

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

Identificación por mínimos cuadrados en
predistorsionadores de Volterra

Autor: Antonio Corral Sierra

Tutor: Luis Javier Reina Tosina

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

Sevilla, 2018



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

Identificación por mínimos cuadrados en predistorsionadores de Volterra

Autor:

Antonio Corral Sierra

Tutor:

Luis Javier Reina Tosina

Profesor Titular de Universidad

Dep. de Teoría de la Señal y Comunicaciones

Escuela Técnica Superior de Ingeniería

Universidad de Sevilla

Sevilla, 2018

Trabajo Fin de Grado: Identificación por mínimos cuadrados en predistorsionadores de Volterra

Autor: Antonio Corral Sierra

Tutor: Luis Javier Reina Tosina

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

Presidente:

Vocales:

Secretario:

Acuerdan otorgarle la calificación de:

Sevilla, febrero de 2018

El Secretario del Tribunal

A mi familia

A mis maestros

Agradecimientos

En primer lugar, agradecer a mi tutor y profesor Luis Javier Reina Tosina por el excelente trato que ha tenido conmigo durante el transcurso de este proyecto, por el interés mostrado en cada momento y por la ayuda recibida en tutoría.

Y en segundo lugar, a mi familia por haber confiado siempre en mí y transmitirme toda la fuerza y compañía que me han permitido llegar al final de esta carrera. Sin vuestra ayuda no hubiera sido posible.

Antonio Corral Sierra

Sevilla, 2018

Este trabajo de Fin de Grado aborda el problema de la linealización de amplificadores de potencia. En particular, se centra en los problemas numéricos que resultan de la inversión de matrices necesaria para la determinación de los parámetros de predistorsionadores digitales y posibles alternativas al cálculo de la matriz inversa, común en problemas de regresión. Se comienza planteando el contexto en el que estamos trabajando y se definen los objetivos que se esperan alcanzar. A continuación, se presentan las técnicas de linealización existentes, haciendo especial énfasis en la predistorsión digital. El tercer capítulo analiza el algoritmo más extendido en el cálculo de los coeficientes del predistorsionador, el algoritmo Least Square (LS). Una vez analizado, se discuten los problemas asociados al condicionamiento de las matrices de regresión y se plantean estrategias que intentan solventar las limitaciones de éste, manteniendo su exactitud en la estima.

El cuarto capítulo presenta la plataforma sobre la que se ha trabajado y realiza ciertas aclaraciones sobre el código generado. Las pruebas y resultados obtenidos al aplicar la predistorsión digital con los diferentes algoritmos de cálculo serán expuestos en el quinto capítulo.

El capítulo final recoge las conclusiones obtenidas en base a las pruebas realizadas y las líneas futuras de investigación. En él queda reflejado que los métodos analizados constituyen una alternativa eficiente para la resolución de los problemas identificados, así como la importancia de tener en cuenta la memoria, para una mejor linealización, a la hora de diseñar un predistorsionador.

Abstract

This Final Degree Project addresses the problem of the linearization of power amplifiers. In particular, it focuses on the numerical problems that result from the inversion of matrices necessary for the determination of the parameters of digital predistortionators and possible alternatives to the calculation of the inverse matrix, common in regression problems. It begins by raising the context in which we are working and defining the objectives that are expected to be achieved. Next, the existing linearization techniques are presented, with special emphasis on digital predistortion. The third chapter analyzes the most widespread algorithm in the calculation of predistortion coefficients, the Least Square (LS) algorithm. Once analyzed, the problems associated with the conditioning of the regression matrices are discussed and strategies are proposed that try to solve their limitations, maintaining its accuracy in the estimation.

The fourth chapter presents the platform on which we have worked and makes certain clarifications about the generated code. The tests and results obtained by applying the digital predistortion with the different calculation algorithms will be explained in the fifth chapter.

The final chapter includes the conclusions obtained based on the tests carried out and the future lines of research. It shows that the methods analyzed are an efficient alternative for solving the problems identified, as well as the importance of taking memory into account, for a better linearization, when designing a predistorter.

Índice

Agradecimientos	ix
Resumen	xi
Abstract	xiii
Índice	xv
Índice de acrónimos	xvii
1 Introducción	1
1.1 <i>Motivación</i>	1
1.2 <i>Objetivos</i>	3
1.3 <i>Estructura de la memoria</i>	3
2 Predistorsión Digital mediante Modelos de Comportamiento	5
2.1 <i>Generalidades</i>	5
2.1.1 <i>Linealización a nivel de circuito y a nivel de sistema</i>	5
2.2 <i>Modelos de Volterra</i>	11
2.2.1 <i>Modelo sin memoria (ML)</i>	12
2.2.2 <i>Memory Polynomial (MP)</i>	12
2.3 <i>Métodos de aprendizaje para predistorsionadores</i>	13
2.3.1 <i>Predistorsión con aprendizaje directo</i>	13
2.3.2 <i>Predistorsión con aprendizaje indirecto</i>	14
3 Construcción de la Solución	17
3.1 <i>Procedimientos existentes</i>	17
3.1.1 <i>Algoritmo de Newton amortiguado</i>	21
3.2 <i>Problema del orden, complejidad y mal condicionamiento</i>	22
3.3 <i>Métodos considerados</i>	23
3.3.1 <i>Cálculo de la inversa por bloques</i>	23
3.3.2 <i>Aproximación de la inversa por una matriz de Covarianza</i>	24
3.4 <i>Figuras de mérito</i>	25
3.4.1 <i>EVM (Error Vector Magnitude)</i>	25
3.4.2 <i>ACPR (Adjacent Channel Power Ratio)</i>	26
3.4.3 <i>NMSE (Normalised Mean Square Error)</i>	26
4 Materiales Utilizados	29
4.1 <i>Plataforma WebLab</i>	29
4.2 <i>Matlab</i>	31
4.3 <i>Consideraciones sobre el código generado</i>	32
4.3.1 <i>Señal de entrada</i>	32
4.3.2 <i>Descripción del código generado</i>	34
5 Resultados	37
5.1 <i>Caso sin DPD</i>	37
5.2 <i>ML con algoritmo LS</i>	41
5.3 <i>MP con algoritmo LS</i>	44

5.4	<i>ML con algoritmo de cálculo de la inversa por bloques</i>	46
5.5	<i>MP con algoritmo de cálculo de la inversa por bloques</i>	49
5.6	<i>ML con algoritmo de aproximación de la inversa por una matriz de covarianza</i>	51
5.7	<i>MP con algoritmo de aproximación de la inversa por una matriz de covarianza</i>	54
6	Conclusiones y líneas futuras de investigación	59
6.1	<i>Conclusiones</i>	59
6.2	<i>Líneas futuras de investigación</i>	60
	Referencias	61

Índice de acrónimos

5G	Quinta generación de telefonía móvil
ACPR	Relación de potencia en el canal adyacente
BB	Banda base
CALLUM	Combined Analog-Locked Loop Universal Modulator
CV	Algoritmo de aproximación de la inversa por una matriz de covarianza
DPD	Digital Predistortion
DSP	Procesador digital de señal
DUT	Dispositivo bajo prueba
DVB	Digital Video Broadcasting
EE&R	Envelope Elimination and Restoration
EVM	Error Vector Magnitude
FPGA	Field Programmable Gate Array
IB	Algoritmo de cálculo de la inversa por bloques
ICA	Interferencia en el canal adyacente
IF	Frecuencia intermedia
IMS	International Microwave Symposium
LINC	Linear Amplification Using Nonlinear Components
LMS	Least Mean Square
LS	Least Square
ML	Memoryless
MP	Memory Polynomial
NMSE	Normalised Mean Square Error
OFDM	Multiplexación por división de frecuencias ortogonales
PA	Amplificador de potencia
PAPR	Potencia de pico a potencia promedio
QAM	Modulación de amplitud en cuadratura
RF	Radio frecuencia
RMS	Root Mean Square
VST	Transceptor de señal vectorial

1 INTRODUCCIÓN

1.1 Motivación

El desarrollo actual de las telecomunicaciones sigue un ritmo implacable, hasta el punto de que la quinta generación (5G) está a punto de desplegarse para su explotación. Este vertiginoso crecimiento exige tasas de transmisión cada vez más altas, y aunque inicialmente sólo se abarcaba el servicio de voz, las nuevas tendencias convergen hacia la prestación de una gran variedad de servicios multimedia como por ejemplo la transmisión de vídeo de alta definición.

