1]x[k-1]|x[k-2]|^4 + \\
& h_1[0,0]x[k]|x[k]|^2 + h_1[0,1]x[k]|x[k-1]|^2 + h_1[1,0]x[k]|x[k-1]|^2 + h_1[1,1]x[k-1]|x[k]|^2 \\
& + h_2[0,0]x[k]|x[k]|^4 + h_2[0,1]x[k]|x[k-1]|^4 + h_2[1,0]x[k]|x[k-1]|^4 \\
& + h_2[1,1]x[k-1]|x[k]|^4
\end{aligned} \quad (3.18)$$

Se puede observar que hay regresores repetidos aún utilizando unos límites P, Q y M bastante bajos.

Además, como puede verse en el ejemplo, la operación tiene forma de producto escalar. Así, se define el vector de coeficientes \mathbf{h}_p como:

$$\mathbf{h}_p = [h_1[0], \dots, h_1[Q], h_p[0], \dots, h_p[Q], \dots, h_1[0,0], h_1[0, Q], h_p[0,0], \dots, h_p[Q, M]] \quad (3.19)$$

Además, para que se vea más claramente en el código qué coeficiente le corresponde a cada regresor se han dividido los coeficientes en tres partes: los coeficientes 'a' corresponden al primer sumando de la ecuación (3.17), los coeficientes 'b' al segundo sumando y los coeficientes 'c' al tercero.

Por otro lado, los límites P del modelo GMP pasarán a llamarse K con el subíndice del sumando al que corresponden. Por ejemplo, los límites P del primer sumando pasarán a ser K_a . Lo mismo ocurre con los límites Q, que pasarán a llamarse L. así, el vector de coeficientes quedará como:

$$\mathbf{h}_p = [a[K_a, L_a], b[K_b, L_b, M_b], c[K_c, L_c, M_c]] \quad (3.20)$$

Por otro lado, la matriz de regresores \mathbf{X} se define como:

$$\mathbf{X} = \begin{bmatrix} x[0] & x[-1] & \dots & x[-Q] & x[0]|x[0]|^2 & \dots & x[-Q]|x[-Q]|^2 & \dots \\ x[1] & x[0] & \dots & \vdots & x[1]|x[1]|^2 & \dots & \vdots & \dots \\ x[2] & x[1] & \dots & x[-1] & x[2]|x[2]|^2 & \dots & x[-1]|x[-1]|^2 & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots & \vdots & \dots \\ \vdots & \vdots & \dots & \vdots & \vdots & \dots & \vdots & \dots \\ x[N] & x[N-1] & \dots & \dots & x[N]|x[N]|^2 & \dots & \dots & \dots \\ & & & & & & x[N-Q]|x[N-Q]|^2 & \dots \end{bmatrix} \quad (3.21)$$

Cabe destacar que los índices negativos corresponden a las muestras tomadas antes del instante que se ha considerado como inicio, para así contemplar los efectos de la memoria. También es importante señalar que N corresponde al número de muestras que se han tomado.

3.5 Predistorsión digital

La predistorsión digital es una técnica que pretende contrarrestar los efectos no lineales de un amplificador de potencia, modificando a cada iteración la señal de entrada para que así la salida conjunta del sistema sea equivalente a la de un sistema lineal.

Un predistorsionador digital aplica distorsión inversa a la señal de entrada del amplificador para disminuir la distorsión generada por el amplificador. Una idea gráfica de lo que ocurre se puede observar en la figura 3.5.

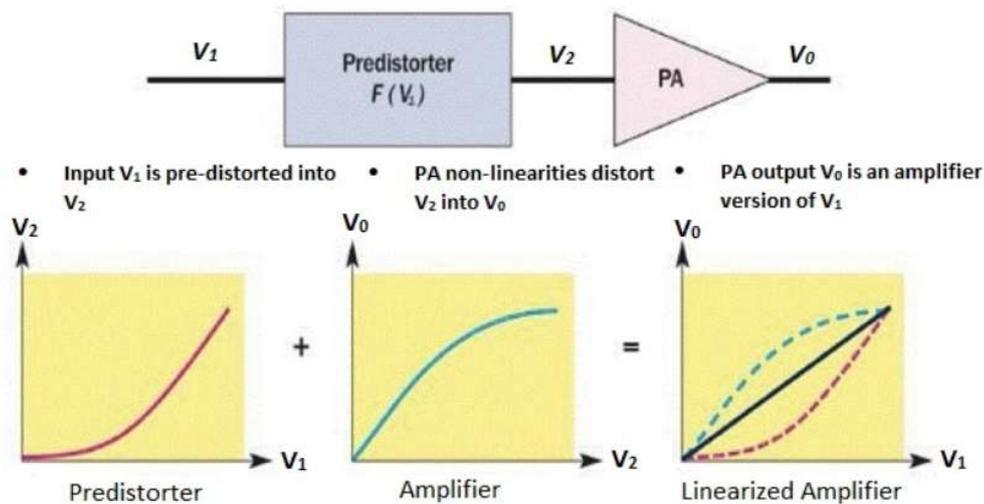


Figura 3.5: Esquema de predistorsión digital.

Hay varias formas de llevar a cabo la predistorsión, pero este trabajo se centra en una de ellas: la arquitectura de aprendizaje directo (DLA).

La idea consiste en minimizar el vector error, teniendo en cuenta que, si el sistema fuese lineal, la única diferencia entre la señal de entrada y la de salida sería la ganancia del amplificador. Por ello, lo que se pretende en cada iteración es eliminar el error mediante la función de costes. La estimación del error en cada iteración, $\hat{e} = U w_i$, se resta a la señal de entrada para conseguir un sistema conjunto lo más lineal posible, tal como se observa en la figura 3.6.

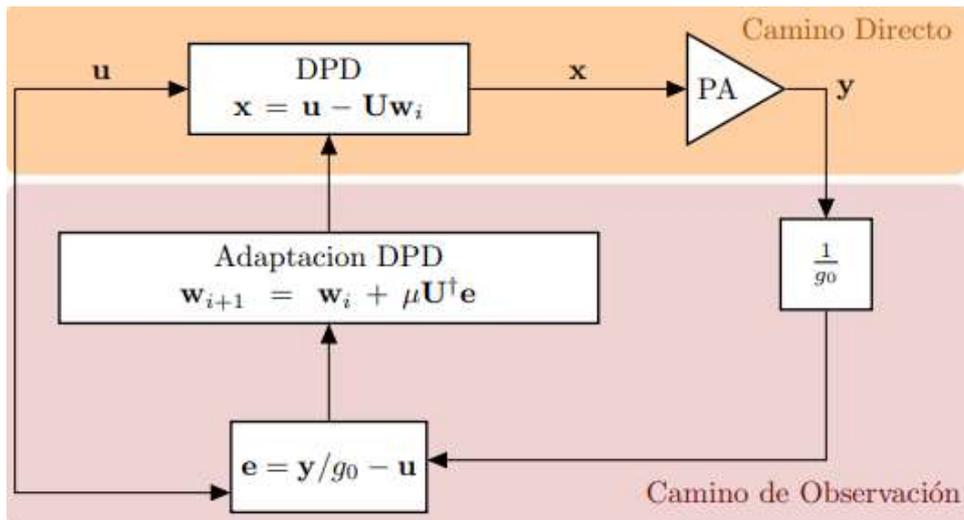


Figura 3.6: Técnica de DLA.

En el caso de un sistema MIMO 2x2, todos los bloques de la figura 3.6 tendrán dos señales, ya que en este caso tendremos dos señales de entrada y dos de salida. Esta idea se muestra en la figura 3.7.

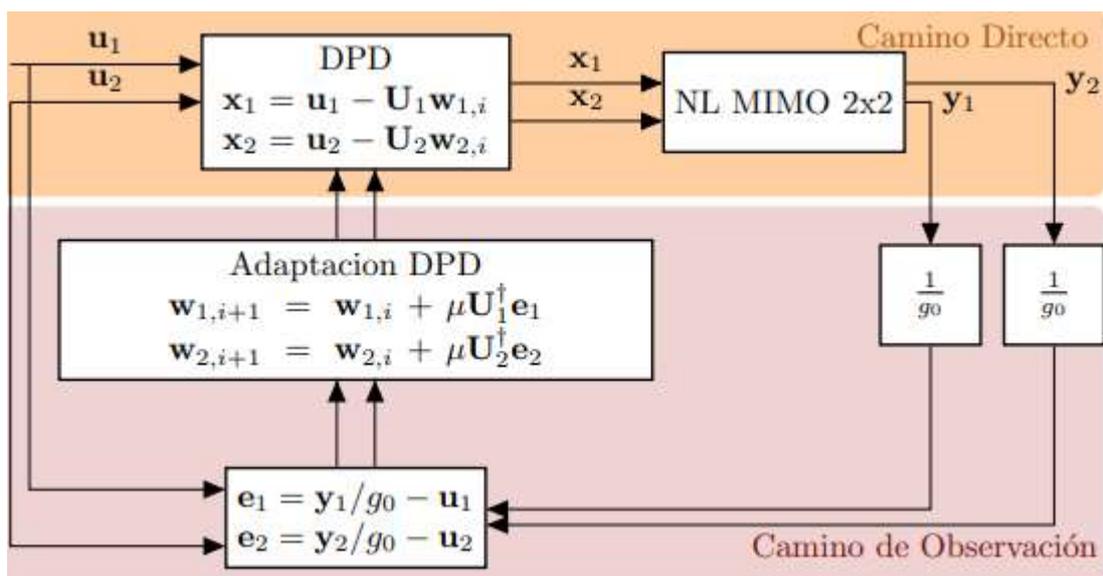


Figura 3.7: Técnica de aprendizaje directo para sistema MIMO 2x2.

Donde el bloque *NL MIMO 2x2* se refiere al bloque no lineal propio de un sistema MIMO 2x2, que se explica en la sección 4.2. de este trabajo y se ilustra en la figura 4.2.

4 MODELOS DE MÚLTIPLES ENTRADAS

Este capítulo está centrado en el artículo de investigación titulado *Behavioral Modeling and Linearization of Crosstalk and Memory Effects in RF MIMO Transmitters*, publicado por Shoaib Amin en 2014. En este artículo se presentan los cuatro modelos que se utilizan posteriormente en el proyecto, que son cuatro alternativas a las Series de Volterra explicadas en el capítulo anterior, ya que estas solo son capaces de explicar los efectos de acoplamiento en sistemas SISO, de una única entrada y una salida. Estos modelos parten del Generalized Memory Polinomial (GMP) de la Serie de Volterra, adaptándolo según el tipo de acoplamiento que haya.

Además, en este artículo también se detalla el montaje experimental que se ha utilizado en este proyecto.

4.1 Modelos bidimensionales para el caso de doble banda concurrente

Para entender mejor el código de este trabajo, es necesario hablar de modelos de doble banda concurrente, ya que se utiliza parte de sus expresiones.

En primer lugar, cabe destacar que una señal en doble banda concurrente es aquella en la que hay dos tonos distanciados en frecuencia y con distinto ancho de banda dentro del mismo espectro, tal como se muestra en la figura 4.1.

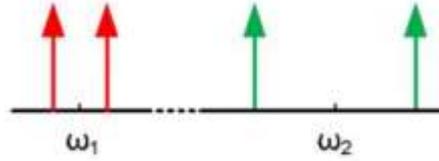


Figura 4.1: Espectro de una señal de doble banda concurrente

Por tanto, la idea es modificar el modelo GMP de la sección 3.3, teniendo en cuenta dos señales de entrada separadas en frecuencia. El resultado obtenido se denomina 2D-GMP y se formula como:

$$\begin{aligned}
 y_1[n] = & \sum_{k \in K_a^{IB}} \sum_{l \in L_a^{IB}} a_{kl}^{IB} x_1[n-l] |x_1[n-l]|^k + \\
 & \sum_{k \in K_b^{IB}} \sum_{l \in L_b^{IB}} \sum_{m \in M_b^{IB}} b_{klm}^{IB} x_1[n-l] |x_1[n-l-m]|^k + \\
 & \sum_{k \in K_c^{IB}} \sum_{l \in L_c^{IB}} \sum_{m \in M_c^{IB}} c_{klm}^{IB} x_1[n-l] |x_1[n-l+m]|^k + \\
 & \sum_{k \in K_a^{CB}} \sum_{l \in L_a^{CB}} a_{kl}^{CB} x_1[n-l] |x_2[n-l]|^k + \\
 & \sum_{k \in K_b^{CB}} \sum_{l \in L_b^{CB}} \sum_{m \in M_b^{CB}} b_{klm}^{CB} x_1[n-l] |x_2[n-l-m]|^k +
 \end{aligned}$$

$$\begin{aligned}
& \sum_{k \in K_c^{CB}} \sum_{l \in L_c^{CB}} \sum_{m \in M_c^{CB}} c_{klm}^{CB} x_1[n-l] |x_2[n-l+m]|^k \\
+ & \sum_{k_1 \in K_a^{MX}} \sum_{k_2 \in K_a^{MX}} \sum_{l \in L_a^{MX}} a_{k_1 k_2}^{MX} l x_1[n-l] |x_1[n-l]|^{k_1} |x_2[n-l]|^{k_2}
\end{aligned} \tag{4.1}$$

$$\begin{aligned}
y_2[n] = & \sum_{k \in K_a^{IB}} \sum_{l \in L_a^{IB}} a_{kl}^{IB} x_2[n-l] |x_2[n-l]|^k + \\
& \sum_{k \in K_b^{IB}} \sum_{l \in L_b^{IB}} \sum_{m \in M_b^{IB}} b_{klm}^{IB} x_2[n-l] |x_2[n-l-m]|^k + \\
& \sum_{k \in K_c^{IB}} \sum_{l \in L_c^{IB}} \sum_{m \in M_c^{IB}} c_{klm}^{IB} x_2[n-l] |x_2[n-l+m]|^k + \\
& \sum_{k \in K_a^{CB}} \sum_{l \in L_a^{CB}} a_{kl}^{CB} x_2[n-l] |x_1[n-l]|^k + \\
& \sum_{k \in K_b^{CB}} \sum_{l \in L_b^{CB}} \sum_{m \in M_b^{CB}} b_{klm}^{CB} x_2[n-l] |x_1[n-l-m]|^k + \\
& \sum_{k \in K_c^{CB}} \sum_{l \in L_c^{CB}} \sum_{m \in M_c^{CB}} c_{klm}^{CB} x_2[n-l] |x_1[n-l+m]|^k \\
+ & \sum_{k_2 \in K_a^{MX}} \sum_{k_1 \in K_a^{MX}} \sum_{l \in L_a^{MX}} a_{k_2 k_1}^{MX} l x_2[n-l] |x_2[n-l]|^{k_2} |x_1[n-l]|^{k_1}
\end{aligned}$$

donde IB significa distorsión intra banda, es decir, la distorsión de la señal consigo misma, por lo que no aparecen términos cruzados; CB significa distorsión de banda cruzada, es decir, los términos de una banda que afectan a la otra; y MX es banda mixta, una mezcla entre los efectos de IB y CB.

4.2 Montaje experimental

Como ya se ha citado anteriormente, en este trabajo se utiliza una señal tipo MIMO 2x2. Por tanto, se trata de dos señales independientes, $x_1(t)$ y $x_2(t)$, que se transmiten al mismo tiempo. Esto hace necesario la utilización de dos amplificadores de potencia (PA), que amplificarán una señal de entrada cada uno, y además una pequeña diafonía o interferencia de la otra señal de entrada. Este acoplamiento se conoce como acoplamiento no lineal. Por otro lado, una vez las señales son amplificadas volverán a acoplarse antes de llegar al receptor. Este acoplamiento será lineal.

En la figura 4.1 se representa más claramente lo explicado anteriormente, donde α , β , δ y γ son las respuestas al impulso de los filtros lineales antes y después del PA, que indican la cantidad de acoplamiento representada entre 0 y 1, y LO es el oscilador local que fuerza que las dos señales Sig 1 y Sig 2 estén a la misma frecuencia.

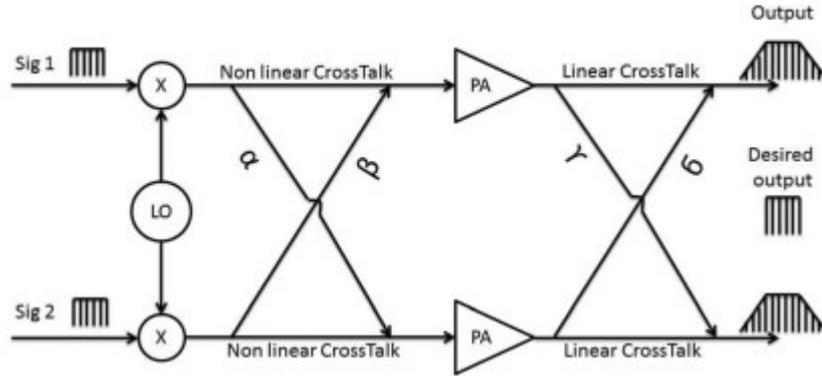


Figura 4.2: Esquema de utilización de un transmisor MIMO

4.3 GMP para transmisores MIMO propuesto por S. Amin

Los modelos que se van a utilizar como base en este proyecto se describen a continuación.

4.3.1 Generalized Memory Polinomial for Linear Crosstalk (GMPLC)

En el modelo GMPLC, como su propio nombre indica, solo se tiene en cuenta el acoplamiento lineal, es decir, si seguimos el esquema de la figura 4.1, $\alpha=0$ y $\beta=0$.

A nivel de ecuaciones, esto supone que las funciones de transferencia no lineales de los PA solo amplifican $x_1(t)$ y $x_2(t)$, respectivamente.

Sin embargo, lo que los diferencia de un sistema SISO, es que las señales una vez que son amplificadas se combinan linealmente entre ellas con unos pesos δ y γ . Cuanto más cercanos a uno sean estos parámetros más acoplada estará la señal, y, por tanto, presentará mayor no linealidad.

Para un sistema MIMO 2x2, la señal de salida y_1 se puede modelar como:

$$y_1[k] = \sum_{p=1}^P \sum_{q=0}^Q \sum_{m=0}^M g_{1,p}[q, m] x_1[k - q] |x_1[k - q - m]|^{2(p-1)} + \sum_{p=1}^P \sum_{q=0}^Q \sum_{m=0}^M g_{2,p}[q, m] x_2[k - q] |x_2[k - q - m]|^{2(p-1)} \quad (4.2)$$

donde P está relacionado con el orden de no linealidad, Q y M los pesos de memoria y en los parámetros 'g' están incluidos los acoplamientos lineales.

Se puede observar en este caso que la señal de salida es una combinación lineal de los modelos GMP de las Series de Volterra de dos señales de entrada $x_1(t)$ y $x_2(t)$. Es decir, el modelo GMPLC solo contiene términos no lineales SISO provenientes de un amplificador combinados linealmente con términos no lineales SISO provenientes del otro amplificador.

4.3.2 Generalized Memory Polinomial for NonLinear Crosstalk (GMPNLC)

En el caso del modelo GMPNLC, las señales x_1 y x_2 se combinan antes del amplificador. Esto supone que la señal que pasa por la función de transferencia no lineal del PA está ya acoplada, por lo que el modelo GMPLC se vuelve insuficiente.

Para compensar los efectos del acoplamiento no lineal es necesario que el modelo incluya términos cruzados entre las señales de entrada x_1 y x_2 , además de la combinación lineal de las funciones de transferencia de

ambos amplificadores.

Para un sistema MIMO2x2, la señal de salida y_1 se puede modelar como:

$$\begin{aligned}
 y_1[k] = & \sum_{p=1}^P \sum_{n=1}^{P-p+1} \sum_{q=0}^Q \sum_{m=0}^M g_{1,p,n}[q,m] x_1[k-q] |x_1[k-q-m]|^{2(p-1)} |x_2[k-q-m]|^{2(n-1)} \\
 & + \sum_{p=1}^P \sum_{n=1}^{P-p+1} \sum_{q=0}^Q \sum_{m=0}^M g_{2,p,n}[q,m] x_2[k-q] |x_2[k-q-m]|^{2(p-1)} |x_1[k-q-m]|^{2(n-1)}
 \end{aligned}
 \tag{4.3}$$

Los acoplamientos lineales y no lineales están incluidos en los términos ‘g’.

4.3.3 Extended Generalized Memory Polynomial for NonLinear Crosstalk (EGMPNLC)

Este modelo es una extensión del modelo anterior. La única diferencia entre ambos modelos es que el EGMPNLC incluye muchos más términos cruzados que el GMPNLC.