Para lograr el cometido anteriormente expuesto las generaciones más avanzadas consideran nuevas modulaciones basadas en la utilización de envolvente no constante en detrimento de las tradicionales modulaciones de envolvente constante que ofrecen una menor tasa de transmisión. Se exploran nuevas bandas de frecuencias y se procede a la reutilización de éstas debido al aumento de la densidad de usuarios. Además, esto último conlleva a recurrir a esquemas de modulación de alta eficiencia espectral, que implica soportar variaciones tanto en la amplitud como en la fase de la señal transmitida. También se desarrollan nuevas técnicas de procesado de la señal con el mismo objetivo anterior: aprovechar eficientemente el espectro radioeléctrico.

Dentro de los dispositivos que permiten el procesado eficiente de la señal destaca el amplificador de potencia (PA, del inglés Power Amplifier), el cual presenta un comportamiento no lineal y está presente en todos los transmisores actuales. Los amplificadores de potencia, en particular los de la etapa de RF (radiofrecuencia), suelen trabajar próximos a su zona de saturación, es decir, en la región no lineal de su característica entrada-salida, ya que se busca operar con la mayor eficiencia posible y es conocido que la eficiencia de los amplificadores está enfrentada a su linealidad. Para una mayor eficiencia, se ha comenzado a sustituir de manera gradual en algunos sistemas los tradicionales amplificadores de potencia por otras estructuras más eficientes, como el Doherty o el seguidor de envolvente (envelope tracking).

La búsqueda de una mayor eficiencia obliga al PA a trabajar próximo a su zona no lineal, constituyendo un reto actual el desarrollo de amplificadores de potencia de RF que cumplan una relación de compromiso entre eficiencia y linealidad, pudiendo de este modo satisfacer los requisitos de los nuevos estándares de comunicación.

En comunicaciones móviles el espectro de RF se encuentra dividido en canales con un determinado ancho de banda, estos canales son asignados a los usuarios durante la comunicación. El carácter no lineal de los amplificadores de potencia provoca un ensanchamiento o recrecimiento en el espectro de la señal transmitida, y debido a este ensanchamiento un canal puede interferir con su adyacente. Este fenómeno se conoce como interferencia en canal adyacente (ICA) y es un efecto que se debe controlar para que su impacto quede por debajo de unos umbrales establecidos para el correcto funcionamiento de la comunicación. Otro factor importante en los sistemas de comunicaciones modernos es la eficiencia en potencia, en particular en sistemas donde la autonomía es crucial como es el caso de los equipos móviles y satélites.

El reto ante el que nos encontramos es intentar conciliar alta linealidad con alta eficiencia en potencia, lo cual se antoja complicado. Es aquí donde surge la posibilidad de aplicar técnicas de linealización de los amplificadores de potencia, que permiten la utilización conjunta de modulaciones de envolvente no constante y amplificadores

de alta eficiencia a costa de aumentar la complejidad del transmisor. Algunas de estas técnicas son feedforward, predistorsión, linealización mediante componentes no lineales LINC o EER. La incorporación de mecanismos adaptativos en las técnicas de linealización se ha visto facilitada por el auge tecnológico de los procesadores digitales de señal (DSP).

Para poder aplicar las técnicas de linealización es necesario conocer el comportamiento del PA que se trata de linealizar. Con el fin de poder realizar simulaciones que permitan prescindir del dispositivo físico, y gracias a los avances tanto en la capacidad de procesamiento de los procesadores actuales como en la velocidad y capacidad de las memorias RAM, se pueden desarrollar modelos de comportamiento que reproduzcan fielmente el funcionamiento del dispositivo real.

Es imprescindible incluir en el modelo que caracteriza el comportamiento del PA los efectos no lineales que aparecen cuando se aumenta la potencia de éste y los asociados a la elevación de la temperatura del dispositivo [1]. Se distinguen dos tipos de modelos para amplificadores de potencia: basados en un circuito equivalente y modelos de comportamiento. Los primeros buscan establecer una relación entre las tensiones y corrientes de una estructura circuital mediante un conjunto de ecuaciones no lineales. Estos son apropiados para la simulación a nivel de circuito. Cuando se desea una simulación a nivel de sistema completo, aparecen los basados en modelos de comportamiento. Estos tratan al PA como una “caja negra” donde es indiferente la estructura interna de éste; en ellos, el comportamiento está caracterizado por un conjunto de ecuaciones obtenidas mediante observaciones entrada-salida [2].

Los modelos de comportamiento se pueden clasificar en modelos sin memoria o modelos con memoria, dependiendo si tienen en cuenta la dependencia en la función de transferencia del amplificador de muestras correspondientes a instantes pasados de la señal de entrada. Los efectos de memoria causan una distorsión tanto en frecuencia como en amplitud, haciendo que las características AM/AM y AM/PM dejen de ser funciones estáticas y provocando asimetrías entre los niveles de los canales adyacentes al principal. De entre los modelos que tienen memoria, destacan por su generalidad en la caracterización de las no linealidades, los modelos derivados de las series de Volterra.

La técnica que está siendo más estudiada y la seleccionada para linealizar el PA analizado en este proyecto es la predistorsión digital (DPD, del inglés Digital PreDistortion). La exigencia de una mayor tasa de transmisión y una mayor eficiencia provoca que el amplificador presente un comportamiento no lineal más fuerte, y esto se traduce en la necesidad de acudir a DPDs basados en modelos de comportamiento cada vez más complejos. Esta complejidad se ve reflejada en un aumento del número de coeficientes del modelo y un empeoramiento en el número de condición de la matriz de medición. El crecimiento del número de coeficientes del modelo hace que la complejidad computacional aumente y los problemas de condicionamiento de las matrices implica que la precisión de la estima se vuelve más sensible a los errores numéricos introducidos en el proceso de cálculo de los coeficientes.

Para el cálculo de los coeficientes del modelo es necesario realizar la pseudoinversa de la matriz de medición. La operación de inversión supone una gran carga computacional, consumiendo importantes recursos. A lo largo de este proyecto se plantean distintos enfoques para abordar este problema. El primero de ellos consiste en utilizar la función de Matlab “pinv” para calcular la pseudoinversa de la matriz de medición. La función “pinv” computa una solución por mínimos cuadrados (solución óptima) en un sistema de ecuaciones lineales que no posee solución única. Se compararán los resultados alcanzados con este método de referencia con otros procedimientos que mostraremos en el tercer capítulo de este proyecto que permiten calcular los coeficientes del predistorsionador sin la necesidad de realizar inversión de matrices.

1.2 Objetivos

El objetivo del presente proyecto es la linealización de un PA mediante DPD. Este amplificador pertenece a la plataforma RF WebLab gestionada por la Universidad Tecnológica de Chalmers (Suecia). El acceso a esta plataforma se realiza de manera remota mediante una serie de funciones implementadas en Matlab, y es común a todos los usuarios que deseen utilizarlo. Para conformar el DPD se implementarán dos modelos. Un primer modelo más sencillo que no incluirá los efectos de memoria, denominado Memoryless (ML) y en segundo lugar un modelo que sí tendrá en cuenta los efectos de memoria, denominado Memory Polynomial (MP). Ambos serán comparados para poner de manifiesto las diferencias que conlleva el hecho de tener o no en cuenta los efectos de memoria en parámetros como el ACPR (Adjacent Channel Power Ratio) o la diferencia entre los espectros de la señal a la salida del amplificador previo paso por cada uno de los predistorsionadores.

Además, se realizará una segunda comparación entre los diferentes métodos utilizados para el cálculo de los coeficientes del predistorsionador. Se mostrarán tres propuestas. La primera de ellas utiliza la función “pinv” de Matlab que implementa un método robusto que busca la solución por mínimos cuadrados en un sistema de ecuaciones lineales que no posee solución única. La segunda propuesta es la expuesta en [3], donde los coeficientes se calculan mediante una estimación modificada de mínimos cuadrados en la cuál no hay realizar operaciones de inversión de matriz. La tercera y última propuesta es la expuesta en [4], en la cual se sustituye el cálculo de la pseudoinversa por la matriz de covarianza. Se evaluarán términos como la precisión en el cálculo de los coeficientes, la rapidez o la carga computacional de cada uno de los métodos.

1.3 Estructura de la memoria

Este documento se estructura como sigue:

- **Capítulo 1:** Introducción. En él se expone la motivación de este proyecto y se establece la línea de trabajo que se pretende seguir.
- **Capítulo 2:** Predistorsión digital mediante modelos de comportamiento. Se inicia con una introducción al concepto de predistorsión y se revisan los modelos propuestos basados en series de Volterra, haciendo especial hincapié en dos casos particulares de dichas series como son el Memoryless y el Memory Polynomial (MP). Para finalizar este capítulo se exponen las diferencias entre predistorsión con aprendizaje indirecto y predistorsión con aprendizaje directo.
- **Capítulo 3:** Construcción de la solución. Se exponen procedimientos llevados a cabo en el cálculo de los coeficientes del modelo y comparación entre ellos. Además del método de referencia basado en el cálculo de la pseudoinversa en Matlab mediante la función “pinv” se analizan dos alternativas. En el primer caso se utiliza una estimación modificada de mínimos cuadrados y en el segundo se hace uso de las propiedades de la matriz de covarianza. Ambos buscan el ahorro computacional que supone la inversión de matrices y la eliminación de errores numéricos.
- **Capítulo 4:** Materiales utilizados. Se presenta la plataforma de medidas utilizada en los experimentos realizados. Además, se resaltan algunas consideraciones sobre el código generado.
- **Capítulo 5:** Resultados. Se presentan las medidas y resultados obtenidos, así como su discusión. Como señal de entrada se utiliza una modulación OFDM con 512 subportadoras. Esta entrada es común a los distintos modelos de predistorsión analizados, comprobándose de este modo las diferencias entre uno y otro.
- **Capítulo 6:** Conclusiones y líneas futuras de investigación.

2 PREDISTORSIÓN DIGITAL MEDIANTE MODELOS DE COMPORTAMIENTO

2.1 Generalidades

Como se ha comentado en el capítulo anterior, los amplificadores de potencia son los elementos que mayor potencia consumen en los sistemas de RF (entorno al 70% del total), y de ahí la importancia de su eficiencia. Las modulaciones actuales presentan envolventes no constantes, con valores altos de relación de potencia de pico a potencia promedio (PAPR), lo que dificulta el aprovechamiento del margen dinámico de estos. Es en este ámbito donde toman importancia las técnicas de linealización, ya que gracias a ellas se puede extender el rango en el que el comportamiento del PA es lineal, sin que se vea mermada su eficiencia al mantener el punto de trabajo en la zona de saturación.

A continuación, se exponen las diferencias entre la linealización a nivel de circuito y a nivel de sistema, presentando algunas técnicas basadas en el segundo tipo.

2.1.1 Linealización a nivel de circuito y a nivel de sistema

Atendiendo al compromiso linealidad-eficiencia, existen diversas técnicas que permiten la linealización en el proceso de amplificación sin degradar en exceso la eficiencia del amplificador de potencia.

La linealización puede llevarse a cabo a dos niveles: circuito y sistema. Este Trabajo Fin de Grado se centra en la segunda, donde consideramos el amplificador como un modelo de “caja negra”, siendo indiferente su estructura interna. Estos modelos permiten describir fenómenos donde no se conocen todos procesos físicos involucrados. A nivel circuital, los sistemas de comunicaciones modernos e inalámbricos son a menudo demasiado complejos, dependiendo de un gran número de componentes y parámetros, lo que supone un serio problema tanto en tiempo de simulación como en memoria. Como alternativa, a menudo se reemplaza la descripción circuital de los subsistemas, como en el caso del PA, por una representación simplificada pero suficientemente precisa, denominada modelo de comportamiento. En este tipo de modelo a nivel de sistema, como hemos dicho anteriormente, los dispositivos modelados se consideran como una caja negra, obteniéndose la información del modelo a partir de las respuestas externas del dispositivo. Los modelos de comportamiento permiten la simulación completa a un nivel de abstracción superior, pero conservando la precisión en la representación del funcionamiento. A su vez, estos modelos de comportamiento proporcionan una forma de proteger la propiedad intelectual asociada a un circuito o bloque funcional, al encapsular las características del mismo. Además, ofrecen la posibilidad de diseñar un sistema incluso antes de tener disponibles los componentes hardware del mismo. La linealización a nivel de sistema permite reducir los efectos del comportamiento no lineal. A continuación, se presentan algunas de las técnicas de linealización a nivel de sistema existentes.

2.1.1.1 Feedback

En la figura 2-1 se muestra el esquema clásico de retroalimentación o feedback:

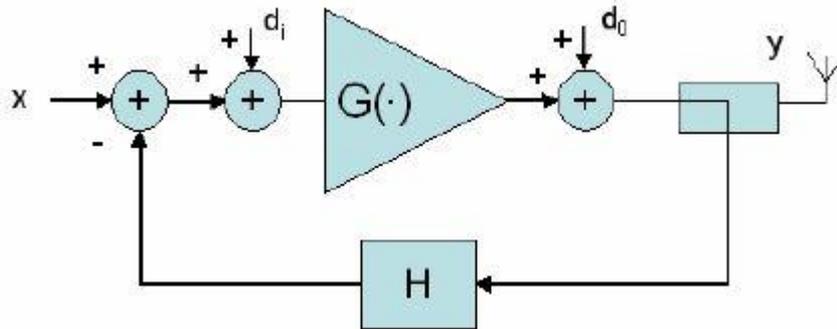


Figura 2-1. Esquema Feedback [5].

Siguiendo a [5], las señales d_i e d_o son perturbaciones que se suman a la entrada y salida del amplificador. La función de transferencia del bucle cerrado se considera lineal y la salida del sistema se obtiene como sigue:

$$y = \frac{G}{1 + G \cdot H} \cdot x + \frac{G}{1 + G \cdot H} \cdot d_i + \frac{1}{1 + G \cdot H} \cdot d_o \quad (2-1)$$

donde G es la función de transferencia del amplificador y H es la función de transferencia del lazo de retroalimentación. Observando la función de salida se aprecia que el efecto de la perturbación de entrada d_i puede reducirse aumentando el valor de H , mientras que para reducir la perturbación de salida d_o podemos aumentar H o G . Si únicamente se considera la parte correspondiente a la entrada x , se puede definir la función de transferencia del conjunto en frecuencia para las condiciones de estabilidad de la siguiente manera:

$$W = \frac{Y}{X} = \frac{G}{1 + G \cdot H} \quad (2-2)$$

La función W será inestable siempre que $G \cdot H = -1$, o expresado en términos complejos, que el módulo del producto sea unitario y su fase $\pm 180^\circ$. Este producto representa la función de transferencia en bucle abierto y define las condiciones de estabilidad del sistema. Una vez analizado el concepto de feedback, éste es utilizado en la linealización de amplificadores de la siguiente manera: las señales de entrada y salida son detectadas y filtradas pasobajo. Las señales resultantes en banda base son comparadas, creando una señal de error V_e , que es utilizada para controlar la ganancia del amplificador mediante un atenuador de voltaje variable. Este control de la ganancia permite modificar las características del amplificador teniendo como objetivo reducir la distorsión. La figura 2-2 aclara lo explicado anteriormente. Este esquema se aplica en comunicaciones de banda estrecha ya que presenta dificultades para responder a cambios en caso de utilizar envolventes con anchos de banda de varios megahercios.