Para un orden de no linealidad p , las funciones base del modelo EGMPNLC se muestran en a continuación, donde $L_1 = [k - q]$ y $L_2 = [k - q - m]$

$$\begin{aligned}
 & x_1(L_1) |x_1(L_2)|^{(p-1)} \\
 & x_1(L_1) |x_2(L_2)|^{(p-1)} \\
 & x_2(L_1) |x_1(L_2)|^{(p-1)} \\
 & x_2(L_1) |x_2(L_2)|^{(p-1)} \\
 & x_1(L_1) |x_1(L_2)|^{(p-3)} x_1(L_2) x_2^*(L_2) \\
 & x_1(L_1) |x_1(L_2)|^{(p-3)} x_1^*(L_2) x_2(L_2) \\
 & x_1(L_1) |x_2(L_2)|^{(p-3)} x_1(L_2) x_2^*(L_2) \\
 & x_1(L_1) |x_2(L_2)|^{(p-3)} x_1^*(L_2) x_2(L_2) \\
 & x_1(L_1) |x_1(L_2)|^{(p-1)/2} |x_2(L_2)|^{(p-1)/2} \\
 & x_2(L_1) |x_1(L_2)|^{(p-1)/2} |x_2(L_2)|^{(p-1)/2} \\
 & x_2(L_1) |x_1(L_2)|^{(p-3)} x_1(L_2) x_2^*(L_2) \\
 & x_2(L_1) |x_1(L_2)|^{(p-3)} x_1^*(L_2) x_2(L_2) \\
 & x_2(L_1) |x_2(L_2)|^{(p-3)} x_1(L_2) x_2^*(L_2) \\
 & x_2(L_1) |x_2(L_2)|^{(p-3)} x_1^*(L_2) x_2(L_2)
 \end{aligned}
 \tag{4.4}$$

4.3.4 Parallel Hammerstein for a MIMO Transmitter 2x2 (PH2x2)

Este modelo es el más sencillo, pues se trata de una extensión del modelo SISO PH, para adaptarlo a una situación MIMO 2x2. Las ecuaciones que lo modelan se muestran a continuación:

$$\begin{aligned}
y_1 = & \sum_{p=1}^P \sum_{q=1}^{P-p+1} \sum_{m=0}^M g_{1,p,q,m}[q, m] x_1[k-m] |x_1[k-m]|^{2(p-1)} |x_2[n-m]|^{2(q-1)} \\
& + \sum_{p=1}^P \sum_{q=1}^{P-p+1} \sum_{m=0}^M g_{2,p,q,m}[q, m] x_2[k-m] |x_2[k-m]|^{2(p-1)} |x_1[n-m]|^{2(q-1)}
\end{aligned} \tag{4.5}$$

Este modelo también tiene términos cruzados, y su principal diferencia con el modelo GMPNLC es que solo existe un peso de memoria, igual para las dos señales x_1 y x_2 .

4.3.5 Mejoras en la implementación del modelo

En este trabajo se utiliza la expresión matricial de los cuatro modelos propuestos por S. Amin pero con la diferencia de que en los modelos GMPLC y GMPNLC también se incluye la parte ‘c’ de IB y CB del modelo 2D-GMP de la sección 4.1.

Además, como se explicará en el capítulo 6 de este trabajo, en una de las propuestas de mejora del modelo inicial se incluirán términos que dependen de la señal imagen, es decir, $x_1^*(k)$ y $x_2^*(k)$. Esto se explicará con más detalle en la sección 6.2., concretamente en las pruebas 2 y 3.

4.4 Procedimiento para identificar el modelo: la regresión

En un sistema MIMO 2x2, las señales de salida y_1 y y_2 se pueden escribir en forma matricial como:

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} X_1 & 0 \\ 0 & X_2 \end{bmatrix} \begin{bmatrix} h_1 \\ h_2 \end{bmatrix} + \begin{bmatrix} e_1 \\ e_2 \end{bmatrix}$$

donde X_1 y X_2 son las matrices de regresión para los canales 1 y 2 respectivamente, y h_1 y h_2 son los coeficientes del modelo. Las señales e_1 y e_2 son señales de ruido que se asumen incorreladas y de media cero, además de errores de modelaje, ya que las predicciones del modelo pueden contener algún error a tener en cuenta.

De esta forma se consigue que y_1 e y_2 puedan ser estimadas independientemente, ya que solo dependen del comportamiento de su propio canal.

Con la regresión lo que se pretende es estimar la relación entre las variables y_1 e y_2 , y las variables h_1 y h_2 .

Como se trata de una estimación, se emplea la siguiente notación:

$$\hat{\mathbf{y}} = \mathbf{X} \hat{\mathbf{h}} \tag{4.6}$$

donde $\hat{\mathbf{y}} = \mathbf{y} + \mathbf{e}$. Por tanto, lo que se pretende con la regresión es encontrar un vector de coeficientes $\hat{\mathbf{h}}$ que minimice el vector error. Lo más utilizado en este caso es aplicar mínimos cuadrados (LS), es decir, minimizar la suma residual de cuadrados. La aplicación queda como:

$$\hat{\mathbf{h}}_{LS} = \mathbf{X}^\dagger \mathbf{y} \tag{4.7}$$

donde \mathbf{X}^\dagger es la pseudo-inversa de Moore-Penrose, que se define como:

$$\mathbf{X}^\dagger = (\mathbf{X}^H \mathbf{X})^{-1} \mathbf{X}^H \tag{4.8}$$

donde el superíndice H denota la matriz hermítica.

5 RESULTADOS CON AMPLIFICADOR SINTÉTICO

Para comprobar antes de ir al laboratorio que todas las funciones en Matlab funcionaban correctamente, y que el modelo que se estaba planteando y sus respectivas ecuaciones daban buenos resultados en una situación ideal, se planteó el uso de un amplificador sintético. Este amplificador no es más que una función de Matlab cuyos datos de entrada son la señal de entrada $x(t)$; unos parámetros α y γ que influirán en el cálculo de la ganancia no lineal del amplificador y en la memoria incluida en señal de salida, respectivamente; y la SNR considerada para las señales a la salida del amplificador. Por su parte, la señal de salida se calculará como la multiplicación de la ganancia y la señal de entrada, considerando la suma de diferentes versiones retrasadas de la señal ponderadas por los pesos γ , a lo que se le suma un ruido aleatorio. El código de la función del amplificador sintético se presenta en el anexo B.3.

Además de permitir una primera toma de contacto con las ecuaciones y funciones que se iban a utilizar posteriormente en laboratorio, el uso de este amplificador aporta otras ventajas. Por un lado, permite modificar el acoplamiento de las señales de entrada $x_1(t)$ y $x_2(t)$, y también entre las de salida $y_1(t)$ y $y_2(t)$. Por otro lado, permite modificar los parámetros del amplificador a nuestro gusto y comprobar así en qué situación funciona mejor el montaje.

En cuanto al acoplamiento, en este trabajo, y únicamente para el modelado, se probó con acoplamientos de 10 y 30 dB. Para la predistorsión se eligió el que daba mejores prestaciones de los dos.

En el siguiente apartado se describen las distintas pruebas de modelado que se hicieron con el amplificador sintético.

5.1. Modelado

Para el modelado se realizaron dos pruebas. En la primera de ellas, se probó con el mayor orden no lineal que permitía cada modelo por el método de mínimos cuadrados, es decir, realizando la pseudoinversa de la matriz de medida incluyendo todos los regresores del modelo.

La segunda prueba consistió en configurar a los cuatro modelos el mismo orden no lineal, y como por el método de la pseudoinversa en algunos de los casos Matlab se quedaba sin memoria, se utilizó una técnica de selección de componentes basada en el algoritmo DOMP. Este método basa su funcionamiento en la idea de que no todos los regresores del modelo aportan al resultado final, y que por tanto hay unos más importantes que otros. Por ello, este método selecciona entre todos los regresores aquellos que considera más importantes y es únicamente a estos a los que se le realiza la pseudoinversa. Por tanto, al utilizar únicamente algunos regresores, es más complicado que Matlab se quede sin memoria.

El código de Matlab con el que se implementa tanto las pruebas de modelado como la generación de los regresores para los diferentes casos está disponible en el Anexo B.

5.1.1. Prueba por el método de mínimos cuadrados

En este caso, la configuración que se utilizó para cada modelo se detalla en la Tabla 5.1.

<i>Conjunto</i>	GMPLC	PH 2x2	GMPNLC	EGMPNLC
<i>Ka</i>	[0 1 2 3 4 5 6 7]	[0 1 2 3 4 5]	[0 1 2 3 4 5]	[]
<i>La</i>	[5 3 1 1 1 1 1 1]	[5 3 1 1 1 1]	[5 3 1 1 1 1]	[]
<i>Kb</i>	[1 2 3 4 5 6]	[1 2 3 4 5]	[1 2 3 4 5]	[0 1 2 3 4 5 6 7]
<i>Lb</i>	[1 1 1 1 1 1]	[1 1 1 1 1]	[1 1 1 1 1]	[1 1 1 1 1 1 1]
<i>Mb</i>	[1 1 1 1 1 1]	[1 1 1 1 1]	[1 1 1 1 1]	[1 1 1 1 1 1 1]
<i>Kc</i>	[1 2 3 4 5 6]	[1 2 3 4 5]	[1 2 3 4 5]	[0 1 2 3 4 5 6 7]
<i>Lc</i>	[1 1 1 1 1 1]	[1 1 1 1 1]	[1 1 1 1 1]	[1 1 1 1 1 1 1]
<i>Mc</i>	[1 1 1 1 1 1]	[1 1 1 1 1]	[1 1 1 1 1]	[1 1 1 1 1 1 1]

Tabla 5.1: Orden no lineal y profundidad de memoria de los diferentes modelos en la prueba con LS.

Los resultados obtenidos son distintos para cada acoplamiento.

En primer lugar, se muestran los datos para un acoplamiento de 30 dB, tanto de ACEPR para el peor caso entre los dos canales adyacentes como de NMSE. Como los resultados obtenidos para las salidas y_1 y y_2 eran bastante similares, se ha cogido para cada caso de las Tablas 5.2. y 5.3. la salida con peores prestaciones. Para cada modelo, se diferencian cuatro casos: cuando existe acoplamiento lineal y no lineal (NL&LC), cuando solo existe lineal (LC), cuando solo existe no lineal (NLC), y cuando no existe ningún acoplamiento (NC).

Modelo	Número de coeficientes	NL&LC	NLC	LC	NC
GMPLC	80/80	-51.53 dB	-52.12 dB	-55.88 dB	-55.79 dB
PH 2x2	152/152	-48.24 dB	-48.17 dB	-48.47 dB	-47.82 dB
GMPNLC	104/104	-53.77 dB	-53.14 dB	-55.30 dB	-52.14 dB
EGMPNLC	240/240	-50.20 dB	-48.61 dB	-50.18 dB	-48.87 dB

Tabla 5.2: NMSE para los distintos modelos y acoplamiento de 30 dB con LS.

Modelo	NL&LC	NLC	LC	NC
GMPLC	-61.96 dB	-61.18 dB	-66.96 dB	-64.6 dB
PH 2x2	-61.31 dB	-60.98 dB	-62.83 dB	-59.06 dB
GMPNLC	-61.74 dB	-61.32 dB	-63.44 dB	-59.07 dB
EGMPNLC	-58.81 dB	-58.31 dB	-58.07 dB	-56.81 dB

Tabla 5.3: ACEPR para los distintos modelos y acoplamiento de 30 dB con LS.

Por otro lado, para un acoplamiento de 10 dB, se obtienen los siguientes datos:

Modelo	Número de coeficientes	NL&LC	NLC	LC	NC
GMPLC	80/80	-37.79 dB	-36.58 dB	-55.81 dB	-55.79 dB
PH 2x2	152/152	-34.61 dB	-34.77 dB	-46.32 dB	-47.82 dB
GMPNLC	104/104	-41.28 dB	-41-07 dB	-51.31 dB	-52.14 dB
EGMPNLC	240/240	-47.54 dB	-48.20 dB	-42.27 dB	-48.87 dB

Tabla 5.4: NMSE para los distintos modelos y acoplamiento de 10 dB con LS.

Modelo	NL&LC	NLC	LC	NC
GMPLC	-47.71 dB	-46.91 dB	-64.52 dB	-64.6 dB
PH 2x2	-44.10 dB	-44.31 dB	-59.80 dB	-59.06 dB
GMPNLC	-51.78 dB	-50.36 dB	-61.48 dB	-59.07 dB
EGMPNLC	-56.52 dB	-57.40 dB	-53.17 dB	-56.81 dB

Tabla 5.5: ACEPR para los distintos modelos y acoplamiento de 10 dB con LS.

La información más interesante que se obtiene de las Tablas 5.2-5.5 es, en primer lugar, que para todos los modelos, cuanto menor sea el acoplamiento mejores serán los datos obtenidos.

Esto coincide con lo teóricamente establecido, ya que el acoplamiento aporta no linealidad al modelo. Por tanto, cuanto mayor sea el valor del acoplamiento empleado en los acopladores direccionales, más lineal será el sistema y tendrá mejores prestaciones.

Por otro lado, cabe destacar que cuando existe acoplamiento no lineal, es decir, en las dos primeras columnas de cada tabla, la diferencia entre los dos acoplamientos es mucho más notoria. Esto se debe a que el acoplamiento no lineal afecta mucho más a los resultados del modelo.

Por último, cuando no existe acoplamiento los resultados son iguales en los dos casos, como es obvio.

En cuanto a los modelos, los resultados son buenos para los cuatro. Sin embargo, el más sencillo de implementar, el PH2x2, es el que peores resultados obtiene. El GMPLC es un modelo más robusto aunque lineal, por eso las prestaciones son mejores que en el PH2x2 pero peores que para los modelos GMPNLC y EGMPNLC. Esto se observa más claramente para un acoplamiento de 10 dB.

5.1.2. Prueba por el método DOMP

En este caso, se utiliza la misma configuración para todos los modelos, que es la que se muestra en la Tabla 5.6.

Conjunto	GMPLC	PH 2x2	GMPNLC	EGMPNLC
<i>Ka</i>	[0 1 2 3 4 5 6 7]	[0 1 2 3 4 5 6 7]	[0 1 2 3 4 5 6 7]	[]
<i>La</i>	[5 3 1 1 1 1 1 1]	[5 3 1 1 1 1 1 1]	[5 3 1 1 1 1 1 1]	[]
<i>Kb</i>	[1 2 3 4 5 6 7]	[1 2 3 4 5 6 7]	[1 2 3 4 5 6 7]	[0 1 2 3 4 5 6 7]
<i>Lb</i>	[1 1 1 1 1 1 1]	[1 1 1 1 1 1 1]	[1 1 1 1 1 1 1]	[1 1 1 1 1 1 1]
<i>Mb</i>	[1 1 1 1 1 1 1]	[1 1 1 1 1 1 1]	[1 1 1 1 1 1 1]	[1 1 1 1 1 1 1]
<i>Kc</i>	[1 2 3 4 5 6 7]	[1 2 3 4 5 6 7]	[1 2 3 4 5 6 7]	[0 1 2 3 4 5 6 7]
<i>Lc</i>	[1 1 1 1 1 1 1]	[1 1 1 1 1 1 1]	[1 1 1 1 1 1 1]	[1 1 1 1 1 1 1]
<i>Mc</i>	[1 1 1 1 1 1 1]	[1 1 1 1 1 1 1]	[1 1 1 1 1 1 1]	[1 1 1 1 1 1 1]

Tabla 5.6: Orden no lineal y profundidad de memoria de los diferentes modelos en la prueba con DOMP.

En este caso, solo se obtienen resultados para un acoplamiento de 30 dB, ya que en los resultados anteriores se ha demostrado que es el caso para el que se consigue un mejor ajuste de los modelos.

Modelo	Número de coeficientes	NL&LC	Número de coeficientes	NLC	Número de coeficientes	LC	Número de coeficientes	NC
GMPLC	57/73	-52.84	66/54	-52.68	54/58	-57.92	49/47	-55.04
PH 2x2	35/27	-49.10	26/86	-49.61	27/75	-53.03	44/38	-46.41
GMPNLC	150/99	-56.37	142/152	-54.02	204/162	-57.16	214/185	-54.27
EGMPNLC	90/92	-50.56	64/61	-51.01	144/106	-49.42	130/74	-50.00

Tabla 5.7: NMSE, en dB, para los distintos modelos y acoplamiento de 30 dB con DOMP.

Modelo	NL&LC	NLC	LC	NC
GMPLC	-62.33 dB	-62.27 dB	-67.21 dB	-64.64 dB
PH 2x2	-62.10 dB	-60.45 dB	-65.41 dB	-58.24 dB
GMPNLC	-67.17 dB	-62.72 dB	-67.31 dB	-65.22 dB
EGMPNLC	-62.14 dB	-62.22 dB	-66.34 dB	-63.83 dB

Tabla 5.8: ACEPR para los distintos modelos y acoplamiento de 30 dB con DOMP.

En este caso, la comparación entre los distintos modelos es más justa, ya que todos parten de una

configuración con el mismo valor máximo para el orden de la no linealidad y la memoria para los regresores, seleccionando el algoritmo DOMP aquellos que son más importantes en cada caso.

Se puede observar que en igualdad de condiciones el que mejores resultados obtiene es el GMPNLC, y el que peores, el PH2x2.

5.2. Predistorsión

En este caso, se utilizará un único modelo como ilustración del funcionamiento para el amplificador sintético. El modelo elegido es el GMPLC ya que obtiene buenos resultados con un número de coeficientes no muy elevado, permitiendo así una demostración sencilla y rápida computacionalmente.

El código de Matlab con el que se implementa el esquema de predistorsión mediante DLA está disponible en el Anexo B.3.

De esta prueba cabe destacar que se emplean dos velocidades de aprendizaje, μ , distintas, para comprobar que se cumple con lo teóricamente establecido, que cuanto más cercano a uno sea la velocidad de aprendizaje, más rápidamente se encontrará tanto el ACPR como el NMSE óptimo del modelo.

Los datos obtenidos se muestran en las siguientes gráficas.

Para las gráficas de ACPR, es importante señalar que se muestra en línea continua el dato de ACPR por la izquierda, es decir, $ACPR_{-1}$, y en línea discontinua el dato de ACPR por la derecha, es decir, $ACPR_{+1}$.

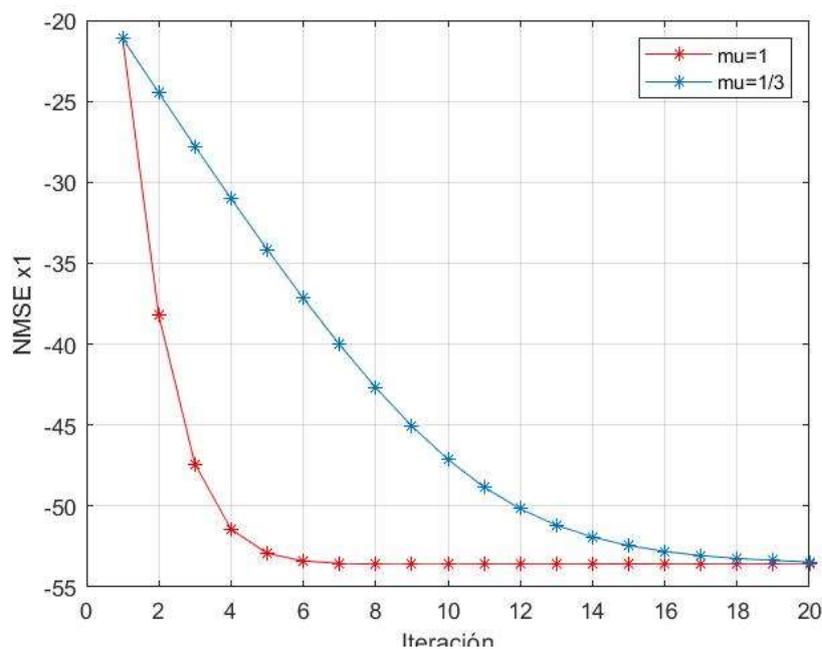


Figura 5.1: NMSE de x_1 para distintas velocidades de aprendizaje.

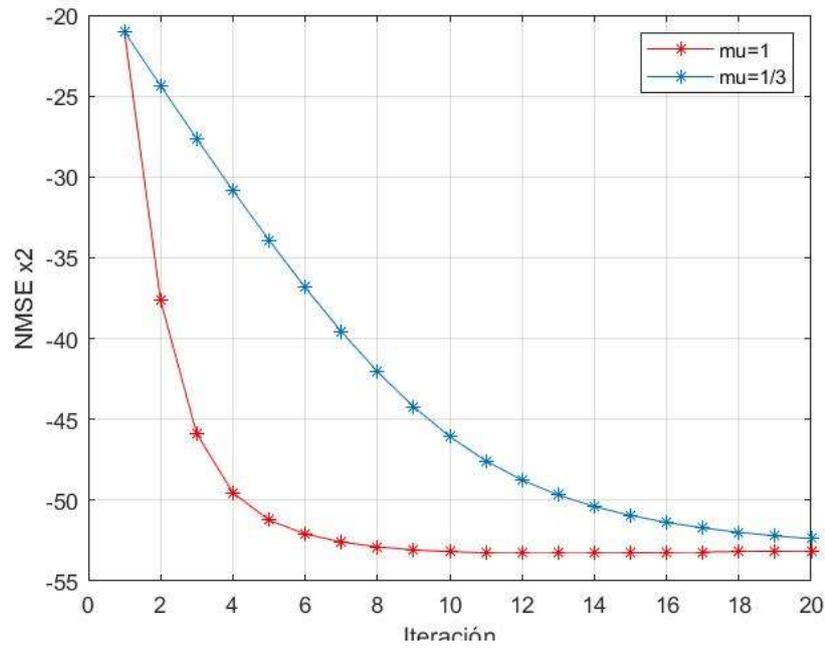


Figura 5.2: NMSE de x_2 para distintas velocidades de aprendizaje.

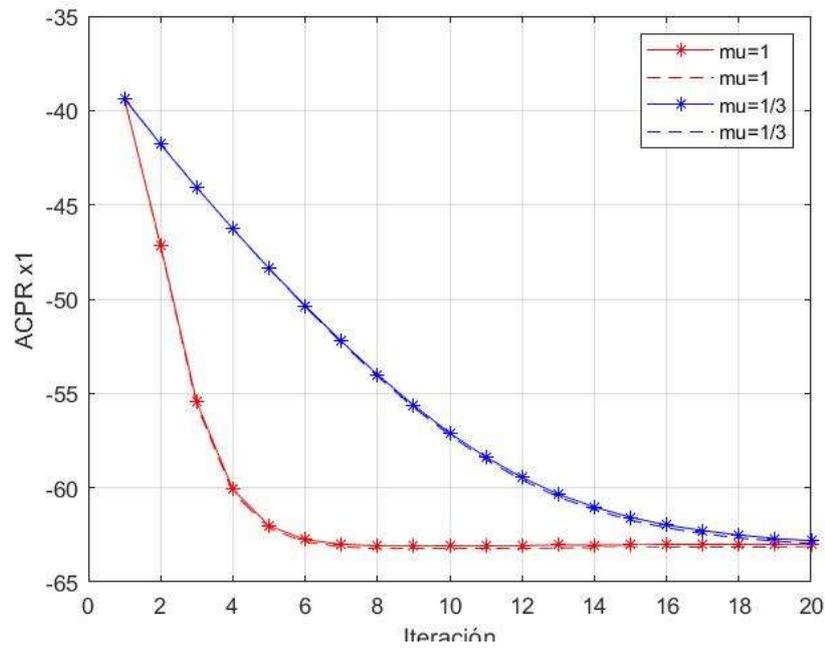


Figura 5.3: ACPR de x_1 para distintas velocidades de aprendizaje.