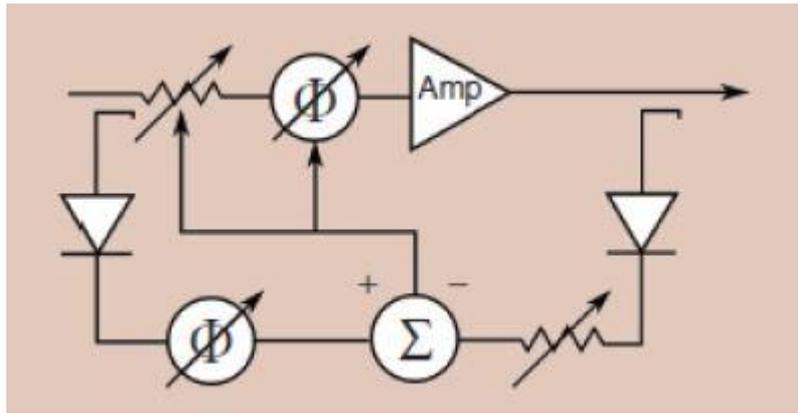


Figura 2-2. Esquema del linealizador feedback [6].

2.1.1.2 Feedforward

La figura 2-3 representa el esquema típico del linealizador feedforward.

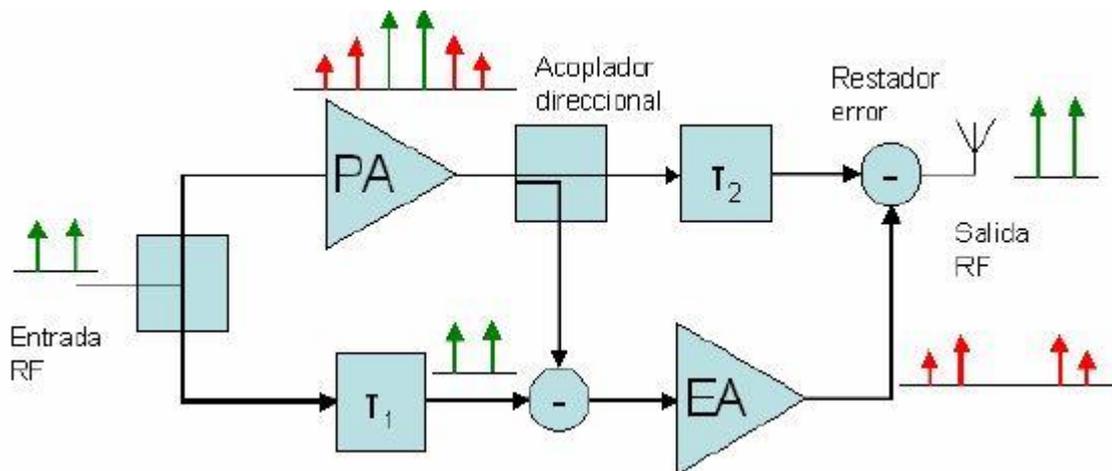


Figura 2-3. Esquema del linealizador Feedforward [5].

Como entrada se tiene una señal de dos tonos, que se bifurca en dos ramas, la superior y la inferior. En la rama superior la señal de entrada atraviesa el PA obteniéndose a su salida la señal amplificada más los productos de intermodulación y armónicos provocados por la no linealidad del amplificador. Una versión atenuada de la señal distorsionada por el amplificador se acopla a la rama inferior. En la rama inferior la señal de entrada ha sido retrasada un tiempo (T_1) equivalente al retardo inherente al amplificador, dicha señal se resta con la procedente de la rama superior, provocando que a la salida del restador se obtenga la distorsión provocada por el amplificador de potencia de la rama superior. Esta señal de distorsión es amplificada y pasada a la rama superior, mientras tanto en dicha rama superior la señal de salida del amplificador (señal amplificada + distorsión) es retrasada un retardo (T_2) equivalente al tiempo de operación del amplificador de la rama inferior. Ambas señales son restadas obteniéndose a la salida la señal inicial de dos tonos amplificada.

La técnica es estable, pero se debe tener en cuenta las tolerancias de los componentes, retardos, equilibrios entre los dos caminos, que deben ser compensados para evitar una mayor degradación de la señal. Existen versiones mejoradas de la estructura clásica [7].

2.1.1.3 LINC (Linear Amplification Using Nonlinear Components)

Este método consiste en separar la señal de entrada en dos señales de envolvente constante y modulada en fase que, tras pasar cada uno por sus correspondientes amplificadores de potencia no lineales y ser sumadas, dan como resultado la señal de salida lineal. La limitación de este método reside en el ancho de banda de las dos nuevas señales en las que dividimos la original, que son del orden de diez veces mayor que la primera. El esquema de este linealizador se muestra en la figura 2-4:

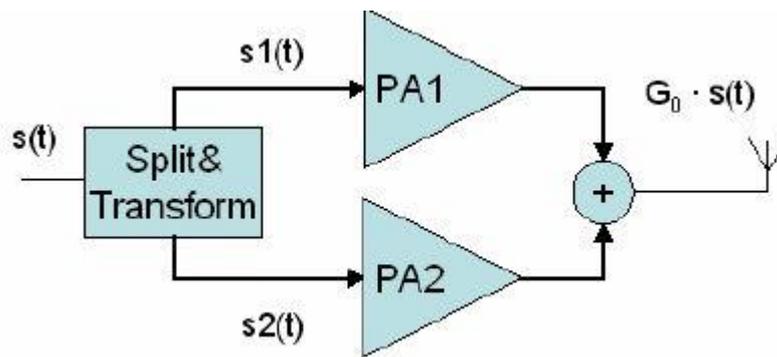


Figura 2-4. Esquema del linealizador LINC [5].

2.1.1.4 CALLUM (Combined Analog-Locked Loop Universal Modulator)

Se trata de una modificación del método LINC que consiste en dividir la señal de entrada en sus componentes de fase I y cuadratura Q para ser comparadas con las componentes de fase y cuadratura retroalimentadas de la salida de los dos amplificadores de potencia del caso LINC. La señal de error de cada una de las componentes controla un oscilador controlado por tensión (VCO) cuya salida es la entrada de cada amplificador. Esta técnica presenta problemas de estabilidad, y de ahí que su uso quede limitado a aplicaciones de banda estrecha.

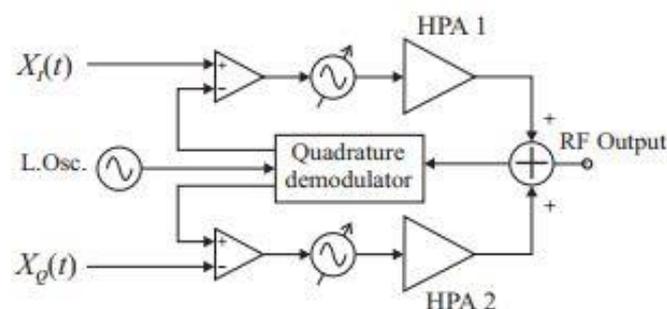


Figura 2-5. Esquema simplificado del linealizador CALLUM.

2.1.1.5 EE&R (Envelope Elimination and Restoration)

La señal modulada de entrada se divide en dos señales (figura 2-6): una primera que contiene la información de envolvente y la segunda que contiene la información de fase. La señal con la información correspondiente a la fase tiene amplitud constante y es la que se introduce en el PA, mientras que la señal que contiene la información

referente a la envolvente, que es de amplitud variable, modula la tensión de alimentación del amplificador. Esta técnica ha sido utilizada en sistemas OFDM y DVB, entre otros.

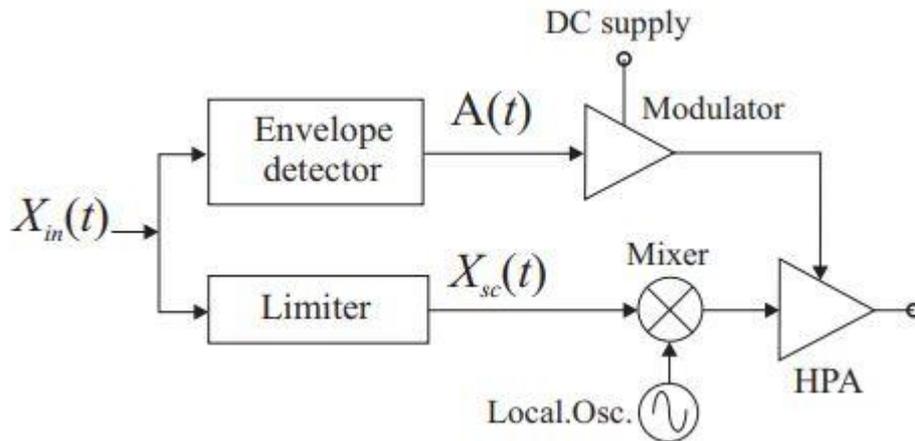


Figura 2-6. Esquema del linealizador EE&R [8].

2.1.1.6 Predistorsión digital

Es la técnica considerada en este Trabajo. Consiste en modificar la señal de entrada al amplificador, de forma que en su previo paso por el bloque denominado predistorsionador se modifica de tal manera que la señal a la salida de éste es la función inversa de la introducida por el PA (véase la figura 2-7). De esta manera, la salida del amplificador se aproxima a la característica ideal.

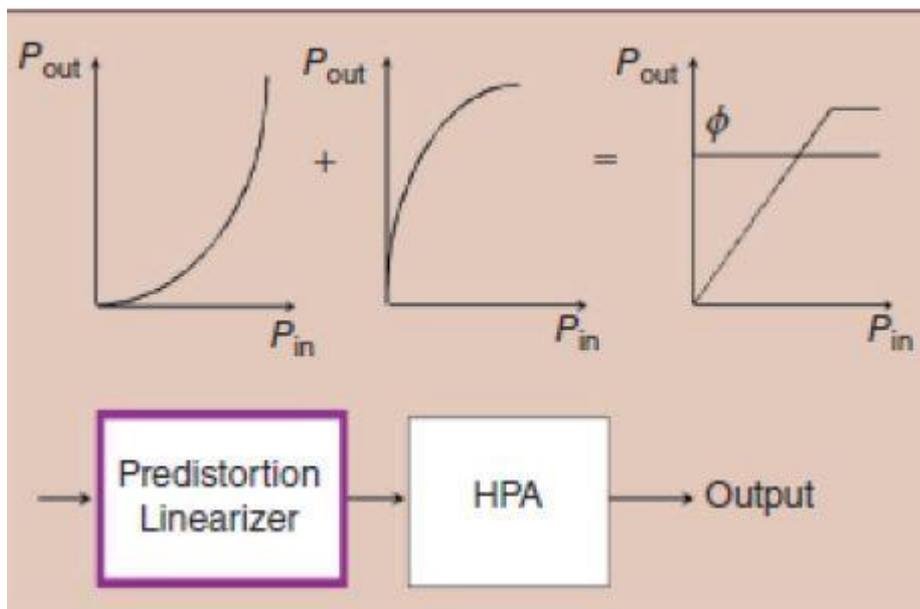


Figura 2-7. Linealización mediante predistorsión digital [6].

Las técnicas de predistorsión digital pueden aplicarse sobre la señal o los datos. La predistorsión aplicada sobre los datos es una técnica específica de cada tipo de modulación, y pretende compensar el espacio vectorial de los

datos o constelación. Los coeficientes del predistorsionador se optimizan minimizando el parámetro EVM (del inglés, Error Vector Magnitude). El gran inconveniente de este tipo de predistorsión radica en que no tiene en cuenta la distorsión fuera de banda, ya que actúa directamente sobre los datos y esto supone un problema para cumplir determinadas especificaciones de los estándares como por ejemplo los niveles de ICA. Otro de los problemas de esta técnica es que no es genérica sino específica para cada tipo de modulación.

Por su parte, la predistorsión digital aplicada sobre la señal genera una señal predistorsionada que, al pasar a través del PA, el cual produce sobre la señal la función inversa a la del predistorsionador, no debería tener distorsión a la salida, es decir, convertiría al amplificador de potencia en un dispositivo ideal lineal hasta el punto de saturación. Esta técnica, a diferencia de la anterior, tiene en cuenta tanto la distorsión en banda como fuera de ella.

La predistorsión de señal puede ser realizada en radiofrecuencia (RF), frecuencia intermedia (IF) o banda base (BB), siendo preferible los dos últimos casos. En IF o BB, los costes de los convertidores ADC o DCA son menores y además es más robusta la predistorsión. Su inconveniente es que en el paso a RF los mezcladores pueden introducir alguna forma de no linealidad.

Inicialmente, los predistorsionadores se diseñaban para compensar el comportamiento no lineal instantáneo del PA mediante predistorsión instantánea, es decir, no se tenían en cuenta los efectos de memoria. El comportamiento instantáneo queda determinado por las curvas estáticas AM/AM y AM/PM, debido a que la amplitud y la fase se consideraban funciones de los valores instantáneos de la señal de entrada.

Las modulaciones digitales multinivel utilizadas en los actuales estándares de comunicación requieren de un mayor ancho de banda en BB, y por lo tanto los modelos sin memoria no son adecuados para dicho propósito. Para este cometido se utilizan modelos que tengan en cuenta los efectos de memoria, y las curvas AM/AM y AM/PM presentan un comportamiento dinámico que pretende ser compensado por este tipo de modelos. Su valor se modifica dinámicamente con el tiempo y se pasa de tener una sola curva a un conjunto de éstas.

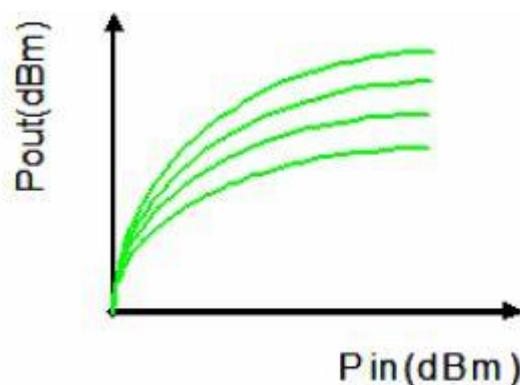


Figura 2-8. Curvas AM/AM considerando los efectos de memoria [7].

La forma más general de representar una función no lineal con memoria es una serie de Volterra, por lo que en este Trabajo se incidirá en el uso de predistorsionadores basados en este tipo de modelos, que se describen a continuación.

2.2 Modelos de Volterra

El matemático italiano Vito Volterra (1860-1940) basó sus estudios en el desarrollo de la rama de matemáticas conocida como análisis funcional. Sin embargo, la aplicación de dichas series en la ingeniería de sistemas se debe a Norbert Wiener, quien desarrolló la hipótesis de que un sistema no lineal invariante en el tiempo podía ser descrito mediante una serie de parámetros funcionales de Volterra, siempre y cuando el sistema a caracterizar no presentase no linealidades demasiado fuertes. Además de lo anterior, Wiener también propuso un método para identificar los coeficientes de las funciones de Volterra utilizando ruido blanco. En los años 60 la teoría de Volterra se aplicó a los circuitos no lineales y sistemas de comunicaciones. Sin embargo, a partir de los 70 su uso decayó debido a la aparición de dispositivos que conseguían mejoras en la linealidad de los circuitos y sistemas. La teoría de Volterra se redescubrió en los años 80 cuando la distorsión en los circuitos de comunicaciones comenzó a ser un problema de interés. La reaparición de esta teoría fue consecuencia del uso de dispositivos de bajo ruido y alta frecuencia que ofrecen unas buenas prestaciones pero no son muy lineales. Pero no es hasta la actualidad cuando las series de Volterra alcanzan una gran dimensión y ello se debe al reciente interés por obtener modelos de comportamiento de sistemas no lineales.