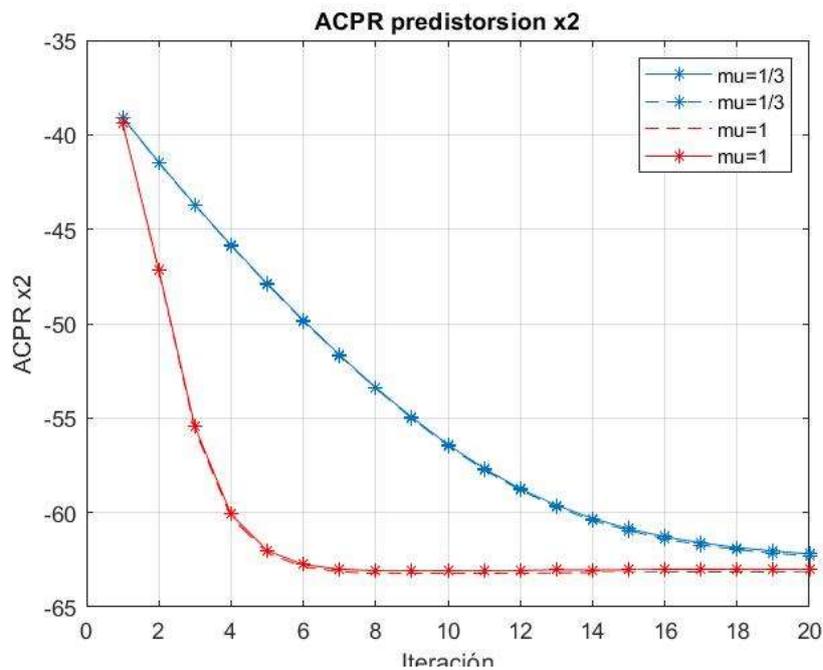


Figura 5.4: ACPR de x_2 para distintas velocidades de aprendizaje.

Cabe destacar de estos resultados que tanto el NMSE como el ACPR son iguales para las dos señales de entrada x_1 y x_2 , y que en el ACPR los resultados a la derecha de la señal, que se muestran en línea continua, y a la izquierda de la señal, que se muestran en línea discontinua, son idénticos. Esto se debe a que el amplificador se ha programado y por tanto su comportamiento es ideal.

Los datos obtenidos, para ambas velocidades de aprendizaje, son buenos, con un NMSE de -53 dB y un ACPR de -62 dB.

Además, para observar cómo se va linealizando la señal de salida a cada iteración, se muestran a continuación las gráficas de la característica AM/PM y de la ganancia de la primera y la última iteración.

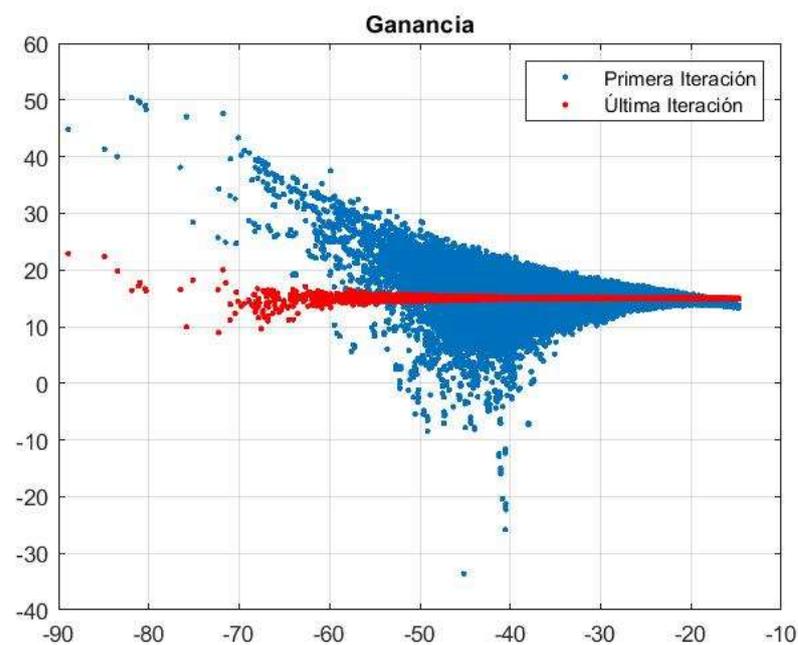


Figura 5.5: Ganancia en la primera y última iteración.

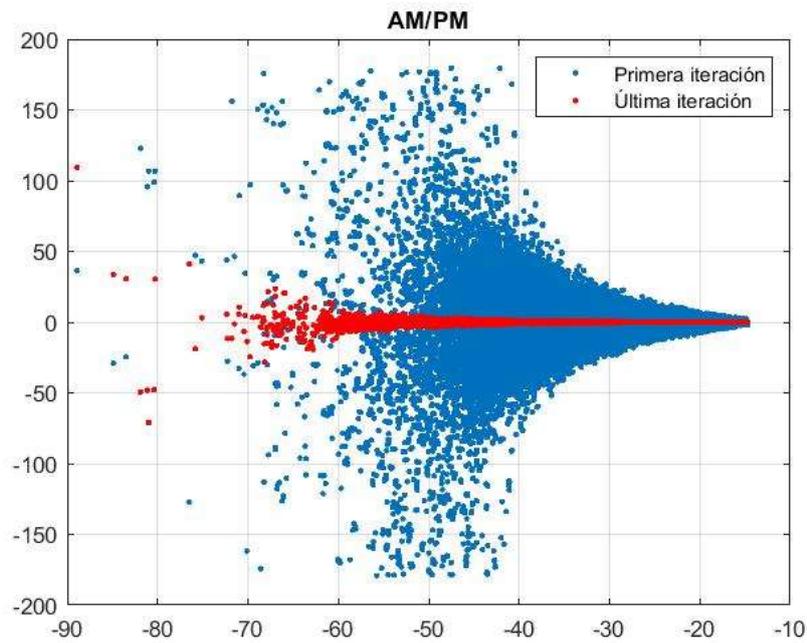


Figura 5.6: Característica AM/PM en la primera y última iteración.

Se puede observar claramente en ambas gráficas que el comportamiento a la salida del amplificador se linealiza, ya que en la primera iteración los resultados son no lineales, y en la última tanto la ganancia como la característica AM/PM se asemejan bastante a una línea recta.

Con esto se cierra el capítulo de resultados del amplificador sintético, ya que se han obtenido una buena cantidad de resultados que demuestran que lo implementado teóricamente funciona. El siguiente capítulo se centrará en el montaje de laboratorio y su comportamiento.

6 RESULTADOS DE LABORATORIO

Para realizar el set-up de este proyecto se ha imitado el del artículo de Shoaib Amin explicado en el Capítulo 4 de este trabajo. Para el montaje, se necesitarán dos amplificadores, cuatro acopladores, un generador de señal (VSG) con dos puertos de salida independientes, dos preamplificadores, un RF switch, un analizador vectorial de señal (VSA) y todos los cables coaxiales que los unen. Las hojas de especificaciones de los amplificadores y acopladores se muestran en el Anexo A.

En la siguiente imagen se puede observar dicho montaje, que se ha realizado en el laboratorio de Teoría de la Señal y Comunicaciones, situado en la primera planta del bloque L-3 de la Escuela Técnica Superior de Ingeniería (ETSI).

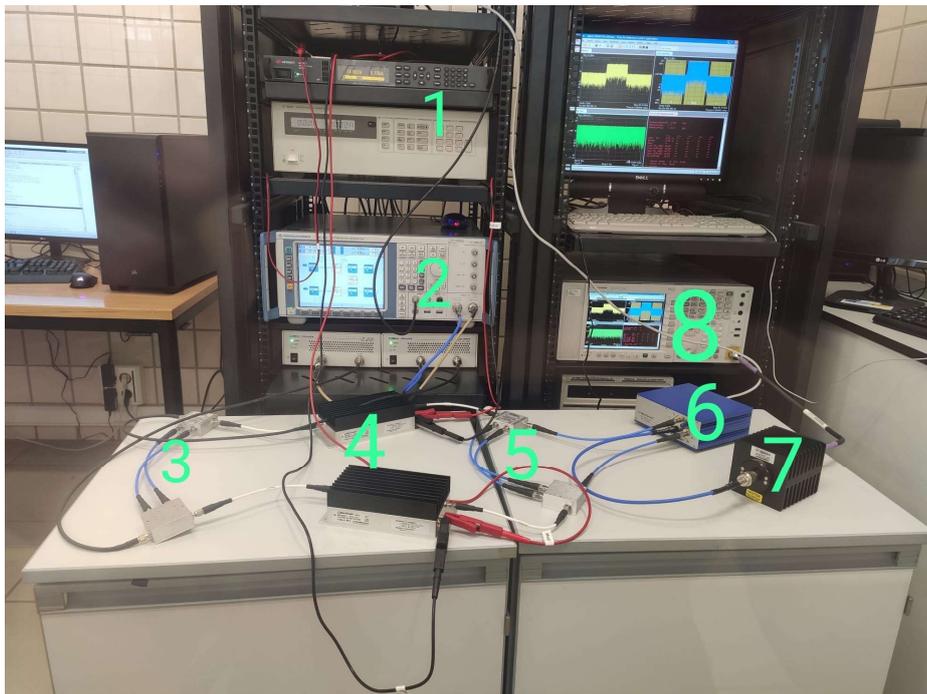


Figura 6.1: Set-up de laboratorio.

En la imagen se han numerado los distintos dispositivos utilizados. El número 1 corresponde a las dos fuentes de alimentación, el número 2 al generador de señal, el número 3 y el número 5 son los cuatro acopladores, el número 4 los dos amplificadores de potencia, el número 6 el RF switch, el número 7 un atenuador para que la cantidad de señal que entre en el receptor no sea demasiado grande, y el número 8 el receptor.

Una de las diferencias entre los datos que se obtienen en laboratorio en comparación con el amplificador sintético es que en este caso no se puede diferenciar entre cuatro casos para distintos tipos de acoplamiento, sino que el acoplamiento es fijo y no se conoce a priori. Por tanto, todos los datos que se muestran en este apartado son para el caso más genérico, el de acoplamiento lineal y no lineal, ya que no se puede cancelar el efecto de ninguno de los acopladores.

Al igual que en el caso del amplificador sintético, se mostrarán separados los datos obtenidos para modelado y para predistorsión digital.

6.1. Modelado

En lo que al modelado se refiere, se probó el set-up con los cuatro modelos y para dos niveles de potencia promedio distintos: para una potencia en el generador de -35 dBm y para una potencia de -37 dBm. Los datos obtenidos se muestran a continuación, donde el primer dato mostrado es el NMSE para la salida y_1 y el

segundo el NMSE para la salida y_2 , ambos en dB.

	IDENTIFICACIÓN		VALIDACIÓN	
	$P_{\text{gen}} = -35 \text{ dBm}$	$P_{\text{gen}} = -37 \text{ dBm}$	$P_{\text{gen}} = -35 \text{ dBm}$	$P_{\text{gen}} = -37 \text{ dBm}$
GMPLC	-38.01 / -37.76	-39.38 / -37.98	-28.77 / -37.51	-37.55 / -37.59
PH 2x2	-37.06 / -37.56	-39.04 / -37.89	-28.60 / -37.32	-37.28 / -37.58
GMPNLC	-36.61 / -36.26	-39.30 / -38.00	-28.12 / -35.18	-38.55 / -37.62
EGMPNLC	-36.98 / -37.43	-38.96 / -37.49	-28.06 / -36.92	-37.00 / -37.04

Tabla 6.1: NMSE, en dB, para niveles de potencia en el generador de -35 y -37 dBm.

Como es lógico, los datos obtenidos para un nivel de potencia de -35 dBm son peores que los obtenidos para un nivel de potencia de -37 dBm para todos los modelos. Esto se debe a que aplicar un nivel de potencia mayor a la entrada del amplificador supone hacerlo operar en condiciones de mayor no linealidad, y por tanto la señal a su salida está más fuertemente distorsionada y es más difícil de modelar.

Por ello, en los siguientes apartados solo se utiliza un nivel de potencia de -37 dBm, para obtener los mejores resultados posibles con los distintos modelos.

También cabe destacar que los resultados obtenidos en validación, para la mayoría de los modelos, son mejores para la salida y_1 que para la salida y_2 . No hay una razón clara de por qué esto ocurre, sino que forma parte de la propia aleatoriedad del comportamiento de los elementos del set-up, principalmente el amplificador, y muestra la dificultad de linealizarlo y modelarlo.

Por otro lado, se puede observar que en este caso, y a diferencia de los resultados obtenidos en el Capítulo 5 de este trabajo, el modelo que mejores resultados aporta en validación es el GMPNLC, y el que peores el EGMPNLC, pero con poca diferencia entre el mejor caso y el peor, por lo que se puede concluir que los cuatro modelos aportan una predicción del comportamiento parecida.

En el siguiente apartado, con técnicas de predistorsión digital, se va a intentar linealizar la salida mediante los cuatro modelos minimizando el vector error como se hizo con el amplificador sintético.

6.2. Predistorsión digital

En este apartado del trabajo se hicieron cuatro pruebas diferentes, de las cuales dos se presentarán en este capítulo y las otras dos se mostrarán en los Anexos C.1 y C.2.

La primera de ellas consiste en utilizar los modelos propuestos por S. Amin con una única mejora consistente en incluir también los términos con la envolvente adelantada con respecto a la señal (tercer y sexto sumatorios del modelo 2D-GMP), de la misma forma que se han utilizado para el amplificador sintético, con la misma configuración para los cuatro modelos, la que se muestra en la Tabla 5.6.

En cuanto a la segunda prueba, se añadieron a los cuatro modelos los siguientes términos:

$$x_1^*(n-l) \text{ y } x_2^*(n-l)$$

donde el asterisco * significa el término conjugado de ambas señales x_1 y x_2 .

En la tercera prueba se modificó la segunda añadiendo términos cruzados para los modelos no lineales, es decir, el GMPNLC y EGMPNLC, siendo estos:

$$x_1^*(n-l) \times |x_2(n-l)| \text{ y } x_2^*(n-l) \times |x_1(n-l)|$$

Sin embargo, para los modelos lineales, el GMPLC y el PH2x2, se añadieron los siguientes términos:

$$x_1^*(n-l) \times |x_1(n-l)| \text{ y } x_2^*(n-l) \times |x_2(n-l)|$$

Por último, en la cuarta prueba se mantuvieron los términos cruzados de la tercera, pero además se modificó la configuración de no linealidad y de memoria de los cuatro modelos a la que se muestra a continuación:

<i>Conjunto</i>	GMPLC	PH 2x2	GMPNLC	EGMPNLC
<i>Ka</i>	[0 1 2 3 4 5 6 7 8 9 10]	[0 1 2 3 4 5 6 7 8 9 10]	[0 1 2 3 4 5 6 7 8 9 10]	[]
<i>La</i>	[10 5 3 1 1 1 1 1 1 1 1]	[10 5 3 1 1 1 1 1 1 1 1]	[10 5 3 1 1 1 1 1 1 1 1]	[]
<i>Kb</i>	[1 2 3 4 5]	[1 2 3 4 5]	[1 2 3 4 5]	[0 1 2 3 4 5 6 7 8 9 10]
<i>Lb</i>	[2 2 2 2 2]	[2 2 2 2 2]	[2 2 2 2 2]	[10 5 3 1 1 1 1 1 1 1 1]
<i>Mb</i>	[1 1 1 1 1 1]	[1 1 1 1 1 1]	[1 1 1 1 1 1]	[1 1 1 1 1 1 1 1 1 1 1 1]
<i>Kc</i>	[1 2 3 4 5]	[1 2 3 4 5]	[1 2 3 4 5]	[0 1 2 3 4 5 6 7 8 9 10]
<i>Lc</i>	[2 2 2 2 2]	[2 2 2 2 2]	[2 2 2 2 2]	[10 5 3 1 1 1 1 1 1 1 1]
<i>Mc</i>	[1 1 1 1 1 1]	[1 1 1 1 1 1]	[1 1 1 1 1 1 1 1]	[1 1 1 1 1 1 1 1 1 1 1 1]

Tabla 6.2: Configuración del modelo para la prueba 4 en predistorsión.

Las pruebas que se mostrarán en este capítulo son la primera y la última, quedando la segunda y tercera en los Anexos C.1 y C.2.

En cuanto a la primera prueba, se muestran a continuación los resultados obtenidos para los modelos EGMPNLC y GMPNLC, ya que son los que incluyen más no linealidad y son más adecuados para el comportamiento real esperado de los amplificadores.

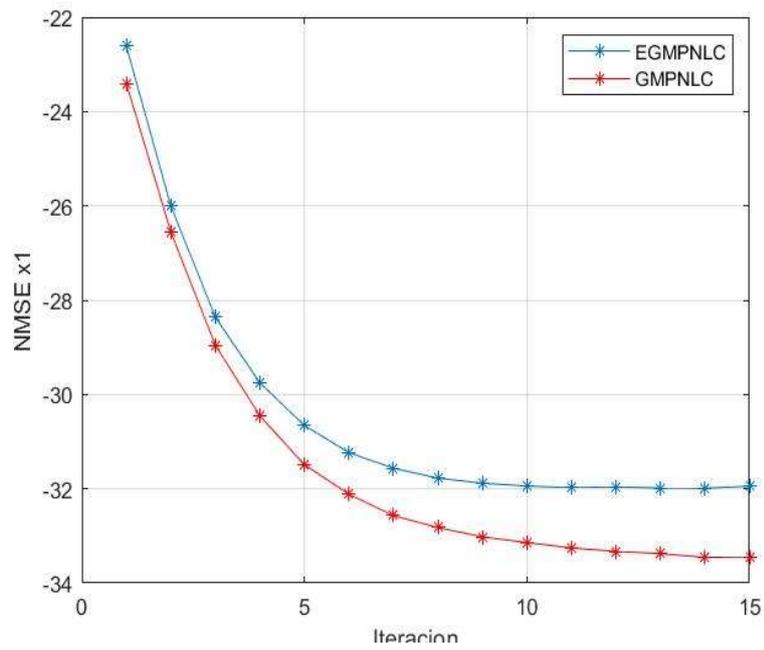


Figura 6.2: NMSE de la señal x_1 para los distintos modelos GMPNLC y EGMPNLC.

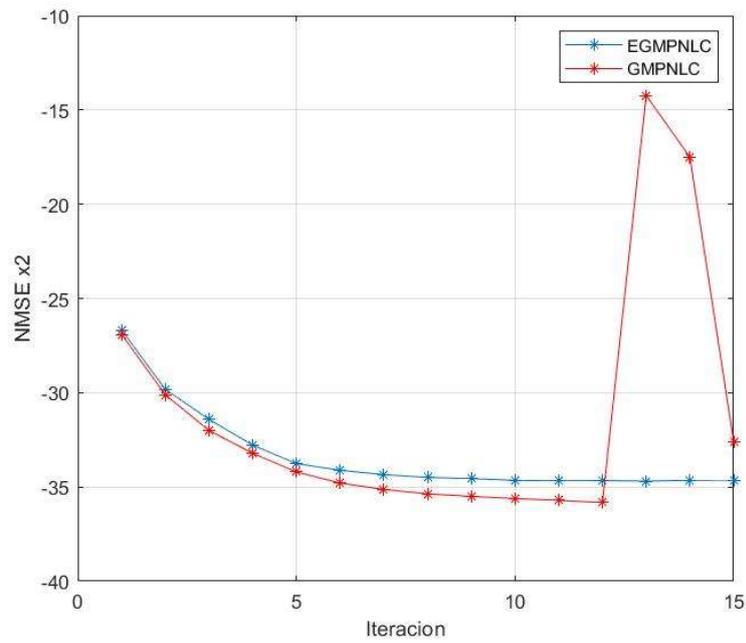


Figura 6.3: NMSE de la señal x_2 para los modelos GMPNLC y EGMPNLC

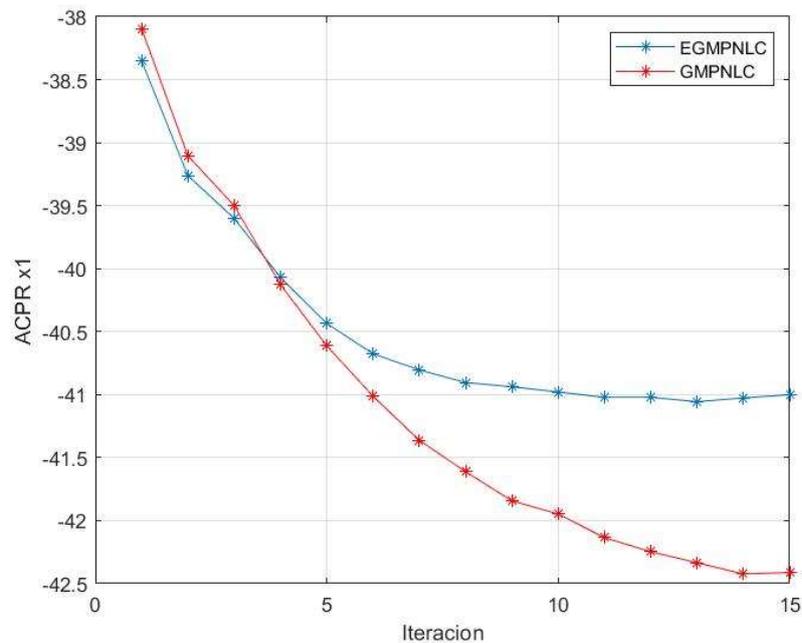


Figura 6.4: Peor valor de ACPR de la señal x_1 para los modelos GMPNLC y EGMPNLC

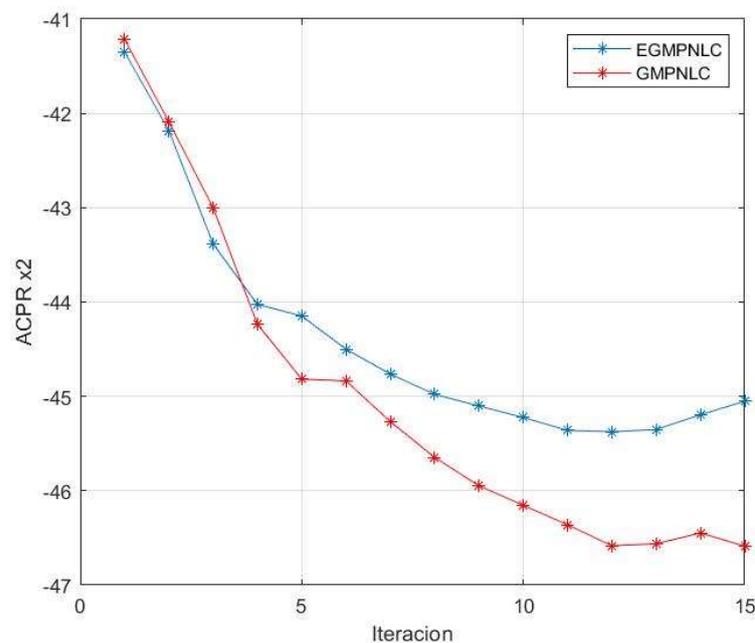


Figura 6.5: Peor valor de ACPR de la señal x_2 para los modelos GMPNLC y EGMPNLC.