Las series de Volterra son similares a las series de Taylor, hasta el punto que se pueden ver como una serie de Taylor con memoria. Las series de Volterra deben ser convergentes, y para ello deben cumplir con algún criterio de suavidad. En la práctica, el número de términos necesarios para cumplir la condición de convergencia puede ser muy elevado.

La carga computacional requerida para el cálculo de los coeficientes de una serie de Volterra está ligada al grado del sistema polinomial que describe el sistema no lineal. Es decir, cuando el grado de éste aumenta también lo hace la carga computacional. Por este motivo, en la práctica suele truncarse el grado del sistema polinomial.

Los sistemas que se modelan utilizando las series de Volterra son cuasilineales, es decir, las no linealidades son débiles y por tanto una aproximación de bajo orden se considera apropiada.

Los modelos de Volterra llevan siendo estudiados durante mucho tiempo y poseen una amplia base teórica. Permiten modelar el comportamiento de sistemas no lineales con la particularidad de que las no linealidades de estos deben ser de carácter débil. Sin embargo, esto no representa una desventaja ya que los sistemas de comunicaciones actuales poseen un comportamiento no lineal a modelar que se encuentra situado en niveles bajos de la señal de distorsión. Lo que sí representa una desventaja para los métodos de Volterra es su complejidad computacional, pero como se ha visto anteriormente esta complejidad puede ser tratada mediante el truncamiento del grado del polinomio que describe la no linealidad.

Una serie de Volterra es la combinación de una convolución lineal y una serie de potencias no lineal que se utiliza para describir la relación entrada-salida de un sistema invariante en el tiempo causal no lineal con memoria desvaneciente. En el dominio del tiempo discreto, una serie de Volterra se expresa como:

$$y(l) = \sum_{p=1}^{\infty} \sum_{i_1=0}^{\infty} \cdots \sum_{i_p=1}^{\infty} h_p(i_1, \dots, i_p) \prod_{j=1}^p x(l - i_j) \quad (2-3)$$

Donde $x(l)$ e $y(l)$ representan la entrada y salida respectivamente, y a $h_p(i_1, \dots, i_p)$ se le denomina kernel o núcleo de Volterra de orden p . Un sistema de Volterra estará caracterizado mediante el conocimiento de sus

kernels. Las series de Volterra se truncan generalmente hasta un orden finito no lineal P y una memoria finita Q . Se conoce que la componente par de la intermodulación no interfiere con la banda de señal y los coeficientes de la serie de Volterra pueden suponerse simétricos sin pérdida de generalidad. Por dicho motivo, sólo los órdenes impares de la serie de Volterra han de ser considerados en una representación de BB.

Las series de Volterra se emplean para describir la relación entre la entrada y la salida de un amplificador con memoria de forma general. La ventaja de las series de Volterra es que la salida del modelo de Volterra es lineal con respecto a sus coeficientes, como ocurre con el modelo polinomial. Bajo el supuesto de estacionaridad, si calculamos los coeficientes con el criterio de media mínima o mínimo error cuadrático, tendremos un mínimo global único. Por lo tanto, es posible extraer el modelo no lineal de forma directa mediante el uso de algoritmos de regresión lineal, por ejemplo, mínimos cuadrados. Por otra parte, la desventaja de los sistemas que emplean series de Volterra reside en la complejidad computacional y en el gran número de parámetros que hay que calcular, siendo muchos de ellos innecesarios. Las líneas actuales de investigación sobre modelado basado en series de Volterra siguen apostando por modelos basados en coeficientes para determinar la no linealidad pero buscan omitir aquellos coeficientes que tienen una menor influencia en el comportamiento no lineal. De este modo se obtiene un modelo más sencillo que requiere de una menor carga computacional.

A continuación, se muestran dos simplificaciones para las series de Volterra en BB, el modelo sin memoria (Memoryless) y el Memory Polynomial (MP).

2.2.1 Modelo sin memoria (ML)

Se caracteriza porque la salida únicamente depende de instante actual (muestra actual), es decir, no se consideran muestras anteriores. La implementación de este modelo viene descrita por la siguiente ecuación:

$$y(l) = \sum_{k=0}^K a_k \cdot x(l) \cdot |x(l)|^{2k} \quad (2-4)$$

Donde $x(l)$ e $y(l)$ representan la entrada y salida del sistema, a_k los coeficientes del modelo y P representa el orden de la no linealidad expresado como $P = 2 \cdot K + 1$.

2.2.2 Memory Polynomial (MP)

También conocido como modelo de Volterra diagonal, introduce pares de muestras retrasadas respecto a la entrada hasta un orden P para describir los efectos no lineales y de memoria. La profundidad de la memoria viene determinada por el valor de Q . El modelo queda como sigue:

$$y_{MP} = \sum_{m=0}^Q \sum_{k=0}^K a_{m,k} \cdot x(l - \tau_m) \cdot |x(l - \tau_m)|^{2k} \quad (2-5)$$

Donde $a_{m,k}$ son los coeficientes del modelo. P , como se indicó anteriormente, es el orden de la no linealidad y viene descrito por la siguiente ecuación $P = 2 \cdot K + 1$. El retraso del modelo se asume como $\tau_m = m \cdot \tau_0$ donde τ_0 es el periodo de muestreo de la señal de entrada. Este modelo puede implementarse mediante el uso de un grupo de bancos de filtros. El hecho de que todos los términos fuera de la diagonal de la serie de Volterra se igualen a cero, justifica que se le denomine modelo de Volterra diagonal. Este modelo tiene sus ventajas e inconvenientes. Por un lado, reduce enormemente la complejidad de la serie de Volterra completa, lo cual es positivo, pero por ende disminuye la fidelidad del mismo ya que en algunos casos los términos fuera de la diagonal, llamados términos cruzados, pueden jugar un papel importante en el comportamiento del amplificador.

2.3 Métodos de aprendizaje para predistorsionadores

Una vez modelado el comportamiento del PA puede procederse a la construcción del predistorsionador, y para ello se consideran dos configuraciones básicas para la estimación de los coeficientes del mismo.

2.3.1 Predistorsión con aprendizaje directo

El funcionamiento del DPD mediante el método directo viene descrito por el diagrama de bloques de la figura 2-9.

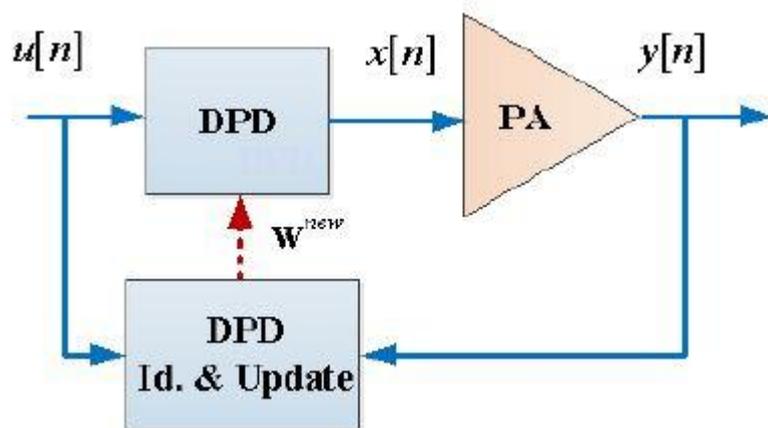


Figura 2-9. Esquema de predistorsión digital con aprendizaje directo [9].

La predistorsión realizada en el DPD conforme a la notación del diagrama de bloques (figura 2-9) pero utilizando como variable de tiempo discreto l en lugar de n , se puede expresar como:

$$x[l] = u[l] - \mathbf{U} \cdot \mathbf{w} \quad (2-6)$$

Donde \mathbf{w} es el vector de coeficientes del modelo y \mathbf{U} es la matriz de regresores o bases del modelo de comportamiento que se esté utilizando. El bloque de adaptación (denotado por DPD Id. & Update en la figura 2-9) es el que se encarga de la identificación y actualización de los coeficientes del DPD. La extracción de estos se realiza de manera iterativa mediante algoritmos adaptativos tipo LMS, donde en cada iteración i se intenta minimizar el error residual, siguiendo la ecuación (2-7):

$$\mathbf{w}^{i+1} = \mathbf{w}^i + \mu \Delta \mathbf{w} \quad (2-7)$$

Donde μ ($0 \leq \mu \leq 1$) es un factor de peso y $\Delta \mathbf{w}$ se obtiene como

$$\Delta \mathbf{w} = (\mathbf{U}^H \cdot \mathbf{U})^{-1} \cdot \mathbf{U}^H \cdot \mathbf{e} \quad (2-8)$$

donde \mathbf{U} es la matriz de regresores y \mathbf{e} el vector de error que se define (siguiendo la notación de la figura 2-9) como

$$e[l] = \frac{y[l]}{G_0} - u[l] \quad (2-9)$$

Siendo G_0 la ganancia lineal deseada del PA.

El método directo es muy similar al indirecto, con la diferencia de que en el primero se busca que los coeficientes converjan hacia su valor más óptimo de manera iterativa.

2.3.2 Predistorsión con aprendizaje indirecto

Es el método seguido para la construcción de los distintos predistorsionadores presentados en este Trabajo. Con este método, se hace una estimación del modelo inverso del amplificador de potencia a partir de los datos de entrada y de salida. Cuando el proceso de aprendizaje finaliza, los coeficientes del post-distorsionador se copian idénticamente en el modelo del predistorsionador en la entrada.

Con esta configuración no es necesario calcular el modelo del amplificador de potencia, sino que basta con disponer de su entrada y salida para estimar el modelo del post-distorsionador. La ecuación (2-10) muestra dicho funcionamiento:

$$F_{post}(G(x)) = G_0 \cdot x \quad (2-10)$$

En la figura 2-10 se muestra un esquema del método de aprendizaje indirecto:

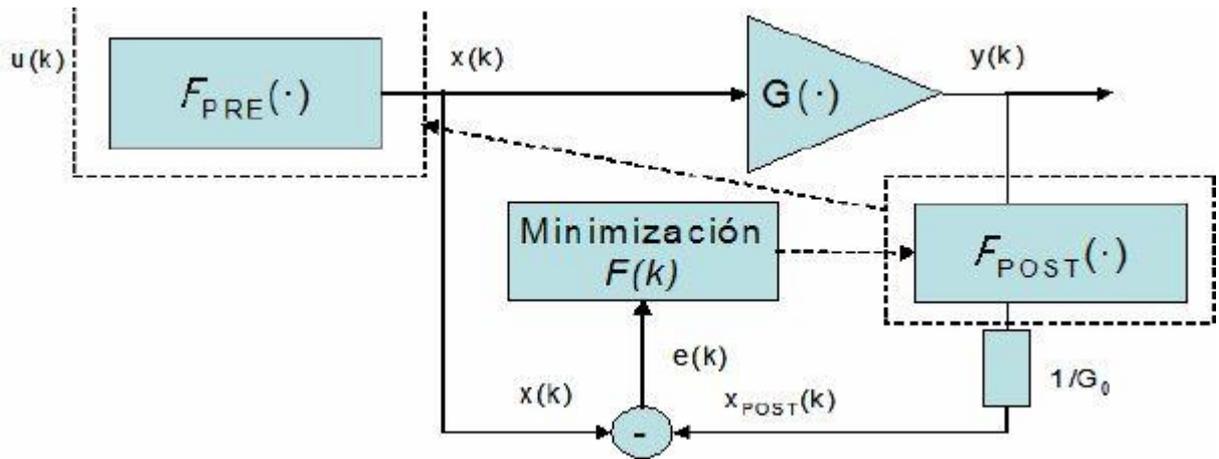


Figura 2-10. Esquema de predistorsionador con aprendizaje indirecto [5].

3 CONSTRUCCIÓN DE LA SOLUCIÓN

3.1 Procedimientos existentes

Este capítulo se centra en el cálculo de los coeficientes del predistorsionador, mostrando algunos de los procedimientos existentes para ello. Consideraremos el modelo Memory Polynomial (MP) en la configuración de método de aprendizaje indirecto. Como se indicó en el capítulo anterior, este modelo es una serie de Volterra truncada y consiste en una serie de retrasos y funciones base (regresores). Sólo tiene en cuenta los términos diagonales de los kernels, de este modo se consigue una reducción de la complejidad computacional en comparación con la serie de Volterra completa. El modelo MP viene descrito por la ecuación (3-1):

$$y_{MP}(l) = \sum_{m=0}^Q \sum_{k=0}^K a_{m,k} \cdot x(l - \tau_m) \cdot |x(l - \tau_m)|^{2k} \quad (3-1)$$

donde $x(l)$ es la envolvente compleja discreta de la señal de entrada e $y(l)$ es la envolvente compleja discreta de la función de salida. La señal de salida es una combinación de Q muestras de la señal de entrada retrasadas. Este modelo sólo considera los términos de orden impar y puede representarse mediante la estructura de bloques mostrada en la figura 3-1:

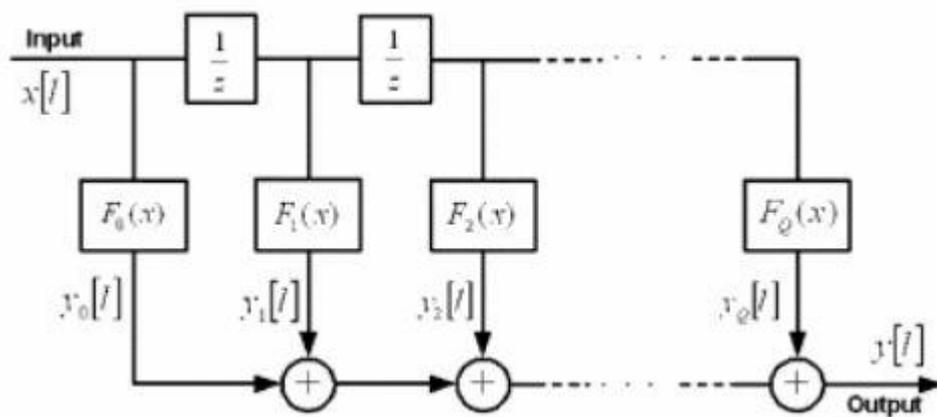


Figura 3-1. Estructura de bloques del Memory Polynomial.

Los bloques $F_q(x)$ representan funciones no lineales sin memoria de orden impar, expresadas mediante la siguiente ecuación:

$$F_q(x) = \sum_{k=0}^K a_{k,q} \cdot |x|^{2k} \cdot x \quad (3-2)$$

Para poder obtener el valor de los coeficientes complejos que modelan el predistorsionador o el comportamiento del PA, si la configuración utilizada es la del método de aprendizaje directo, debemos expresar la ecuación (3-1) en forma matricial. A partir de los datos medidos de la señal de entrada y de salida definimos:

$$\mathbf{Y} = [y(l) \ y(l+1) \ \dots \ y(l+M-1)]^T \quad (3-3)$$

Donde M es el número de muestras escogido para modelar de la señal de salida $y(l)$, l es el índice de la primera muestra. El vector representa un fragmento de la señal de salida completa $y(l)$. En el presente Trabajo, este vector no ha sido seleccionado de manera aleatoria sino que se ha buscado la muestra con mayor potencia, se ha centrado y se han seleccionado $N/2 + Q$ muestras anteriores y $N/2 - 1$ muestras posteriores a ella. El tamaño de este vector es variable dependiendo de la profundidad de la memoria utilizada. El valor de M viene dado por la siguiente expresión $M = N + Q$, donde Q es la profundidad de la memoria.