Cabe destacar que en todos los casos el modelo GMPNLC ofrece mejores prestaciones que el modelo EGMPNLC, y que la señal x_2 tiene, tanto para los resultados de ACPR como para los de NMSE, un comportamiento más lineal que la señal x_1 .

También es importante señalar que en el NMSE de la señal x_2 , el modelo GMPNLC rebosa en la iteración 13 y empieza a obtener muy malas prestaciones, pero esto se va mejorando en las siguientes iteraciones hasta casi llegar a estabilizarse. Esto puede deberse a algún problema de tipo numérico en la regresión que el método DOMP realiza para que afecta principalmente en banda, ya que se puede observar también que en el ACPR de esta misma señal no ocurre, más bien al contrario, se ofrece una ACPR de -46.5 dB, que es un dato bastante

bueno.

Los datos obtenidos son buenos pero mejorables, ya que por ejemplo la señal x_1 no llega en ningún momento a un ACPR de -45 dB, que es el mínimo que exigen los estándares de comunicaciones móviles para la señal a transmitir.

Por ello, en las siguientes pruebas se intenta mejorar estos datos, y además se prueba con los cuatro modelos. Los resultados de ACPR y NMSE de la última prueba se muestran a continuación, donde además hay una iteración 16 señalada con un círculo, que muestra los resultados de validación de cada modelo.

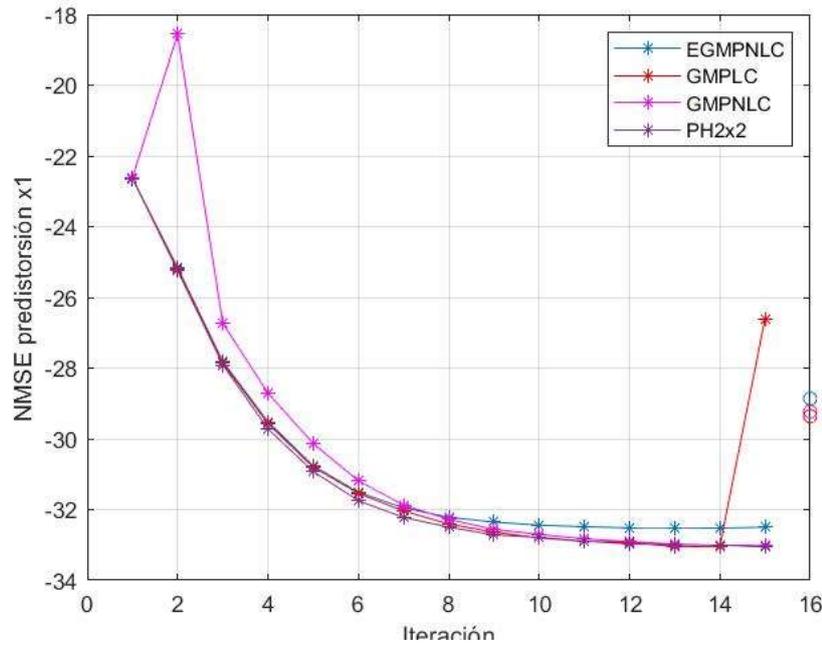


Figura 6.6: NMSE de la señal x_1 .

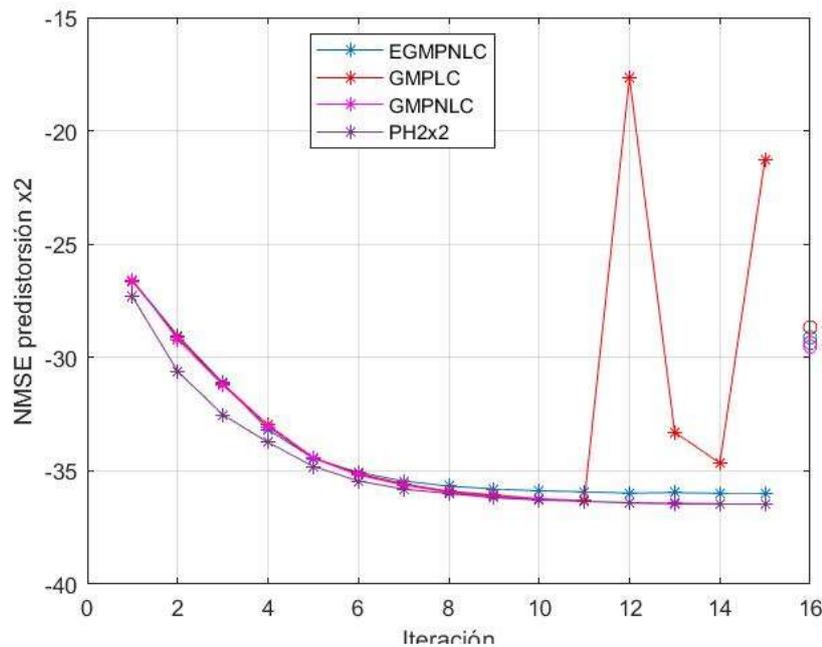
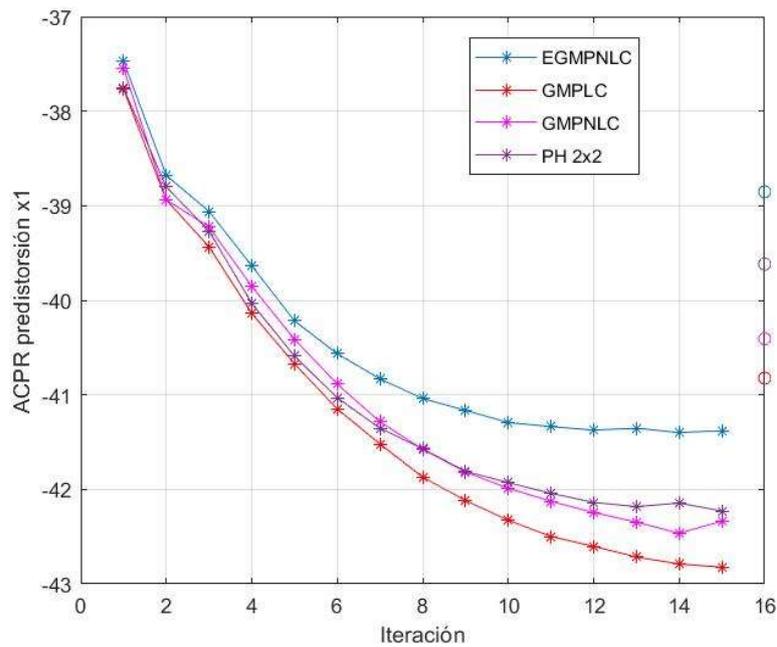
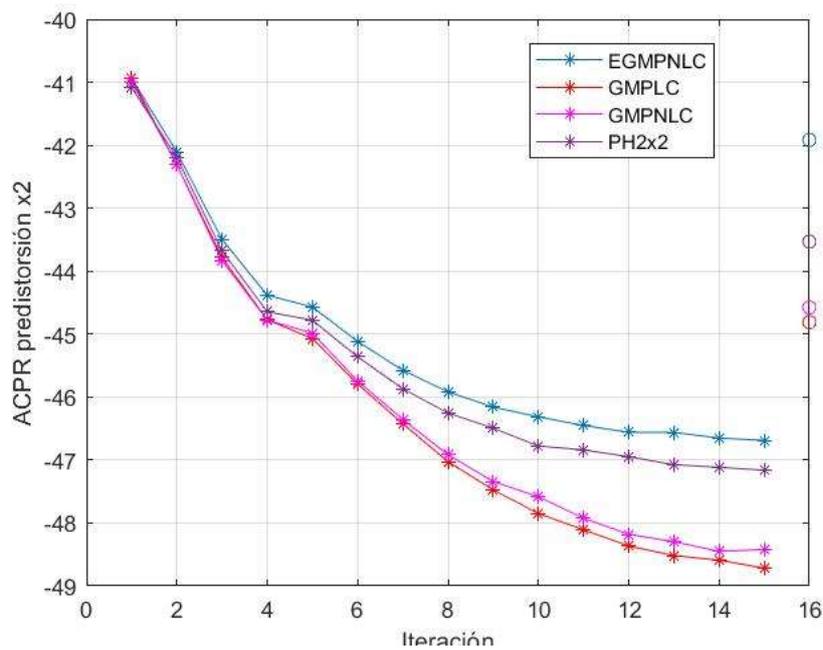


Figura 6.7: NMSE de la señal x_2 .

Figura 6.8: ACPR de la señal x_1 .Figura 6.9: ACPR de la señal x_2 .

Los resultados obtenidos para esta prueba no ofrecen grandes mejoras respecto a la primera, por lo que se concluyó que este era el límite que el modelo ofrecía.

Además, para esta configuración también se ha salvado el espectro y la constelación de la primera iteración y de la última, para ver si se observan mejoras entre ambas. Como el caso de los cuatro modelos es similar, se va a tomar como ejemplo el modelo PH 2x2, cuyos resultados se muestran a continuación.

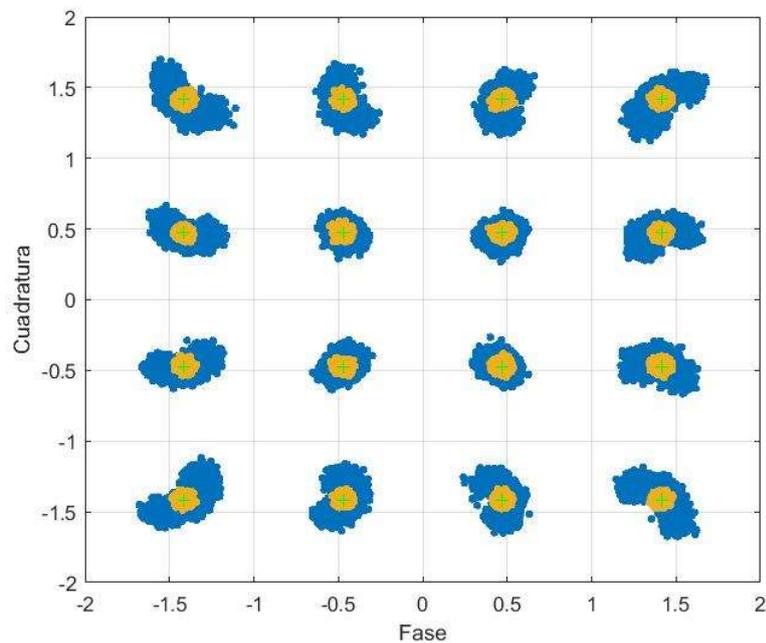


Figura 6.10: Constelación.

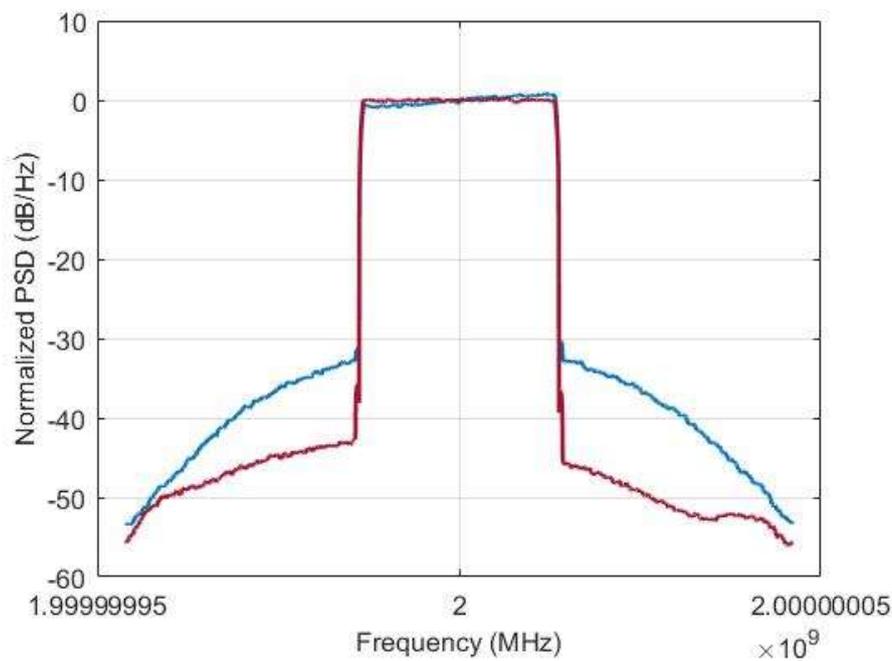


Figura 6.11: Espectro.

En cuanto a la constelación, se puede observar que en la primera iteración, dibujada en azul, los símbolos son más similares a una gota de agua, y en la última esta se reduce y se asimila más a un círculo en cuyo centro se encuentra el símbolo transmitido.

El espectro también mejora significativamente entre la primera iteración y la última. Esto se observa en la reducción del recrecimiento espectral, que es mucho mayor en la primera iteración (en azul), y más plano en la última (en rojo).

Por tanto, el DPD DLA funciona correctamente ya que linealiza el modelo, sin embargo, no se observan mejoras entre las distintas pruebas que se hicieron.

Por último, y para cerrar el capítulo sobre los resultados de laboratorio, resulta interesante mostrar los regresores más importantes de cada modelo, es decir, los términos que mejores prestaciones aportan, y gracias a los que se consigue un NMSE y ACPR mejor. Para ello, hay que fijarse en los primeros regresores que elige el método DOMP, ya que como se explicó anteriormente éste solo utiliza los regresores que más información dan sobre el modelo. Para hacer una buena muestra de los mismos, se van a mostrar los primeros veinte regresores de cada modelo.

MODELO REGRESORES

<i>GMPLC</i>	<p>'x1(n-0)' 'x1(n-1)' 'x1(n-2)' 'x1(n-3)' 'x1(n-4)' 'x1(n-5)' 'x1(n-6)' 'x1(n-7)' 'x1(n-8)' 'x1(n-9)' 'x1(n-10)' 'x1(n-0)*x1(n-0) ^{1}' 'x1(n-1)*x1(n-1) ^{1}' 'x1(n-2)*x1(n-2) ^{1}' 'x1(n-3)*x1(n-3) ^{1}' 'x1(n-4)*x1(n-4) ^{1}' 'x1(n-5)*x1(n-5) ^{1}' 'x1(n-0)*x1(n-0) ^{2}' 'x1(n-1)*x1(n-1) ^{2}' 'x1(n-2)*x1(n-2) ^{2}'</p>
<i>PH 2x2</i>	<p>'x1(n-0)' 'x1(n-1)' 'x1(n-2)' 'x1(n-3)' 'x1(n-4)' 'x1(n-5)' 'x1(n-6)' 'x1(n-7)' 'x1(n-8)' 'x1(n-9)' 'x1(n-10)'</p>

	$'x1(n-0)*x2(n-0)^{\{1\}}'$ $'x1(n-1)*x2(n-1)^{\{1\}}'$ $'x1(n-2)*x2(n-2)^{\{1\}}'$ $'x1(n-3)*x2(n-3)^{\{1\}}'$ $'x1(n-4)*x2(n-4)^{\{1\}}'$ $'x1(n-5)*x2(n-5)^{\{1\}}'$ $'x1(n-6)*x2(n-6)^{\{1\}}'$ $'x1(n-7)*x2(n-7)^{\{1\}}'$ $'x1(n-8)*x2(n-8)^{\{1\}}'$
<i>GMPNLC</i>	$'x1(n-0)'$ $'x1(n-1)'$ $'x1(n-2)'$ $'x1(n-3)*'$ $'x1(n-4)'$ $'x1(n-5)'$ $'x1(n-6)'$ $'x1(n-7)'$ $'x1(n-8)'$ $'x1(n-9)'$ $'x1(n-10)'$ $'x1(n-0)*x2(n-0)^{\{1\}}'$ $'x1(n-1)*x2(n-1)^{\{1\}}'$ $'x1(n-2)*x2(n-2)^{\{1\}}'$ $'x1(n-3)*x2(n-3)^{\{1\}}'$ $'x1(n-4)*x2(n-4)^{\{1\}}'$ $'x1(n-5)*x2(n-5)^{\{1\}}'$ $'x1(n-6)*x2(n-6)^{\{1\}}'$ $'x1(n-7)*x2(n-7)^{\{1\}}'$ $'x1(n-8)*x2(n-8)^{\{1\}}'$
<i>EGMPNLC</i>	$'conj_x1(n-0)'$ $'conj_x1(n-1)'$ $'conj_x1(n-2)'$ $'conj_x1(n-0)*x1(n-0)^{\{1\}}*x2(n-0)^{\{1\}}'$ $'conj_x1(n-1)*x1(n-1)^{\{1\}}*x2(n-1)^{\{1\}}'$ $'conj_x1(n-2)*x1(n-2)^{\{1\}}*x2(n-2)^{\{1\}}'$ $'conj_x2(n-0)*x1(n-0)^{\{1\}}*x2(n-0)^{\{1\}}'$ $'conj_x2(n-1)*x1(n-1)^{\{1\}}*x2(n-1)^{\{1\}}'$

'conj_x2(n-2)* x1(n-2) ^{1}* x2(n-2) ^{1}'
'x1(n-0)'
'x1(n-0)'
'x2(n-0)'
'x2(n-0)'
'x1(n-1)'
'x1(n-1)'
'x2(n-1)'
'x2(n-1)'
'x1(n-2)'
'x1(n-2)'
'x2(n-2)'

Tabla 6.3: Tabla de regresores más importantes para los cuatro modelos.

Se puede observar que el único modelo que elige unos coeficientes distintos es el EGMPNLC. Puede ser este el motivo que haga que sea el modelo que da unas prestaciones diferentes, yendo los otros tres modelos bastante en sintonía, como se puede observar en las figuras 6.6, 6.7, 6.8 y 6.9.

7 CONCLUSIONES Y LÍNEAS FUTURAS

Intentar modelar el comportamiento de un PA es siempre una tarea difícil debido a su no linealidad, pero esta dificultad aumenta cuando la señal de entrada en el amplificador es una señal MIMO. Este es el caso que centra este trabajo, con una señal MIMO 2×2 .

Sin embargo, a pesar del problema que supone trabajar con este tipo de señales, son muy necesarias ya que mejoran la capacidad del sistema sin necesidad de modificar el ancho de banda o la relación señal a ruido, recursos bastante limitados a día de hoy.

Durante este trabajo, se ha demostrado que es necesaria la utilización de un DPD para compensar los efectos no lineales del PA. De esta forma, se obtienen buenos resultados, aunque se observa también que no hay grandes mejoras entre utilizar los modelos tal como se muestra en el artículo de S. Amin o modificarlos, añadiéndole nuevos términos que ayuden a explicar el comportamiento no lineal del PA. Por ello, pretendo, después de finalizar este trabajo, crear nuevos modelos para intentar mejorar el NMSE y ACPR obtenidos. Para la creación de estos modelos me fijaré en los regresores más importantes que el método DOMP elige para cada modelo, e intentaré buscar una relación algebraica entre ellos.

Otra conclusión de este trabajo es que al trabajar con señales MIMO, la cantidad de datos a analizar en el método de mínimos cuadrados se vuelve computacionalmente imposible, ya que el ordenador gasta mucha memoria en realizar la pseudoinversa de la matriz de medida de la señal. Por ello, para trabajar con sistemas de este tipo se hace necesaria la creación de nuevos algoritmos que eliminen la información redundante o innecesaria de la señal, como es el algoritmo DOMP. A la vista de los resultados de las secciones 5.1.1. y 5.1.2., se puede concluir que el método DOMP es un método válido para la búsqueda de soluciones, ya que sus resultados son muy parecidos a los del método LS.

ANEXO A: DATASHEET

En este primer anexo se muestra la hoja de especificaciones de los dos amplificadores modelo ZHL-42 y los cuatro acopladores modelo ZFBDC16-63HP+ utilizados en el set-up de laboratorio de este proyecto.

Anexo A.1. Datasheet de amplificador ZHL-42

Coaxial
Amplifier

50Ω Medium High Power 700 to 4200 MHz

ZHL-42



Features

- wideband, 700 to 4200 MHz
- high IP3, +38 dBm typ
- high gain, 30 dB min.
- medium high power, 28 dBm min.

Applications

- communication systems
- instrumentation
- cellular
- laboratory

CASE STYLE: U36

Connectors	Model	Price	Qty.
SMA	ZHL-42	\$895.00 ea.	(1-9)

Electrical Specifications

MODEL NO.	FREQ. (MHz)	GAIN (dB)		MAXIMUM POWER OUTPUT (dBm)		DYNAMIC RANGE		VSWR (-1)		DC POWER	
		Min.	Max.	(1 dB Compr.)	Input (no damage)	NF (dB)	IP3 (dBm)	In	Out	Volt (V) Nom.	Current (A) Max.
ZHL-42	f _c 700 4200	30	±1.0*	-28	+5	10.0	+38	2.5	2.5	15	0.88

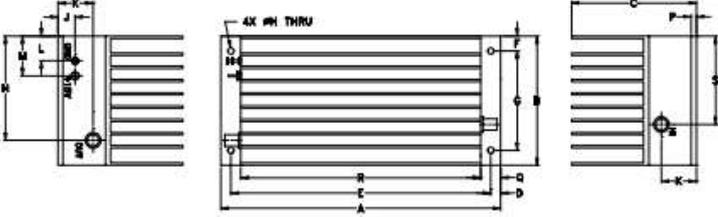
* Measured at 25°C
Open load is not recommended, potentially can cause damage.
With no load derate max input power by 20 dB

Maximum Ratings

Operating Temperature	-20°C to 65°C
Storage Temperature	-55°C to 100°C
DC Voltage	+20V Max.

Permanent damage may occur if any of these limits are exceeded.

Outline Drawing



Outline Dimensions (inches)

A	B	C	D	E	F	G	H	J	K	L	M	N	P	Q	R	S	wt
7.00	3.25	2.13	.25	6.500	.38	2.500	.156	.73	.88	.83	1.13	2.23	.125	.50	8.00	2.23	grams
177.80	82.55	54.10	6.35	165.10	9.65	63.50	3.96	18.54	22.35	16.00	28.70	56.64	3.18	12.70	152.40	66.64	900

Notes

A. Performance and quality attributes and conditions not expressly stated in this specification document are intended to be excluded and do not form a part of this specification document.

B. Electrical specifications and performance data contained in this specification document are based on Mini-Circuit's applicable established test performance criteria and measurement instructions.

C. The parts covered by this specification document are subject to Mini-Circuit's standard limited warranty and terms and conditions (collectively, "Standard Terms"). Purchasers of this part are entitled to the rights and benefits contained therein. For a full statement of the Standard Terms and the exclusive rights and remedies thereunder, please visit Mini-Circuit's website at www.minicircuits.com/MCISalesTerms.jsp



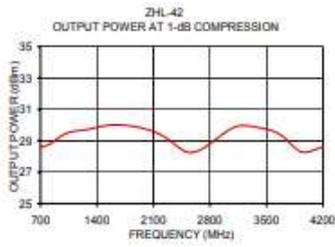
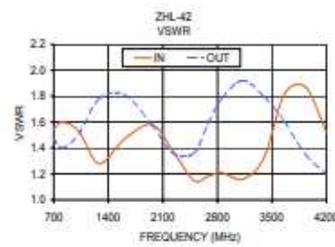
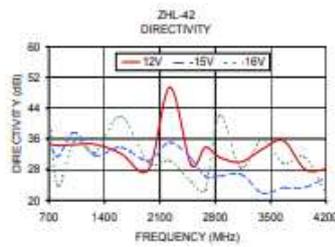
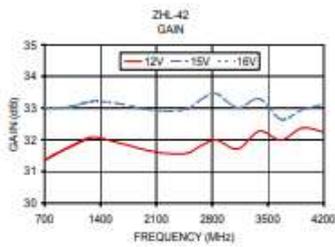
www.minicircuits.com P.O. Box 35180, Brooklyn, NY 11235-0033 (718) 934-4500 sales@minicircuits.com

REV. 01
MSL 142
ZHL-42
131009
Page 1 of 2

Typical Performance Data/Curves

ZHL-42

FREQUENCY (MHz)	GAIN (dB)			DIRECTIVITY (dB)			VSWR (:1)		NOISE FIGURE (dB)	POUT at 1 dB COMPR. (dBm)
	12V	15V	16V	12V	15V	16V	IN	OUT		
700.00	31.34	33.00	32.99	34.70	35.90	41.90	1.55	1.47	8.63	28.60
814.00	31.52	33.02	33.00	34.30	31.40	23.70	1.60	1.40	8.12	28.78
1020.80	31.78	33.03	33.04	34.50	37.70	35.40	1.53	1.51	7.74	29.48
1280.10	32.07	33.21	33.20	34.60	31.70	32.40	1.28	1.78	7.54	29.71
1605.20	31.92	33.15	33.14	32.20	33.80	41.80	1.47	1.81	7.48	30.02
1956.40	31.68	32.97	32.96	28.00	30.30	29.70	1.58	1.58	7.67	29.81
2225.60	31.57	32.90	32.89	40.40	35.10	30.30	1.38	1.38	7.58	29.28
2494.00	31.58	32.98	32.99	29.30	30.80	24.80	1.15	1.36	7.55	28.32
2674.40	31.83	33.29	33.29	33.80	26.30	22.80	1.18	1.58	7.54	28.40
2853.80	31.99	33.45	33.46	31.20	26.30	42.00	1.21	1.78	7.55	29.02
3123.10	31.71	33.00	33.01	30.60	26.80	26.50	1.16	1.92	7.59	29.90
3392.30	32.27	33.29	33.30	33.30	21.90	35.80	1.32	1.80	7.58	29.85
3661.50	31.99	32.65	32.64	35.50	23.30	29.60	1.82	1.60	7.41	29.43
3930.80	32.37	32.95	32.93	28.10	23.40	31.20	1.87	1.35	7.53	28.30
4200.00	32.24	33.12	33.13	28.10	25.90	21.50	1.52	1.20	7.52	28.61



Notes
A. Performance and quality attributes and conditions not expressly stated in this specification document are intended to be excluded and do not form a part of this specification document.
B. Electrical specifications and performance data contained in this specification document are based on Mini-Circuits' applicable established test performance criteria and measurement instructions.
C. The parts covered by this specification document are subject to Mini-Circuits' standard limited warranty and terms and conditions (collectively, "Standard Terms"). Purchasers of this part are entitled to the rights and benefits contained therein. For a full statement of the Standard Terms and the exclusive rights and remedies thereunder, please visit Mini-Circuits' website at www.minicircuits.com/MCStore/terms.jsp



Anexo A.2. Datasheet del acoplador ZFBDC16-63HP+

Coaxial Bi-Directional Coupler ZFBDC16-63HP+

50Ω Up to 75W 700 to 6000 MHz

The Big Deal

- Wideband, 700 to 6000 MHz
- High power handling, up to 75W
- Low mainline loss, 0.55 dB
- High directivity, 25 dB
- Excellent return loss, 20 dB typ.



CASE STYLE: JD1252

Product Overview

Mini-Circuits' ZFBDC16-63HP+ is a coaxial high-power, wideband bi-directional coupler supporting applications from 700 to 6000 MHz. This model is capable of handling up to 75W RF input power and provides high directivity, low mainline loss, and excellent return loss. The coupler comes housed in a rugged aluminum alloy case (2.00 x 2.00 x 0.88") with your choice of SMA or N-Type connectors.

Key Features

Feature	Advantages
Wideband, 700 to 6000 MHz	One device supports a broad range of applications.
Very high input power handling: • 75W, 700 to 2700 MHz • 50W, 2700 to 3500 MHz • 40W, 3500 to 6000 MHz	Produces nearly equal output signals, ideal for parallel path / multichannel systems.
Low mainline loss, 0.55 dB typ.	Provides excellent through-path signal power transmission.
Good directivity, 25 dB	High directivity allows accurate signal sampling through the coupled port with minimal measurement error.
Good return loss, 20 dB typ.	Well-matched for 50Ω systems with minimal signal reflection.

Notes

- A. Performance and quality attributes and conditions not expressly stated in this specification document are intended to be excluded and do not form a part of this specification document.
 B. Electrical specifications and performance data contained in this specification document are based on Mini-Circuits' applicable standardized test performance criteria and measurement instructions.
 C. The parts covered by this specification document are subject to Mini-Circuits' standard limited warranty and terms and conditions (collectively, "Standard Terms"). Purchasers of this part are entitled to the rights and benefits contained therein. For a full statement of the Standard Terms and the exclusive rights and remedies thereunder, please visit Mini-Circuits' website at www.minicircuits.com/MC/Store/terms.jsp



Coaxial Bi-Directional Coupler

50Ω Up to 75W 700 to 6000 MHz

ZFBDC16-63HP+



Generic photo used for illustrative purposes only
SMA version shown
CASE STYLE: JD1252

Connectors	Model
R-Type	ZFBDC16-63HP-16+
SMA	ZFBDC16-63HP-5+

+RoHS Compliant!
The + suffix identifies RoHS Compliance. See our web site for RoHS Compliance methodologies and qualifications.

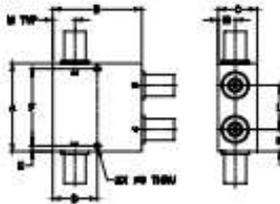
Maximum Ratings

Operating Temperature -55°C to 100°C
Storage Temperature -55°C to 100°C
Performance damage may occur if any of these limits are exceeded.

Coaxial Connections

INPUT	1
OUTPUT	2
COUPLED (forward)	4
COUPLED (reverse)	3

Outline Drawing



Outline Dimensions (inches)

A	B	C	D	E	F	G
0.30	0.00	0.86	1.000	0.73	1.762	0.129
00.81	00.00	21.50	25.40	1.83	44.45	3.26
H	J	K	L	M	N	Q
0.39	-	0.80	1.00	0.50	0.50	0.00
9.91	-	20.32	25.40	12.70	12.70	0.00

Features

- excellent mainline loss, 0.55 dB typ.
- excellent directivity, 25 dB typ.
- high power, up to 75W
- rugged shielded case

Applications

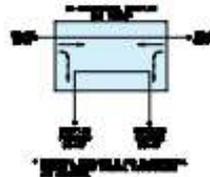
- power leveling & monitoring
- military mobile
- cellular
- W/Max
- PCN
- GSM
- WiFi

Electrical Specifications at 25°C

Parameter	Condition (MHz)	Min.	Typ.	Max.	Unit
Frequency Range	700	700		6000	
Mainline Loss	700 - 2000	—	0.4	0.7	dB
	2000 - 6000	—	0.6	0.9	
Coupling	700 - 1000	—	20.0±1.8	—	
	1000 - 1700	—	20.0±1.4	—	
	1700 - 2000	—	20.0±1.1	—	
	2000 - 2700	—	19.0±1.4	—	
	2700 - 3000	—	17.5±1.1	—	
	3000 - 6000	—	15.5±0.8	—	
Coupling Flatness(±)	1700 - 2000	—	0.6	0.9	dB
	2700 - 3000	—	0.6	1.0	
	3000 - 6000	—	0.5	0.9	
Directivity	700 - 2000	30	25	—	
	2000 - 3000	18	23	—	
	3000 - 4000	15	22	—	
	4000 - 6000	11	18	—	
Return Loss (Input)	700 - 3000	18	20	—	dB
	3000 - 6000	16	20	—	
Return Loss (Output)	700 - 3000	18	20	—	dB
	3000 - 6000	16	20	—	
Return Loss (Coupling)	700 - 3000	17	23	—	dB
	3000 - 6000	15	18	—	
Input Power	700 - 2700	—	—	75	W
	2700 - 3500	—	—	50	
	3500 - 6000	—	—	40	

1. Mainline loss includes theoretical power loss at coupled.

Electrical Schematic



Note

- Performance and quality attributes and conditions not expressly stated in this specification document are intended to be excluded and do not form a part of the specification document.
- Electrical specifications and performance data contained in this specification document are based on Mini-Circuits' applicable established test performance criteria and measurement instructions.
- The parts covered by this specification document are subject to Mini-Circuits' standard limited warranty and terms and conditions collectively, "Standard Terms". Purchasers of this part are entitled to the rights and benefits contained therein. For a full statement of the Standard Terms and the exclusive rights and remedies thereunder, please visit Mini-Circuits' website at www.minicircuits.com/MCUL_Default.asp.



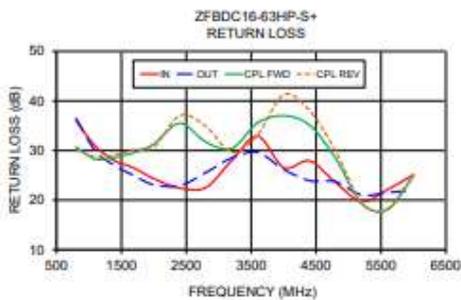
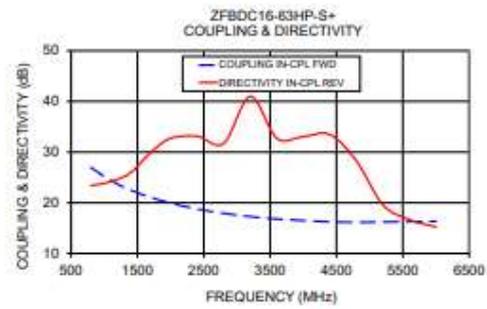
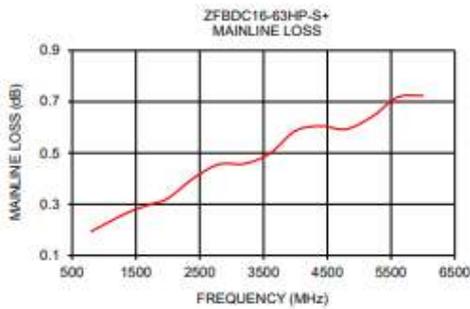
www.minicircuits.com P.O. Box 350100, Brooklyn, NY 11235-0000 (718) 334-4500 sales@minicircuits.com

REV. A
M15716E
30-100000-01
ZFBDC16-63HP+
W/CP/MS
1/10/08
Page 2 of 2

ZFBDC16-63HP+

Typical Performance Data

Frequency (MHz)	Mainline Loss (dB)		Coupling (dB)		Directivity (dB)		Return Loss (dB)			
	In-Out	In-Cpl Fwd	Out-Cpl Rev	Out-Cpl Fwd	In-Cpl Rev	In	Out	Cpl Fwd	Cpl Rev	
700.0	0.18	28.06	28.07	23.61	23.37	39.26	41.15	30.78	31.66	
800.0	0.19	26.96	26.97	23.89	23.41	35.99	36.49	30.71	30.35	
1100.0	0.23	24.37	24.39	24.76	24.14	30.87	29.89	28.24	28.67	
1400.0	0.27	22.50	22.52	27.32	25.98	27.91	26.97	28.62	29.08	
1700.0	0.30	21.07	21.10	31.35	29.90	26.43	24.92	29.60	30.19	
2000.0	0.32	19.93	19.98	32.08	30.71	24.32	23.10	31.31	30.88	
2400.0	0.40	18.78	18.83	30.36	33.13	22.42	22.91	35.50	37.03	
2700.0	0.46	17.92	17.96	30.79	31.68	22.53	25.53	31.72	34.91	
3200.0	0.46	17.27	17.30	25.75	40.97	27.98	28.50	30.42	29.66	
3600.0	0.50	16.81	16.82	25.61	32.71	32.88	29.84	35.58	33.16	
4000.0	0.59	16.50	16.48	28.21	33.04	26.55	26.18	37.01	41.27	
4400.0	0.60	16.25	16.25	24.15	33.46	27.91	23.91	35.11	38.00	
4700.0	0.59	16.16	16.14	27.38	28.26	23.55	23.71	28.43	29.92	
5200.0	0.64	16.22	16.17	21.83	19.57	19.71	21.03	19.38	10.46	
5600.0	0.72	16.28	16.22	17.64	16.67	22.10	21.50	18.08	18.31	
6000.0	0.72	16.35	16.32	17.58	15.21	25.13	21.99	24.82	25.14	



Notes

- A. Performance and quality attributes and conditions not expressly stated in this specification document are intended to be excluded and do not form a part of this specification document.
- B. Electrical specifications and performance data contained in this specification document are based on Mini-Circuits' applicable established test performance criteria and measurement instructions.
- C. The parts covered by this specification document are subject to Mini-Circuits' standard limited warranty (and terms and conditions collectively, "Standard Terms"). Purchasers of this part are entitled to the rights and benefits contained therein. For a full statement of the Standard Terms and the exclusive rights and remedies thereunder, please visit Mini-Circuits' website at www.minicircuits.com/MC3Store/terms.jsp



ANEXO B: CÓDIGO DE MATLAB EMPLEADO EN EL TRABAJO

Anexo B.1. Código principal para modelado. Este código se utiliza para la generación de la señal 5G-NR, su paso por el amplificador sintético, para posteriormente pasar a su identificación y validación.

```
clear all, close all; clc;

% Load static configuration
configurationMIMO;

% Amplificador sintético
alpha = 15; gamma = [1,0.1,0.1]; SNR = 65;
RMSin = -24.5; fc = 2e9;
% a = 0.03162;
% b = 0.03162;
% d = 0.03162;
% g = 0.03162;
d=0;
g=0;
a=0;
b=0;
%% Generacion señal (5GNR 20MHz) en +40 y -40
Foff =0;
fsmcd = 1024*30e3*5; %153.6MHz
fsint = 80e6;

[xid1, xs1, xteor_cp1, dataxid1] = genera_5GNR_multicarrier_v4(50, 30e3 ,20e6, Foff,
fsmcd, 8, 1024);
[xid2, xs2, xteor_cp2, dataxid2] = genera_5GNR_multicarrier_v4(50, 30e3, 20e6, Foff,
fsmcd, 8, 2048);

%% Hard-clipping
% xid = xid - mean(xid);
% xid = xid / max(abs(xid));
%
% PAPRxid = 20*log10(max(abs(xid))/rms(xid));
% fprintf('PAPR antes de clipping: %4.2f dB\n', PAPRxid);
%
% PAPRd = 10;
% clip = 10^((PAPRd-PAPRxid)/20);
% fprintf('Muestras a las que se le aplica clipping: %d (%4.8f %% de %d)\n',
sum(abs(xid) > clip), sum(abs(xid) > clip)/length(xid),length(xid));
% xid(abs(xid) > clip) = clip*exp(1i*angle(xid(abs(xid) > clip)));
%
% PAPRxid = 20*log10(max(abs(xid))/rms(xid));
% fprintf('PAPR despu?s de clipping: %4.2f dB\n', PAPRxid);

xid1 = scale_dBm(xid1,RMSin);
%espectro(xid1-mean(xid1),fsmcd, fc);xlabel('Frequency (MHz)'); ylabel('PSD
(dB/Hz)'); title(sprintf('Señal de entrada al amplificador 1'));
xid2 = scale_dBm(xid2,RMSin);
%espectro(xid2-mean(xid2),fsmcd, fc);xlabel('Frequency (MHz)'); ylabel('PSD
(dB/Hz)'); title(sprintf('Señal de entrada al amplificador 2'));
```

```

%% Medida
x1 = xid1;
x2 = xid2;
y1_int = syntheticPA(x1 + a.*x2, alpha, gamma, SNR);
y2_int= syntheticPA(x2 + b.*x1, alpha, gamma, SNR);
y1 = y1_int+d.*y2_int;
y2 = y2_int+g.*y1_int;
yid1 = y1;
yid2 = y2;
% espectro(yid1-mean(yid1),fsmed,fc);xlabel('Frequency (MHz)'); ylabel('PSD
(dB/Hz)'); title(sprintf('Salida del amplificador 1'));
% espectro(yid2-mean(yid2),fsmed,fc);xlabel('Frequency (MHz)'); ylabel('PSD
(dB/Hz)'); title(sprintf('Salida del amplificador 2'));
Py1 = dBm(yid1);
Px1 = dBm(xid1);
N = length(xid1);

%% Modelado
%[xid1, xid2] = separate_bands(xid,fsmed,fsint, Foff);
%[yid1, yid2] = separate_bands(yid,fsmed,fsint, Foff);

model.indices1 = sel_indices(xid1,yid1,0.015);
model.indices2 = sel_indices(xid2,yid2,0.015);

%% Identificacion
fprintf('\n->Identificación del modelo:\n')
model = model_2Dgmp_domp_omp_v2(yid1,yid2, xid1,xid2, model);
fprintf('NMSE modelado: %4.2f dB / %4.2f dB\n', model.nmse1,model.nmse2);

model.indices1 = 1:length(xid1);
model.indices2 = 1:length(xid2);

%% Validacion
fprintf('\n->Validación del modelo:\n')
modelval = model_2Dgmp_domp_omp_v2(yid1,yid2, xid1,xid2, model);
fprintf('NMSE validacion: %4.2f dB / %4.2f dB\n', modelval.nmse1,modelval.nmse2);
[ACEPR1, ~, Pinband1]= ACEPR_5GNR(y1, modelval.ymod1, dataxid1, fc,400);
[ACEPR2, ~, Pinband2]= ACEPR_5GNR(y2, modelval.ymod2, dataxid2, fc,401);
fprintf('ACEPR: %4.2f dB / %4.2f dB\n', ACEPR1(1),ACEPR2(1));

```

Anexo B.2. Amplificador sintético. Este código se utiliza para la simulación del comportamiento de un amplificador.

```

function y = syntheticPA(x, alpha, gamma, SNR)
% function y = syntheticPA(x, alpha)
% Emulates the behavior of a power amplifier device in a real measurement
% setup for simulation purposes.
% Memory effects and AWGN with SNR = 65 dB are included.
% Inputs:
% x - input complex envelope signal
% alpha - parameter (0<=alpha<=1) controlling the degree of nonlinearity
% gamma - weights for the inclusion of memory effects.
% SNR - desired signal-to-noise ratio (dB).
% Output:
% y - output complex envelope signal

```

```

if nargin == 2,
    gamma0 = 1;    % Weights for the inclusion of memory effects.
    gamma1 = 0.1;
    gamma2 = 0.1;
    SNR = 65;
else
    gamma0 = gamma(1);    % Weights for the inclusion of memory effects.
    gamma1 = gamma(2);
    gamma2 = gamma(3);
    if nargin == 3,
        SNR = 65;
    end
end

% Useful auxiliary functions
dBm = @(x) 10*log10(rms(x).^2/100)+30;
scale_dBm = @(x,P) x*10^((P-dBm(x))/20);

Gampl = 15; % Considered average gain of the power amplifier (dB).

% Synthetic PA with memory effects
g = tanh(alpha*abs(x))./(alpha*abs(x));
y = gamma0*g.*x+gamma1*circshift(g.*x,1)+gamma2*(circshift(g.*x,2));

% AWGN inclusion for certain SNR value
n = randn(size(y))+1i*randn(size(y));
y = scale_dBm(y,dBm(x)+Gampl);
n = scale_dBm(n,dBm(y)-SNR);
y = y+n;

end

```

Anexo B.3. Técnica de aprendizaje directo. Este código se utiliza para la generación de la señal MIMO 2x2, su posterior paso por el amplificador y la aplicación a la señal de las técnicas DLA.

```

clear all, close all

nmse = @(ymod,y) 20*log10(norm(ymod-y)/norm(y));
dBm = @(x) 10*log10(rms(x).^2/100)+30;
scale_dBm = @(x,P) x*10^((P-dBm(x))/20);
PAPR = @(x) 20*log10(max(abs(x))/rms(x));
dBminst = @(x) 10*log10(abs(x).^2/100)+30;

alpha = 15; gamma = [1,0.1,0.1]; SNR = 65;
RMSin = -24.5; fc = 2e9;
a = 0.0316;
b = 0.0316;
d = 0.0316;
g = 0.0316;

configurationMIMO;

mu = 1/3;
Nmeas = 1; % Number of measurements
Nave = 10;
Niter = 20; % Iteration
Nval = 3;

Foff = 0;
fsmcd = 1024*30e3*5; %153.6MHz

```

```

fsint = 80e6;

[u1, xs1, ~, dataxid1] = genera_5GNR_multicarrier_v4(50, 30e3, 20e6, Foff, fsmed, 8,
1024);
[u2, xs2, ~, dataxid2] = genera_5GNR_multicarrier_v4(50, 30e3, 20e6, Foff, fsmed, 8,
2048);

%% First measurement and initialization
u1 = scale_dBm(u1,RMSin);
u2 = scale_dBm(u2,RMSin);
y1_int = syntheticPA(u1 +a.*u2, alpha, gamma, SNR);
y2_int = syntheticPA(u2 + b.*u1, alpha, gamma, SNR);
y1 = y1_int+d.*y2_int;
y2 = y2_int+g.*y1_int;
Pout = dBm(y1);
Pin = dBm(u1);
figure(10), dibuja_AMAM_AMPM(u1,y1);
% Ganancia Estática
G1 = norm(y1)/norm(u1);
G2 = norm(y2)/norm(u2);

%% Calculation of maximum memory buffers (Qpmax) and (Qnmax), 2D-GMP
Qa1 = max(x1_limits.GMPLC.La);
Qa2 = max(x2_limits.GMPLC.La);
Qb1 = max(x1_limits.GMPLC.Lb) + max(x1_limits.GMPLC.Mb);
Qb2 = max(x2_limits.GMPLC.Lb) + max(x2_limits.GMPLC.Mb);
Qc1 = max(x1_limits.GMPLC.Lc) - min(x1_limits.GMPLC.Mc);
Qc2 = max(x2_limits.GMPLC.Lc) - min(x2_limits.GMPLC.Mc);

Qpmax1 = max([Qa1, Qb1, Qc1]);
Qpmax2 = max([Qa2, Qb2, Qc2]);

Qnmax1 = max([x1_limits.GMPLC.Mc, 0]);
Qnmax2 = max([x2_limits.GMPLC.Mc, 0]);

% Number of samples
N = length(y1);
%Construcción de la matriz de regresores
[U1, Rmat1] = buildW_GMPLC(u1, u2, [N-Qpmax1+1:N, 1:N, 1:Qnmax1],...
Qpmax1, Qnmax1, x1_limits, x2_limits);
[U2, Rmat2] = buildW_GMPLC(u2, u1, [N-Qpmax2+1:N, 1:N, 1:Qnmax2],...
Qpmax2, Qnmax2, x2_limits, x1_limits);

indices1 = sel_indices(y1,y1,0.1);
indices2 = sel_indices(y2,y2,0.1);
% [~, sR, noptR, ~, ~] = RCDOMP(UR(indices,:), yRs(indices), RmatR, 51);
[~, s1, nopt1, ~] = omp_domp(U1(indices1,:), y1(indices1), Rmat1, 'DOMP');
[~, s2, nopt2, ~] = omp_domp(U2(indices2,:), y2(indices2), Rmat2, 'DOMP');
cL = nopt1;
cR = nopt2;
s1 = s1(1:cL).';
s2 = s2(1:cR).';

%Inicialización del modelo:
w1 = zeros(cL,1);
dw1 = zeros(cL,1);
w2 = zeros(cR,1);
dw2 = zeros(cR,1);

```

```

U1_norm = U1(:,s1);
U2_norm = U2(:,s2);
U1_inv = (inv(U1_norm'*U1_norm)*U1_norm');
U2_inv = (inv(U2_norm'*U2_norm)*U2_norm');
%% DLA loop

for iter=1:Niter
fprintf(['Iter: ' num2str(iter) '\n']);
%DPD:
x1 = u1-U1(:,s1)*w1;
x2 = u2-U2(:,s2)*w2;

% PAPRx1 = 20*log10(max(abs(x1))/rms(x1));
% disp(sprintf('PAPR antes de clipping: %4.2f dB\n', PAPRx1));
% PAPRd = 12;
% clip = (10^((PAPRd-PAPRx1)/20))*max(abs(x1));
% x1(abs(x1)>clip) = clip*exp(1i*angle(x1(abs(x1)>clip)));
% disp(sprintf('PAPR después de clipping: %4.2f dB\n', PAPR(x1)));
%
% PAPRx2 = 20*log10(max(abs(x2))/rms(x2));
% disp(sprintf('PAPR antes de clipping: %4.2f dB\n', PAPRx2));
% PAPRd = 12;
% clip = (10^((PAPRd-PAPRx2)/20))*max(abs(x2));
% x2(abs(x2)>clip) = clip*exp(1i*angle(x2(abs(x2)>clip)));
% disp(sprintf('PAPR después de clipping: %4.2f dB\n', PAPR(x2)));

%Etapa de Síntesis
y1_int = syntheticPA(x1 + a.*x2, alpha, gamma, SNR);
y2_int = syntheticPA(x2 + b.*x1, alpha, gamma, SNR);
y1 = y1_int+d.*y2_int;
y2 = y2_int+g.*y1_int;

%Etapa de Análisis
e1 = y1/G1-u1;
e2 = y2/G2-u2;

%Adaptación DPD:
dw1 = U1_inv*e1;
w1 = w1 + mu * dw1;
dw2 = U2_inv*e2;
w2 = w2 + mu * dw2;

[ACPRx1, ~, ~, ~, nmse1(iter), ~] = analiza_medidas_5GNR_v2(u1, y1, dataxid1, fc,
400);
[ACPRx2, ~, ~, ~, nmse2(iter), ~] = analiza_medidas_5GNR_v2(u2, y2, dataxid2, fc,
405);
ACPR1_1(iter)=ACPRx1(1);
ACPR1_2(iter)=ACPRx1(2);
ACPR2_1(iter)=ACPRx2(1);
ACPR2_2(iter)=ACPRx2(2);

end
figure(10), dibuja_AMAM_AMPM(u1,y1);
figure(2), plot(1:Niter, nmse1, 'Marker', '*', 'Color', 'r'), xlabel('Iteración'),
ylabel('NMSE 1'), title('NMSE predistorsión x1'), grid on

```

```

figure(3), plot(1:Niter, nmse2, 'Marker', '*', 'Color', 'g'), xlabel('Iteración'),
ylabel('NMSE 2'), title('NMSE predistorción x2'), grid on
figure(4), plot(1:Niter, ACPR1_1, 'Marker', '*', 'Color', 'b'), xlabel('Iteración'),
ylabel('ACPR 1'), title('ACPR predistorción x1'), grid on
figure(5), plot(1:Niter, ACPR1_2, 'Marker', '*', 'Color', 'b', 'LineStyle', '--'),
xlabel('Iteración'), ylabel('ACPR 1'), title('ACPR predistorción x1'), grid on
figure(6), plot(1:Niter, ACPR2_1, 'Marker', '*', 'Color', 'y'), xlabel('Iteración'),
ylabel('ACPR 2'), title('ACPR predistorción x2'), grid on
figure(7), plot(1:Niter, ACPR2_2, 'Marker', '*', 'Color', 'y', 'LineStyle', '--'),
xlabel('Iteración'), ylabel('ACPR 1'), title('ACPR predistorción x1'), grid on

%% Validación

[u1Val,~,~,datau1] = genera_5GNR_multicarrier_v4(50, 30e3 ,20e6, Foff, fsmed, 8, 1);
[u2Val,~,~,datau2] = genera_5GNR_multicarrier_v4(50, 30e3 ,20e6, Foff, fsmed, 8, 2);
u1Val = scale_dBm(u1Val,RMSin);
u2Val = scale_dBm(u2Val,RMSin);
[U1, Rmat1] = buildW_GMPLC(u1Val, u2Val, [N-Qpmax1+1:N , 1:N , 1:Qnmax1],...
Qpmax1, Qnmax1, x1_limits, x2_limits);
[U2, Rmat2] = buildW_GMPLC(u2Val, u1Val, [N-Qpmax2+1:N , 1:N , 1:Qnmax2],...
Qpmax2, Qnmax2, x2_limits, x1_limits);
x1 = u1Val-U1(:,s1)*w1;
x2 = u2Val-U2(:,s2)*w2;

y1_int = syntheticPA(x1 +a.*x2, alpha, gamma, SNR);
y2_int = syntheticPA(x2 + b.*x1, alpha, gamma, SNR);
y1 = y1_int+d.*y2_int;
y2 = y2_int+g.*y1_int;

[ACPRx1val, ~, ~, ~, nmse1val, ~] = analiza_medidas_5GNR_v2(u1Val, y1, datau1, fc,
500);
[ACPRx2val, ~, ~, ~, nmse2val, ~] = analiza_medidas_5GNR_v2(u2Val, y2, datau2, fc,
505);

fprintf('NMSE validacion: %4.2f dB / %4.2f dB\n', nmse1val,nmse2val);
fprintf('ACPR: %4.2f dB / %4.2f dB\n', ACPR1val,ACPR2val)

```

Anexo B.4. Generación de los cuatro modelos. Estos cuatro códigos independientes son los que generan las ecuaciones matriciales de los cuatro modelos utilizados en este trabajo, explicados en la sección 4 de este trabajo.

```

function [Xmat, Rmat] = buildW_GMPLC(x1, x2, n, Qpmax, Qnmax, x1_limits, x2_limits)
% This function builds the measurement matrix of the 2-D GMP-modified model
%
%
%Only one side

%% GMPLC block
Xa1_GMPLC= []; Xb1_GMPLC = []; Xc1_GMPLC = [];
Xa2_GMPLC= []; Xb2_GMPLC = []; Xc2_GMPLC = [];
Regr_a1_GMPLC = []; Repr_b1_GMPLC= []; Repr_c1_GMPLC= [];
Regr_a2_GMPLC = []; Repr_b2_GMPLC= []; Repr_c2_GMPLC= [];
% Repr_a_GMPLC = []; Repr_b_GMPLC= []; Repr_c_GMPLC= [];
% x1 Linear crosstalk
indk = 0;
for k = 1:length(x1_limits.GMPLC.Ka)

```

```

for l = 0:x1_limits.GMPLC.La(k)
    indk = indk + 1;
    Xa1_GMPLC(:,indk) = x1(n(1+Qpmax-1:end-Qnmax-1)).*(abs(x1(n(1+Qpmax-1:end-
Qnmax-1))).^(x1_limits.GMPLC.Ka(k)));
    if(x1_limits.GMPLC.Ka(k)~=0)
        Reqr_a1_GMPLC{indk} = sprintf('x1(n-%d)*|x1(n-
%d)|^{%d}',l,l,x1_limits.GMPLC.Ka(k));
    else
        Reqr_a1_GMPLC{indk} = sprintf('x1(n-%d)',l);
    end
end
end
%Conjugate branch
% for l = 0:x1_limits.intra.La(1)
%     indk = indk + 1;
%     Xa_intra(:,indk) = conj(x1(n(1+Qpmax-1:end-Qnmax-1)));
%     Reqr_a_intra{indk} = sprintf('conj_x1(n-%d)',l);
% end

indk = 0;
for k = 1:length(x1_limits.GMPLC.Kb)
    for l = 0:x1_limits.GMPLC.Lb(k)
        for m = 1:x1_limits.GMPLC.Mb(k)
            indk = indk + 1;
            Xb1_GMPLC(:,indk) = x1(n(1+Qpmax-1:end-Qnmax-1)).*(abs(x1(n(1+Qpmax-1-
m:end-Qnmax-1-m))).^(x1_limits.GMPLC.Kb(k)));
            Reqr_b1_GMPLC{indk} = sprintf('x1(n-%d)|x1(n-
%d)|^{%d}',l,l+m,x1_limits.GMPLC.Kb(k));
        end
    end
end

indk = 0;
for k = 1:length(x1_limits.GMPLC.Kc)
    for l = 0:x1_limits.GMPLC.Lc(k)
        for m = 1:x1_limits.GMPLC.Mc(k)
            indk = indk + 1;
            Xc1_GMPLC(:,indk) = x1(n(1+Qpmax-1:end-Qnmax-1)).*(abs(x1(n(1+Qpmax-
1+m:end-Qnmax-1+m))).^(x1_limits.GMPLC.Kc(k)));
            Reqr_c1_GMPLC{indk} = sprintf('x1(n-%d)|x1(n-%d)|^{%d}',l,l-
m,x1_limits.GMPLC.Kc(k));
        end
    end
end

%x2 linear crosstalk
indk = 0;
for k = 1:length(x2_limits.GMPLC.Ka)
    for l = 0:x2_limits.GMPLC.La(k)
        indk = indk + 1;
        Xa2_GMPLC(:,indk) = x2(n(1+Qpmax-1:end-Qnmax-1)).*(abs(x2(n(1+Qpmax-1:end-
Qnmax-1))).^(x2_limits.GMPLC.Ka(k)));
        if(x2_limits.GMPLC.Ka(k)~=0)
            Reqr_a2_GMPLC{indk} = sprintf('x2(n-%d)*|x2(n-
%d)|^{%d}',l,l,x1_limits.GMPLC.Ka(k));
        else
            Reqr_a2_GMPLC{indk} = sprintf('x2(n-%d)',l);
        end
    end
end
end

```

```

indk = 0;
for k = 1:length(x2_limits.GMPLC.Kb)
    for l = 0:x2_limits.GMPLC.Lb(k)
        for m = 1:x2_limits.GMPLC.Mb(k)
            indk = indk + 1;
            Xb2_GMPLC(:,indk) = x2(n(1+Qpmax-1:end-Qnmax-1)).*(abs(x2(n(1+Qpmax-1-
m:end-Qnmax-1-m))).^(x2_limits.GMPLC.Kb(k)));
            Reqr_b2_GMPLC{indk} = sprintf('x2(n-%d)|x2(n-
%d)|^{%d}',l,l+m,x2_limits.GMPLC.Kb(k));
        end
    end
end

indk = 0;
for k = 1:length(x2_limits.GMPLC.Kc)
    for l = 0:x2_limits.GMPLC.Lc(k)
        for m = 1:x2_limits.GMPLC.Mc(k)
            indk = indk + 1;
            Xc2_GMPLC(:,indk) = x2(n(1+Qpmax-1:end-Qnmax-1)).*(abs(x2(n(1+Qpmax-
1+m:end-Qnmax-1+m))).^(x2_limits.GMPLC.Kc(k)));
            Reqr_c2_GMPLC{indk} = sprintf('x2(n-%d)|x2(n-%d)|^{%d}',l,l-
m,x2_limits.GMPLC.Kc(k));
        end
    end
end

Xmat = [Xa1_GMPLC Xb1_GMPLC Xc1_GMPLC Xa2_GMPLC Xb2_GMPLC Xc2_GMPLC];
Rmat = [Regr_a1_GMPLC, Reqr_b1_GMPLC, Reqr_c1_GMPLC, Reqr_a2_GMPLC, Reqr_b2_GMPLC,
Reqr_c2_GMPLC]';

```

```

function [Xmat, Rmat] = buildW_GMPNLC(x1, x2, n, Qpmax, Qnmax, x1_limits, x2_limits)
Regra1_GMPNLC = [];
Regrb1_GMPNLC = [];
Regrc1_GMPNLC = [];
Xa1_GMPNLC = [];
Xb1_GMPNLC = [];
Xc1_GMPNLC = [];
Regra2_GMPNLC = [];
Regrb2_GMPNLC = [];
Regrc2_GMPNLC = [];
Xa2_GMPNLC = [];
Xb2_GMPNLC = [];
Xc2_GMPNLC = [];

indk = 0;
for k1 = 1:length(x1_limits.GMPNLC.Ka)
    for k2 = 1:length(x1_limits.GMPNLC.Ka)
        for l = 0:x1_limits.GMPNLC.La(k1)

            indk = indk + 1;
            Xa1_GMPNLC(:,indk) = x1(n(1+Qpmax-1:end-Qnmax-1)).*(abs(x1(n(1+Qpmax-
1:end-Qnmax-1))).^(x1_limits.GMPNLC.Ka(k1))).*(abs(x2(n(1+Qpmax-1:end-Qnmax-
1))).^(x1_limits.GMPNLC.Ka(k2)));
            Regra1_GMPNLC{indk} = sprintf('x1(n-%d)*|x1(n-%d)|^{%d}*|x2(n-
%d)|^{%d}',l,l,x1_limits.GMPNLC.Ka(k1),l,x1_limits.GMPNLC.Ka(k2));

        end
    end
end

```

```

end
indk = 0;
for k1 = 1:length(x1_limits.GMPNLC.Kb)
    for k2 = 1:length(x1_limits.GMPNLC.Kb)
        for l = 0:x1_limits.GMPNLC.Lb(k1)
            for m = 1:x1_limits.GMPNLC.Mb(k1)
                indk = indk + 1;
                Xb1_GMPNLC(:,indk) = x1(n(1+Qpmax-1:end-Qnmax-1)).*(abs(x1(n(1+Qpmax-1-m:end-Qnmax-1-m))).^(x1_limits.GMPNLC.Kb(k1))).*(abs(x2(n(1+Qpmax-1-m:end-Qnmax-1-m))).^(x1_limits.GMPNLC.Kb(k2))));
                Regrb1_GMPNLC{indk} = sprintf('x1(n-%d)*|x1(n-%d)|^{%d}*|x2(n-%d)|^{%d}',l,l+m,x1_limits.GMPNLC.Kb(k1),l+m,x1_limits.GMPNLC.Kb(k2));
            end
        end
    end
end
indk = 0;
for k1 = 1:length(x1_limits.GMPNLC.Kc)
    for k2 = 1:length(x1_limits.GMPNLC.Kc)
        for l = 0:x1_limits.GMPNLC.Lc(k1)
            for m = 1:x1_limits.GMPNLC.Mc(k1)
                indk = indk + 1;
                Xc1_GMPNLC(:,indk) = x1(n(1+Qpmax-1:end-Qnmax-1)).*(abs(x1(n(1+Qpmax-1+m:end-Qnmax-1+m))).^(x1_limits.GMPNLC.Kc(k1))).*(abs(x2(n(1+Qpmax-1+m:end-Qnmax-1+m))).^(x1_limits.GMPNLC.Kc(k2))));
                Regrc1_GMPNLC{indk} = sprintf('x1(n-%d)*|x1(n-%d)|^{%d}*|x2(n-%d)|^{%d}',l,l-m,x1_limits.GMPNLC.Kc(k1),l-m,x1_limits.GMPNLC.Kc(k2));
            end
        end
    end
end
indk = 0;
for k1 = 1:length(x2_limits.GMPNLC.Ka)
    for k2 = 1:length(x2_limits.GMPNLC.Ka)
        for l = 0:x2_limits.GMPNLC.La(k1)
            indk = indk + 1;
            Xa2_GMPNLC(:,indk) = x2(n(1+Qpmax-1:end-Qnmax-1)).*(abs(x2(n(1+Qpmax-1:end-Qnmax-1))).^(x2_limits.GMPNLC.Ka(k1))).*(abs(x1(n(1+Qpmax-1:end-Qnmax-1))).^(x2_limits.GMPNLC.Ka(k2))));
            Regra2_GMPNLC{indk} = sprintf('x2(n-%d)*|x2(n-%d)|^{%d}*|x1(n-%d)|^{%d}',l,l,x2_limits.GMPNLC.Ka(k1),l,x2_limits.GMPNLC.Ka(k2));
        end
    end
end
indk = 0;
for k1 = 1:length(x2_limits.GMPNLC.Kb)
    for k2 = 1:length(x2_limits.GMPNLC.Kb)
        for l = 0:x2_limits.GMPNLC.Lb(k1)
            for m = 1:x2_limits.GMPNLC.Mb(k1)
                indk = indk + 1;
                Xb2_GMPNLC(:,indk) = x2(n(1+Qpmax-1:end-Qnmax-1)).*(abs(x2(n(1+Qpmax-1-m:end-Qnmax-1-m))).^(x2_limits.GMPNLC.Kb(k1))).*(abs(x1(n(1+Qpmax-1-m:end-Qnmax-1-m))).^(x2_limits.GMPNLC.Kb(k2))));
                Regrb2_GMPNLC{indk} = sprintf('x2(n-%d)*|x2(n-%d)|^{%d}*|x1(n-%d)|^{%d}',l,l+m,x2_limits.GMPNLC.Kb(k1),l+m,x2_limits.GMPNLC.Kb(k2));
            end
        end
    end
end
end

```

```

end
indk = 0;
for k1 = 1:length(x2_limits.GMPNLC.Kc)
    for k2 = 1:length(x2_limits.GMPNLC.Kc)
        for l = 0:x2_limits.GMPNLC.Lc(k1)
            for m = 1:x2_limits.GMPNLC.Mc(k1)
                indk = indk + 1;
                Xc2_GMPNLC(:,indk) = x2(n(1+Qpmax-l:end-Qnmax-l)).*(abs(x2(n(1+Qpmax-
1+m:end-Qnmax-l+m))).^(x2_limits.GMPNLC.Kc(k1))).*(abs(x1(n(1+Qpmax-l+m:end-Qnmax-
1+m))).^(x2_limits.GMPNLC.Kc(k2)));
                Regrc2_GMPNLC{indk} = sprintf('x2(n-%d)*|x2(n-%d)|^{%d}*|x1(n-
%d)|^{%d}',l,l-m,x2_limits.GMPNLC.Kc(k1),l-m,x2_limits.GMPNLC.Kc(k2));
            end
        end
    end
end
Xmat = [Xa1_GMPNLC, Xb1_GMPNLC, Xc1_GMPNLC, Xa2_GMPNLC, Xb2_GMPNLC, Xc2_GMPNLC];
Rmat = [Regra1_GMPNLC, Regrb1_GMPNLC, Regrc1_GMPNLC, Regra2_GMPNLC, Regrb2_GMPNLC,
Regrc2_GMPNLC]';

```

```

function [Xmat, Rmat] = buildW_EGMPNLC(x1, x2, n, Qpmax, Qnmax, x1_limits, x2_limits)
Xb=[];
Rb=[];
Xc=[];
Rc=[];
indk = 0;
for k = 1:length(x1_limits.EGMPNLC.Kb)
    for l = 0:x1_limits.EGMPNLC.Lb(k)
        for m = 1:x1_limits.EGMPNLC.Mb(k)
            indk=indk+1; %indice 1
            Xb(:,indk) = x1(n(1+Qpmax-l:end-Qnmax-l)).*(abs(x1(n(1+Qpmax-l+m:end-
Qnmax-l+m))).^(x1_limits.EGMPNLC.Kb(k)));
            Rb{indk} = sprintf('x1(n-%d)*|x1(n-%d)|^{%d}',l,l-
m,x1_limits.EGMPNLC.Kb(k));
            indk=indk+1; %indice 2
            Xb(:,indk) = x1(n(1+Qpmax-l:end-Qnmax-l)).*(abs(x2(n(1+Qpmax-l+m:end-
Qnmax-l+m))).^(x2_limits.EGMPNLC.Kb(k)));
            Rb{indk} = sprintf('x1(n-%d)*|x2(n-%d)|^{%d}',l,l-
m,x2_limits.EGMPNLC.Kb(k));
            indk=indk+1; %indice 3
            Xb(:,indk) = x2(n(1+Qpmax-l:end-Qnmax-l)).*(abs(x1(n(1+Qpmax-l+m:end-
Qnmax-l+m))).^(x1_limits.EGMPNLC.Kb(k)));
            Rb{indk} = sprintf('x2(n-%d)*|x1(n-%d)|^{%d}',l,l-
m,x1_limits.EGMPNLC.Kb(k));
            indk=indk+1; %indice 4
            Xb(:,indk) = x2(n(1+Qpmax-l:end-Qnmax-l)).*(abs(x2(n(1+Qpmax-l+m:end-
Qnmax-l+m))).^(x2_limits.EGMPNLC.Kb(k)));
            Rb{indk} = sprintf('x2(n-%d)*|x2(n-%d)|^{%d}',l,l-
m,x2_limits.EGMPNLC.Kb(k));
            if x1_limits.EGMPNLC.Kb(k)-2>=0
                indk=indk+1; %indice 5
                Xb(:,indk) = x1(n(1+Qpmax-l:end-Qnmax-l)).*(abs(x1(n(1+Qpmax-l+m:end-
Qnmax-l+m))).^(x1_limits.EGMPNLC.Kb(k)-2).*x1(n(1+Qpmax-l+m:end-Qnmax-
l+m)).*conj(x2(n(1+Qpmax-l+m:end-Qnmax-l+m))));
                Rb{indk} = sprintf('x1(n-%d)*|x1(n-%d)|^{%d}*x1(n-%d)*x2*(n-%d)',l,l-
m,x1_limits.EGMPNLC.Kb(k)-2, l-m, l-m);
                indk=indk+1; %indice 6
                Xb(:,indk) = x1(n(1+Qpmax-l:end-Qnmax-l)).*(abs(x1(n(1+Qpmax-l+m:end-
Qnmax-l+m))).^(x1_limits.EGMPNLC.Kb(k)-2).*conj(x1(n(1+Qpmax-l+m:end-Qnmax-
l+m))).*x2(n(1+Qpmax-l+m:end-Qnmax-l+m)));
            end
        end
    end
end

```

```

        Rb{indk} = sprintf('x1(n-%d)*|x1(n-%d)|^{%d}*x1*(n-%d)*x2(n-%d)',1,1-
m,x1_limits.EGMPNLC.Kb(k)-2, 1-m, 1-m);
        indk=indk+1; %indice 7
        Xb(:,indk) = x1(n(1+Qpmax-1:end-Qnmax-1)).*(abs(x2(n(1+Qpmax-1+m:end-
Qnmax-1+m))).^(x2_limits.EGMPNLC.Kb(k)-2).*x1(n(1+Qpmax-1+m:end-Qnmax-
1+m)).*conj(x2(n(1+Qpmax-1+m:end-Qnmax-1+m)))));
        Rb{indk} = sprintf('x1(n-%d)*|x2(n-%d)|^{%d}*x1(n-%d)*x2*(n-%d)',1,1-
m,x2_limits.EGMPNLC.Kb(k)-2, 1-m, 1-m);
        indk=indk+1; %indice 8
        Xb(:,indk) = x1(n(1+Qpmax-1:end-Qnmax-1)).*(abs(x2(n(1+Qpmax-1+m:end-
Qnmax-1+m))).^(x2_limits.EGMPNLC.Kb(k)-2).*conj(x1(n(1+Qpmax-1+m:end-Qnmax-
1+m))).*x2(n(1+Qpmax-1+m:end-Qnmax-1+m)))));
        Rb{indk} = sprintf('x1(n-%d)*|x2(n-%d)|^{%d}*x1*(n-%d)*x2(n-%d)',1,1-
m,x2_limits.EGMPNLC.Kb(k)-2, 1-m, 1-m);
        if rem(x1_limits.EGMPNLC.Kb(k),2)==0
            indk=indk+1; %indice 9
            Xb(:,indk) = x1(n(1+Qpmax-1:end-Qnmax-1)).*(abs(x1(n(1+Qpmax-1-
m:end-Qnmax-1-m))).^(x1_limits.EGMPNLC.Kb(k)/2)).*(abs(x2(n(1+Qpmax-1-m:end-Qnmax-1-
m))).^(x2_limits.EGMPNLC.Kb(k)/2)));
            Rb{indk} = sprintf('x1(n-%d)*|x1(n-%d)|^{%d}*|x2(n-
%d)|^{%d}',1,1+m,x1_limits.EGMPNLC.Kb(k)/2,1+m,x2_limits.EGMPNLC.Kb(k)/2);
            indk=indk+1; %indice 10
            Xb(:,indk) = x2(n(1+Qpmax-1:end-Qnmax-1)).*(abs(x1(n(1+Qpmax-1-
m:end-Qnmax-1-m))).^(x1_limits.EGMPNLC.Kb(k)/2)).*(abs(x2(n(1+Qpmax-1-m:end-Qnmax-1-
m))).^(x2_limits.EGMPNLC.Kb(k)/2)));
            Rb{indk} = sprintf('x2(n-%d)*|x1(n-%d)|^{%d}*|x2(n-
%d)|^{%d}',1,1+m,x1_limits.EGMPNLC.Kb(k)/2,1+m,x2_limits.EGMPNLC.Kb(k)/2);
        end
        indk=indk+1; %indice 11
        Xb(:,indk) = x2(n(1+Qpmax-1:end-Qnmax-1)).*(abs(x1(n(1+Qpmax-1+m:end-
Qnmax-1+m))).^(x1_limits.EGMPNLC.Kb(k)-2).*x1(n(1+Qpmax-1+m:end-Qnmax-
1+m)).*conj(x2(n(1+Qpmax-1+m:end-Qnmax-1+m)))));
        Rb{indk} = sprintf('x2(n-%d)*|x1(n-%d)|^{%d}*x1(n-%d)*x2*(n-%d)',1,1-
m,x1_limits.EGMPNLC.Kb(k)-2, 1-m, 1-m);
        indk=indk+1; %indice 12
        Xb(:,indk) = x2(n(1+Qpmax-1:end-Qnmax-1)).*(abs(x1(n(1+Qpmax-1+m:end-
Qnmax-1+m))).^(x1_limits.EGMPNLC.Kb(k)-2).*conj(x1(n(1+Qpmax-1+m:end-Qnmax-
1+m))).*x2(n(1+Qpmax-1+m:end-Qnmax-1+m)))));
        Rb{indk} = sprintf('x2(n-%d)*|x1(n-%d)|^{%d}*x1*(n-%d)*x2(n-%d)',1,1-
m,x1_limits.EGMPNLC.Kb(k)-2, 1-m, 1-m);
        indk=indk+1; %indice 13
        Xb(:,indk) = x2(n(1+Qpmax-1:end-Qnmax-1)).*(abs(x2(n(1+Qpmax-1+m:end-
Qnmax-1+m))).^(x2_limits.EGMPNLC.Kb(k)-2).*x1(n(1+Qpmax-1+m:end-Qnmax-
1+m)).*conj(x2(n(1+Qpmax-1+m:end-Qnmax-1+m)))));
        Rb{indk} = sprintf('x2(n-%d)*|x2(n-%d)|^{%d}*x1(n-%d)*x2*(n-%d)',1,1-
m,x2_limits.EGMPNLC.Kb(k)-2, 1-m, 1-m);
        indk=indk+1; %indice 14
        Xb(:,indk) = x2(n(1+Qpmax-1:end-Qnmax-1)).*(abs(x2(n(1+Qpmax-1+m:end-
Qnmax-1+m))).^(x2_limits.EGMPNLC.Kb(k)-2).*conj(x1(n(1+Qpmax-1+m:end-Qnmax-
1+m))).*x2(n(1+Qpmax-1+m:end-Qnmax-1+m)))));
        Rb{indk} = sprintf('x2(n-%d)*|x2(n-%d)|^{%d}*x1*(n-%d)*x2(n-%d)',1,1-
m,x2_limits.EGMPNLC.Kb(k)-2, 1-m, 1-m);
    end
end
end
end
indk = 0;
for k = 1:length(x1_limits.EGMPNLC.Kc)
    for l = 0:x1_limits.EGMPNLC.Lc(k)
        for m = 1:x1_limits.EGMPNLC.Mc(k)

```

```

        indk=indk+1; %indice 1
        Xc(:,indk) = x1(n(1+Qpmax-1:end-Qnmax-1)).*(abs(x1(n(1+Qpmax-1-m:end-
Qnmax-1-m))).^(x1_limits.EGMPNLC.Kc(k)));
        Rc{indk} = sprintf('x1(n-%d)*|x1(n-
%d)|^{%d}',l,l+m,x1_limits.EGMPNLC.Kc(k));
        indk=indk+1; %indice 2
        Xc(:,indk) = x1(n(1+Qpmax-1:end-Qnmax-1)).*(abs(x2(n(1+Qpmax-1-m:end-
Qnmax-1-m))).^(x2_limits.EGMPNLC.Kc(k)));
        Rc{indk} = sprintf('x1(n-%d)*|x2(n-
%d)|^{%d}',l,l+m,x2_limits.EGMPNLC.Kc(k));
        indk=indk+1; %indice 3
        Xc(:,indk) = x2(n(1+Qpmax-1:end-Qnmax-1)).*(abs(x1(n(1+Qpmax-1-m:end-
Qnmax-1-m))).^(x1_limits.EGMPNLC.Kc(k)));
        Rc{indk} = sprintf('x2(n-%d)*|x1(n-
%d)|^{%d}',l,l+m,x1_limits.EGMPNLC.Kc(k));
        indk=indk+1; %indice 4
        Xc(:,indk) = x2(n(1+Qpmax-1:end-Qnmax-1)).*(abs(x2(n(1+Qpmax-1-m:end-
Qnmax-1-m))).^(x2_limits.EGMPNLC.Kc(k)));
        Rc{indk} = sprintf('x2(n-%d)*|x2(n-
%d)|^{%d}',l,l+m,x2_limits.EGMPNLC.Kc(k));
        if x1_limits.EGMPNLC.Kc(k)-2>=0
            indk=indk+1; %indice 5
            Xc(:,indk) = x1(n(1+Qpmax-1:end-Qnmax-1)).*(abs(x1(n(1+Qpmax-1-m:end-
Qnmax-1-m))).^(x1_limits.EGMPNLC.Kc(k)-2).*x1(n(1+Qpmax-1-m:end-Qnmax-1-
m)).*conj(x2(n(1+Qpmax-1-m:end-Qnmax-1-m))));
            Rc{indk} = sprintf('x1(n-%d)*|x1(n-%d)|^{%d}*x1(n-%d)*x2*(n-
%d)',l,l+m,x1_limits.EGMPNLC.Kc(k)-2, l+m, l+m);
            indk=indk+1; %indice 6
            Xc(:,indk) = x1(n(1+Qpmax-1:end-Qnmax-1)).*(abs(x1(n(1+Qpmax-1-m:end-
Qnmax-1-m))).^(x1_limits.EGMPNLC.Kc(k)-2).*conj(x1(n(1+Qpmax-1-m:end-Qnmax-1-
m))).*x2(n(1+Qpmax-1-m:end-Qnmax-1-m))));
            Rc{indk} = sprintf('x1(n-%d)*|x1(n-%d)|^{%d}*x1*(n-%d)*x2(n-
%d)',l,l+m,x1_limits.EGMPNLC.Kc(k)-2, l+m, l+m);
            indk=indk+1; %indice 7
            Xc(:,indk) = x1(n(1+Qpmax-1:end-Qnmax-1)).*(abs(x2(n(1+Qpmax-1-m:end-
Qnmax-1-m))).^(x2_limits.EGMPNLC.Kc(k)-2).*x1(n(1+Qpmax-1-m:end-Qnmax-1-
m)).*conj(x2(n(1+Qpmax-1-m:end-Qnmax-1-m))));
            Rc{indk} = sprintf('x1(n-%d)*|x2(n-%d)|^{%d}*x1(n-%d)*x2*(n-
%d)',l,l+m,x2_limits.EGMPNLC.Kc(k)-2, l+m, l+m);
            indk=indk+1; %indice 8
            Xc(:,indk) = x1(n(1+Qpmax-1:end-Qnmax-1)).*(abs(x2(n(1+Qpmax-1-m:end-
Qnmax-1-m))).^(x2_limits.EGMPNLC.Kc(k)-2).*conj(x1(n(1+Qpmax-1-m:end-Qnmax-1-
m))).*x2(n(1+Qpmax-1-m:end-Qnmax-1-m))));
            Rc{indk} = sprintf('x1(n-%d)*|x2(n-%d)|^{%d}*x1*(n-%d)*x2(n-
%d)',l,l+m,x2_limits.EGMPNLC.Kc(k)-2, l+m, l+m);
            if rem(x1_limits.EGMPNLC.Kc(k),2)==0
                indk=indk+1; %indice 9
                Xc(:,indk) = x1(n(1+Qpmax-1:end-Qnmax-1)).*(abs(x1(n(1+Qpmax-1-
m:end-Qnmax-1-m))).^(x1_limits.EGMPNLC.Kc(k)/2)).*(abs(x2(n(1+Qpmax-1-m:end-Qnmax-1-
m))).^(x2_limits.EGMPNLC.Kc(k)/2));
                Rc{indk} = sprintf('x1(n-%d)*|x1(n-%d)|^{%d}*|x2(n-
%d)|^{%d}',l,l+m,x1_limits.EGMPNLC.Kc(k)/2,l+m,x2_limits.EGMPNLC.Kc(k)/2);
                indk=indk+1; %indice 10
                Xc(:,indk) = x2(n(1+Qpmax-1:end-Qnmax-1)).*(abs(x1(n(1+Qpmax-1-
m:end-Qnmax-1-m))).^(x1_limits.EGMPNLC.Kc(k)/2)).*(abs(x2(n(1+Qpmax-1-m:end-Qnmax-1-
m))).^(x2_limits.EGMPNLC.Kc(k)/2));
                Rc{indk} = sprintf('x2(n-%d)*|x1(n-%d)|^{%d}*|x2(n-
%d)|^{%d}',l,l+m,x1_limits.EGMPNLC.Kc(k)/2,l+m,x2_limits.EGMPNLC.Kc(k)/2);
            end
            indk=indk+1; %indice 11

```

```

        Xc(:,indk) = x2(n(1+Qpmax-1:end-Qnmax-1)).*(abs(x1(n(1+Qpmax-1-m:end-
Qnmax-1-m))).^(x1_limits.EGMPNLC.Kc(k)-2).*x1(n(1+Qpmax-1-m:end-Qnmax-1-
m)).*conj(x2(n(1+Qpmax-1-m:end-Qnmax-1-m))));
        Rc{indk} = sprintf('x2(n-%d)*|x1(n-%d)|^{%d}*x1(n-%d)*x2*(n-
%d)',1,l+m,x1_limits.EGMPNLC.Kc(k)-2, l+m, l+m);
        indk=indk+1; %indice 12
        Xc(:,indk) = x2(n(1+Qpmax-1:end-Qnmax-1)).*(abs(x1(n(1+Qpmax-1-m:end-
Qnmax-1-m))).^(x1_limits.EGMPNLC.Kc(k)-2).*conj(x1(n(1+Qpmax-1-m:end-Qnmax-1-
m))).*x2(n(1+Qpmax-1-m:end-Qnmax-1-m))));
        Rc{indk} = sprintf('x2(n-%d)*|x1(n-%d)|^{%d}*x1*(n-%d)*x2(n-
%d)',1,l+m,x1_limits.EGMPNLC.Kc(k)-2, l+m, l+m);
        indk=indk+1; %indice 13
        Xc(:,indk) = x2(n(1+Qpmax-1:end-Qnmax-1)).*(abs(x2(n(1+Qpmax-1-m:end-
Qnmax-1-m))).^(x2_limits.EGMPNLC.Kc(k)-2).*x1(n(1+Qpmax-1-m:end-Qnmax-1-
m)).*conj(x2(n(1+Qpmax-1-m:end-Qnmax-1-m))));
        Rc{indk} = sprintf('x2(n-%d)*|x2(n-%d)|^{%d}*x1(n-%d)*x2*(n-
%d)',1,l+m,x2_limits.EGMPNLC.Kc(k)-2, l+m, l+m);
        indk=indk+1; %indice 14
        Xc(:,indk) = x2(n(1+Qpmax-1:end-Qnmax-1)).*(abs(x2(n(1+Qpmax-1-m:end-
Qnmax-1-m))).^(x2_limits.EGMPNLC.Kc(k)-2).*conj(x1(n(1+Qpmax-1-m:end-Qnmax-1-
m))).*x2(n(1+Qpmax-1-m:end-Qnmax-1-m))));
        Rc{indk} = sprintf('x2(n-%d)*|x2(n-%d)|^{%d}*x1*(n-%d)*x2(n-
%d)',1,l+m,x2_limits.EGMPNLC.Kc(k)-2, l+m, l+m);
    end
end
end
end
Xmat=[Xb Xc];
Rmat=[Rb Rc]';

```

```

function [Xmat, Rmat] = buildW_PH2x2(x1, x2, n, Qpmax, Qnmax, x1_limits, x2_limits)

X1=[];
X2=[];
Regr1=[];
Regr2=[];
indk = 0;
for k1 = 1:length(x1_limits.PH2x2.Ka)
    for k2 = 1:(length(x1_limits.PH2x2.Ka)-k1+1)
        for l = 0:x1_limits.PH2x2.La(k1)
            indk = indk + 1;
            X1(:,indk) = x1(n(1+Qpmax-1:end-Qnmax-1)).*(abs(x1(n(1+Qpmax-1:end-Qnmax-
1))).^(x1_limits.PH2x2.Ka(k1))).*(abs(x2(n(1+Qpmax-1:end-Qnmax-
1))).^(x1_limits.PH2x2.Ka(k2))));
            Regr1{indk} = sprintf('x1(n-%d)*|x1(n-%d)|^{%d}*|x2(n-
%d)|^{%d}',1,l,x1_limits.PH2x2.Ka(k1),l,x1_limits.PH2x2.Ka(k2));
        end
    end
end

indk = 0;
for k1 = 1:length(x2_limits.PH2x2.Ka)
    for k2 = 1:(length(x2_limits.PH2x2.Ka)-k1+1)
        for l = 0:x2_limits.PH2x2.La(k1)
            indk = indk + 1;
            X2(:,indk) = x2(n(1+Qpmax-1:end-Qnmax-1)).*(abs(x2(n(1+Qpmax-1:end-Qnmax-
1))).^(x2_limits.PH2x2.Ka(k1))).*(abs(x1(n(1+Qpmax-1:end-Qnmax-
1))).^(x2_limits.PH2x2.Ka(k2))));
            Regr2{indk} = sprintf('x2(n-%d)*|x2(n-%d)|^{%d}*|x1(n-
%d)|^{%d}',1,l,x2_limits.PH2x2.Ka(k1),l,x2_limits.PH2x2.Ka(k2));
        end
    end
end

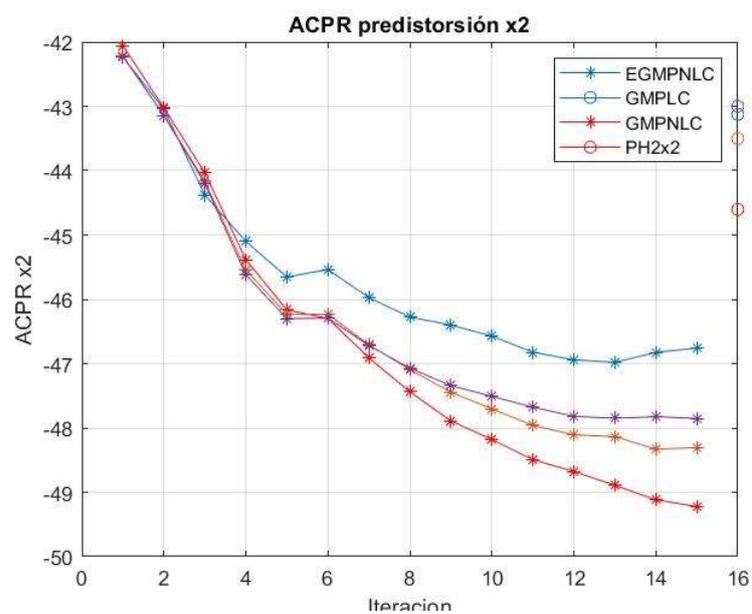
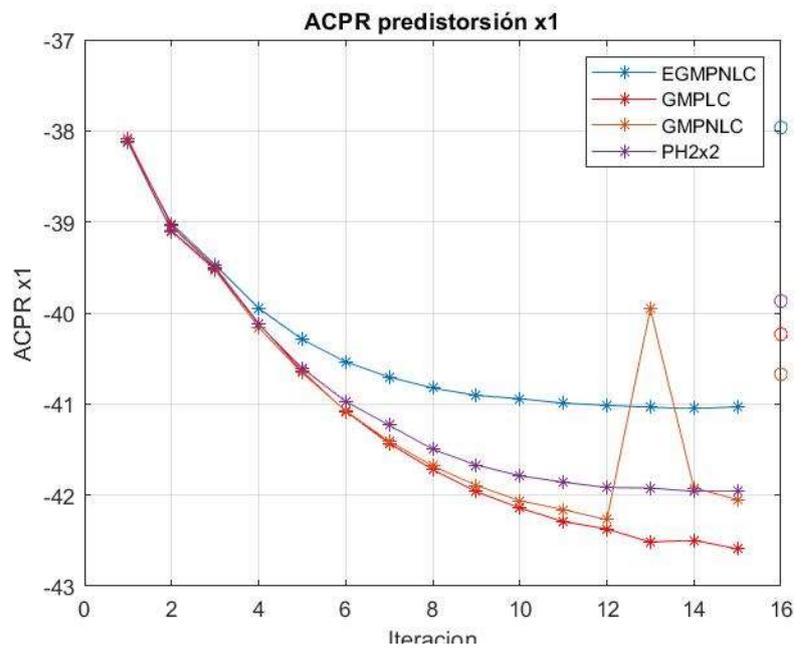
```

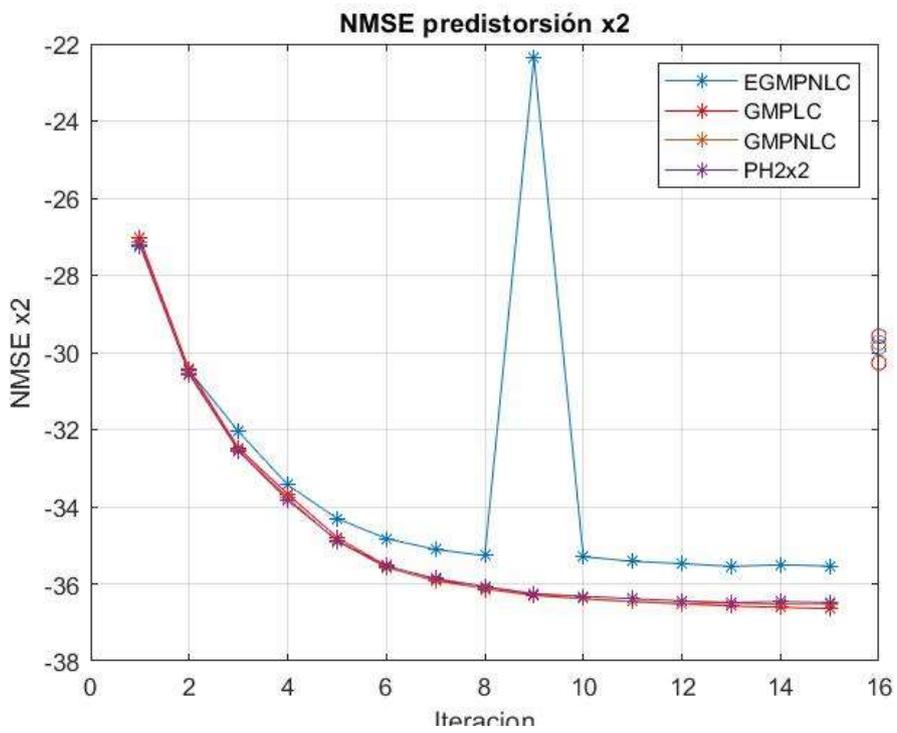
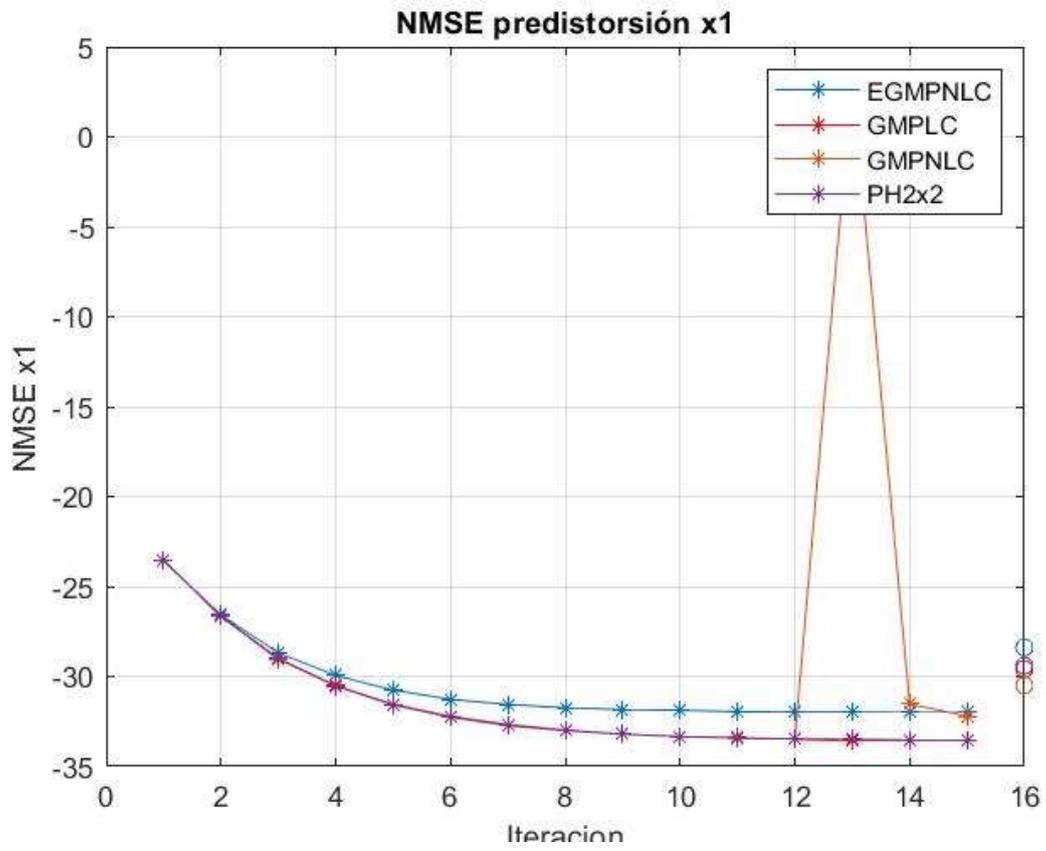
```
        end
    end
end
Xmat=[X1 X2];
Rmat=[Regr1 Regr2];
```

ANEXO C: RESULTADOS ADICIONALES

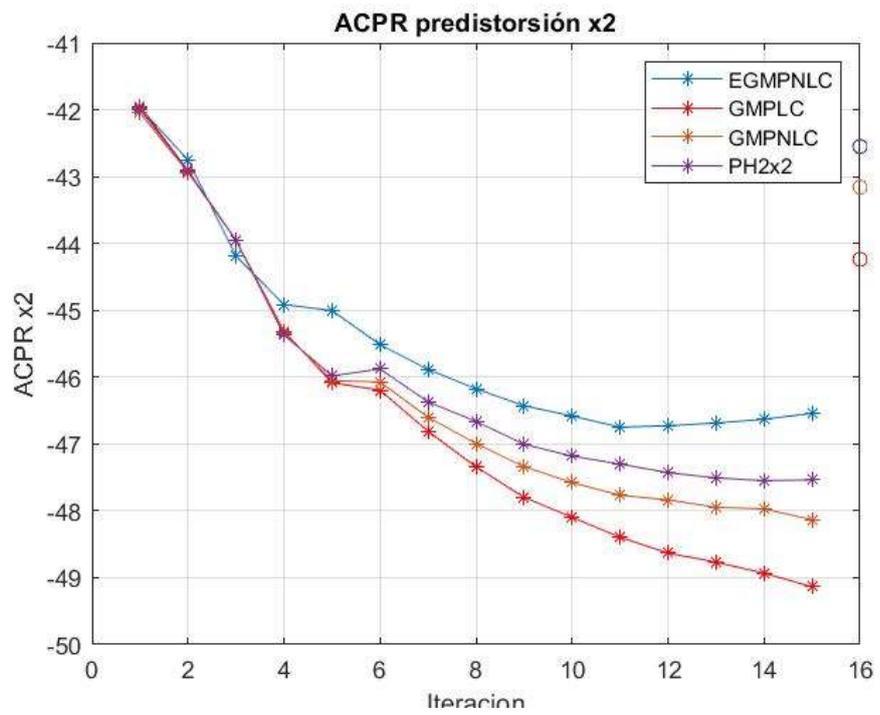
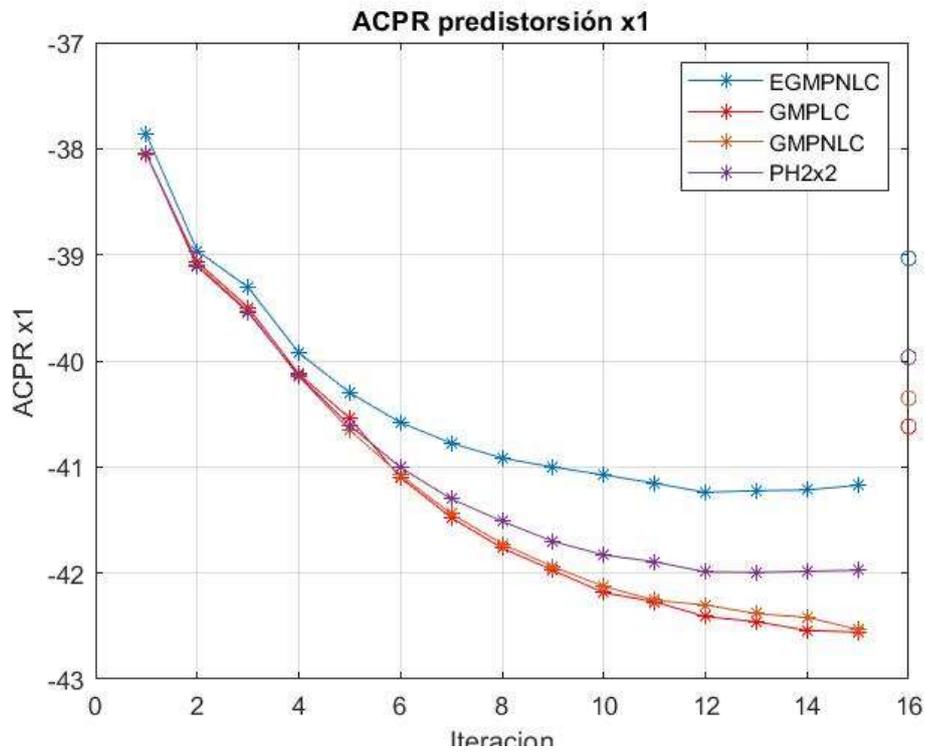
En este anexo se muestran los resultados de las pruebas 2 y 3 con el montaje de laboratorio explicado en la sección 6 de este trabajo.

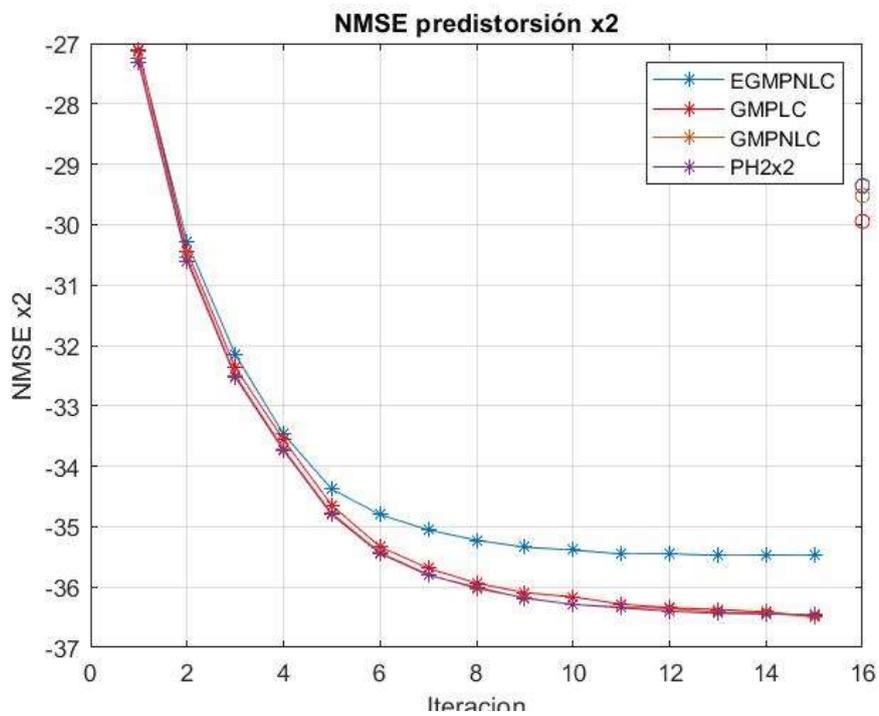
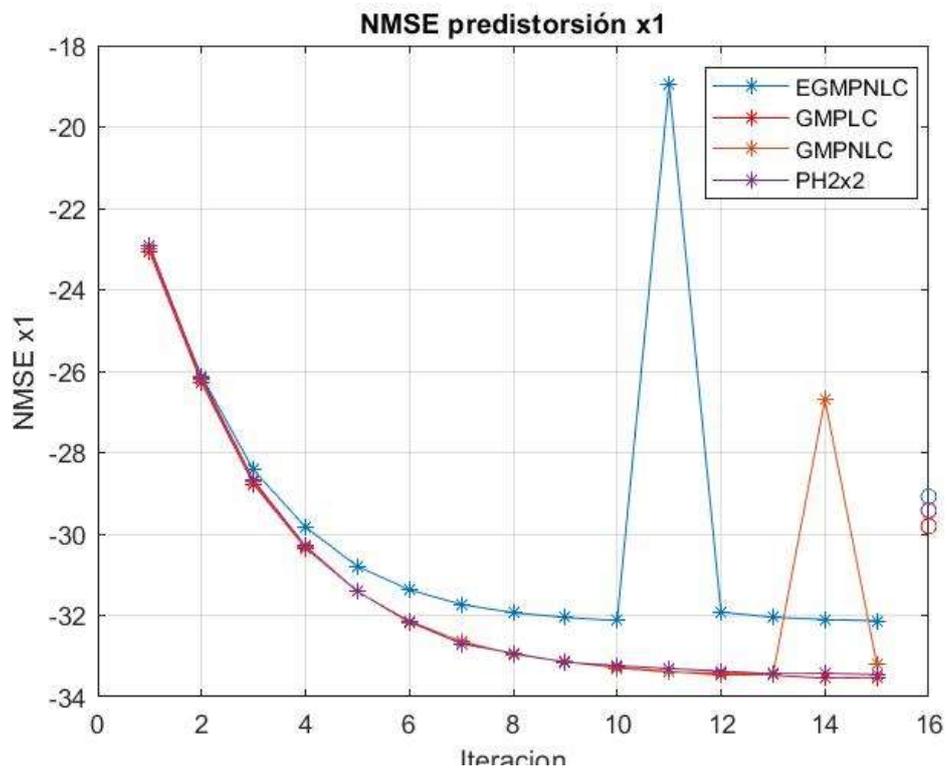
Anexo C.1. Resultados de NMSE y ACPR para la prueba 2.





Anexo C.2. Resultados de NMSE y ACPR para la prueba 3.





BIBLIOGRAFÍA

- [1] B. Jingfu, H. Jimgf, W. Huadong, W. Zhengde, “An Memory Polynomial Model for Power Amplifiers”, pp 1346-1349, IEEE International Conference on Communications, Circuits and Systems, 2008.
- [2] C. Crespo-Cadenas, M. J. Madero-Ayora, J. Reina-Tosina, and J. Becerra-González, “Formal deduction of a Volterra series model for complex-valued systems,” *Signal Processing*, vol. 131, pp. 245–248, 2017. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0165168416302043>
- [3] D. Morgan, Z. Ma, J. Kim, M. Zierdt, and J. Pastalan, “A generalized memory polynomial model for digital predistortion of rf power amplifiers,” *IEEE Transactions on Signal Processing*, vol. 54, no. 10, pp. 3852–3860, Oct. 2006.
- [4] D. Zhou and V. DeBrunner, “A novel adaptive nonlinear predistorter based on the direct learning algorithm,” in 2004 IEEE International Conference on Communications (IEEE Cat. No.04CH37577), vol. 4, Jun. 2004, pp. 2362–2366 Vol.4.
- [5] E. Dahlman, S. Parkvall, J. Skold, 5G NR The next generation wireless access technology, Academic Press, 1^a ed., 2018.
- [6] F. Ghannouchi, O. Hammi, M. Helaoui, “Behavioral modelling and predistortion of wideband wireless transmitters”, John Wiley & Sons, 2015.
- [7] H. Enzinger, K. Freiburger, and C. Vogel, “Competitive linearity for envelope tracking: Dual-band crest factor reduction and 2d-vector-switched digital predistortion,” *IEEE Microwave Magazine*, vol. 19, no. 1, pp. 69–77, Feb. 2018.
- [8] J. A. Becerra, M. J. Madero-Ayora, J. Reina-Tosina, C. Crespo-Cadenas, J. García-Frías, and G. Arce, “A doubly orthogonal matching pursuit algorithm for sparse predistortion of power amplifiers,” *IEEE Microwave and Wireless Components Letters*, vol. 28, no. 8, pp. 726–728, Aug. 2018.
- [9] J. Kim and K. Konstantinou, “Digital predistortion of wideband signals based on power amplifier model with memory,” *IEEE Transactions on Signal Processing*, vol. 37, no. 23, pp. 1417–1418, Nov. 2001.
- [10] J. Pedro and N. Carvalho, *Intermodulation distortion in microwave and wireless circuits*. Boston: Artech House, Jun. 2003.
- [11] J. Vuolevi and T. Rahkonen, *Distortion in RF power amplifiers*. Artech House, 2003.
- [12] J. Wood, *Behavioral modeling and linearization of RF power amplifiers*. Artech House, 2014.
- [13] S. A. Bassam, M. Helaoui, and F. M. Ghannouchi, “2-D digital predistortion (2-D-DPD) architecture for concurrent dual-band transmitters,” *IEEE Trans. Microw. Theory Techn.*, vol. 59, no. 10, pp. 2547–2553, Oct. 2011.
- [14] S. Amin, P. Händel, “Behavioral Modeling and Linearization of Crosstalk and Memory Effects in RF MIMO Transmitters”, *IEEE Transactions on Microwave Theory and Techniques*, Vol. 62, no. 4, 2014.