El número de muestras N escogido para modelar siempre es mucho mayor que el valor de K y Q , por lo tanto atendiendo a la ecuación (3-1) se puede observar que el sistema a resolver es un sistema sobredimensionado ya que el número de ecuaciones (igual al valor de N) es mucho mayor que el número de incógnitas (igual a $(K+1) \cdot (Q+1)$) que se corresponde con los coeficientes del predistorsionador o del modelo del PA.

Para modelar la señal de entrada $x(l)$ se toma una muestra de ésta siguiendo el mismo criterio que para el caso de la señal de salida.

Como el método de aprendizaje utilizado es el indirecto buscamos unos coeficientes que hagan que la señal de salida del amplificador de potencia se ajuste lo máximo posible a la señal de entrada de éste, con esto conseguiremos la función inversa del amplificador. Por este motivo trabajaremos con la siguiente ecuación en sustitución de la mostrada en (3-1):

$$x(l) = \sum_{m=0}^Q \sum_{k=0}^K a_{m,k} \cdot \frac{y}{G_c}(l - \tau_m) \cdot \left| \frac{y}{G_c}(l - \tau_m) \right|^{2k} \quad (3-4)$$

donde $y(l)$ sigue siendo la envolvente compleja discreta de la señal de salida del PA y $x(l)$ la envolvente compleja discreta de la señal de entrada del mismo, pero a la hora de estimar los coeficientes del predistorsionador, el cual tiene como objetivo obtener la salida inversa del amplificador, $y(l)$ será la entrada a este y $x(l)$ su salida, quedando la ecuación (3-4). El valor de G_c representa la ganancia objetivo del PA (típicamente ganancia comprimida), por simplicidad en la notación a partir de ahora denotaremos a y/G_c como y simplemente.

También definimos:

$$\mathbf{H} = [\mathbf{H}_0 \ \dots \ \mathbf{H}_Q]^T \quad (3-5)$$

Donde \mathbf{H}_0 representa el conjunto de regresores sin retraso, mientras que \mathbf{H}_Q representa al conjunto de regresores con el máximo retraso y viene definido por:

$$\mathbf{H}_Q = \begin{bmatrix} h_{Q,0}(l) & h_{Q,1}(l) & \dots & h_{Q,K}(l) \\ h_{Q,0}(l+1) & h_{Q,1}(l+1) & \dots & h_{Q,K}(l+1) \\ \vdots & \vdots & \ddots & \vdots \\ h_{Q,0}(l+N-1) & h_{Q,1}(l+N-1) & \dots & h_{Q,K}(l+N-1) \end{bmatrix} \quad (3-6)$$

y

$$h_{Q,K}(l) = |y(l-Q)|^{2K} \cdot y(l-Q) \quad (3-7)$$

definen las columnas, a las que nos referiremos también como regresores.

Los coeficientes complejos se pueden expresar como:

$$\mathbf{a} = [\mathbf{a}_0 \dots \mathbf{a}_Q]^T \quad (3-8)$$

donde

$$\mathbf{a}_Q = [a_{Q,0} \ a_{Q,1} \ \dots \ a_{Q,K}]^T \quad (3-9)$$

Por tanto, la ecuación matricial final a resolver viene dada por la siguiente expresión:

$$\mathbf{x} = \mathbf{H} \cdot \mathbf{a} \quad (3-10)$$

donde \mathbf{x} es un vector de tamaño $N \times 1$, \mathbf{H} es una matriz de tamaño $N \times (K+1) \cdot (Q+1)$ y \mathbf{a} es un vector $(K+1) \cdot (Q+1) \times 1$.

Una vez planteada la ecuación matricial procedemos al cálculo de los coeficientes complejos. Al tratarse de un sistema sobredimensionado como se indicó anteriormente, la matriz del sistema no es cuadrada, sino rectangular, y por tanto no existe la inversa de ésta. Por dicho motivo hay que estimar la solución con algún criterio, como el de mínimos cuadrados que es el presentado a continuación. Los coeficientes estimados $\hat{\mathbf{a}}$ se pueden obtener despejándolos de la ecuación (3-10), quedando la siguiente expresión:

$$\hat{\mathbf{a}} = \mathbf{H}^{-1} \cdot \mathbf{x} \quad (3-11)$$

donde \mathbf{H}^{-1} denota la pseudoinversa de la matriz \mathbf{H} . Si denominamos a la matriz \mathbf{H} como \mathbf{Y} la ecuación (3-11) también puede ser expresada de la siguiente forma:

$$\hat{\mathbf{a}} = (\mathbf{Y}^H \cdot \mathbf{Y})^{-1} \cdot \mathbf{Y}^H \cdot \mathbf{x} \quad (3-12)$$

donde $(\mathbf{Y}^H \cdot \mathbf{Y})^{-1} \cdot \mathbf{Y}^H$ es la pseudoinversa de \mathbf{Y} .

La solución presentada en (3-12) es la obtenida mediante el algoritmo Least Square (LS). Este algoritmo tiene como objetivo principal intentar encontrar los coeficientes \mathbf{a} del modelo que mejor se ajustan en términos de error cuadrático, para ello requiere de un modelo de un orden concreto, un vector de muestras de la señal de entrada $\mathbf{x} = [x(0), \dots, x(N)]^T$ y otro vector de muestras de la señal de salida como el de la ecuación (3-3). Asumimos que el número de muestras de entrada y salida, y por lo tanto de ecuaciones es mucho más grande que el número de regresores, por este motivo nos encontramos frente a un sistema sobredeterminado y no podemos hallar una solución que satisfaga todas las ecuaciones. Debido al hecho anteriormente señalado, nos centraremos en buscar la solución que minimiza el error cuadrático. A continuación, se muestra el desarrollo matemático para llegar a la solución (3-12).

Partiendo de la ecuación (3-10) particularizada para el caso del Memory Polynomial con configuración de método indirecto y renombrando \mathbf{H} como \mathbf{Y} tenemos que la salida estimada del predistorsionador para unos coeficientes \mathbf{a} determinados es:

$$\hat{\mathbf{x}} = \mathbf{Y} \cdot \mathbf{a} \quad (3-13)$$

donde \mathbf{Y} es la matriz de regresores donde $\mathbf{y}_n = (\varphi_0, \dots, \varphi_{(K+1) \cdot (Q+1)})$ es el vector de regresores del modelo inverso del amplificador ya que estamos trabajando con una configuración de método de aprendizaje indirecto.

Definimos el error cuadrático acumulado como,

$$\sum_{i=0}^N \|e(i)\|^2 = \|x(i) - \hat{x}(i)\|^2 \quad (3-14)$$

$$\sum_{i=0}^N \|e(i)\|^2 = \|x(i) - \mathbf{y}_i \cdot \mathbf{a}\|^2 \quad (3-15)$$

Podemos reescribir esta expresión en notación matricial como:

$$J(\mathbf{a}) = (\mathbf{x} - \mathbf{Y} \cdot \mathbf{a})^H \cdot (\mathbf{x} - \mathbf{Y} \cdot \mathbf{a}) \quad (3-16)$$

Donde $J(\mathbf{a})$, es la función que nos da el error cuadrático acumulado en función del vector de coeficientes \mathbf{a} . Nuestro objetivo ahora será encontrar los coeficientes \mathbf{a} que minimizan la función $J(\mathbf{a})$, para ello tendremos que calcular las derivadas parciales de dicha función con respecto a cada uno de los coeficientes del vector \mathbf{a} e igualarlas a 0, es decir:

$$\frac{\partial J(\mathbf{a})}{\partial a_j} = 0 \quad (3-17)$$

retomando la expresión matricial (3-16) queda:

$$J(\mathbf{a}) = (\mathbf{x} - \mathbf{Y} \cdot \mathbf{a}) \cdot (\mathbf{x} - \mathbf{Y} \cdot \mathbf{a})^H = \mathbf{x}^H \cdot \mathbf{x} - 2 \cdot \mathbf{x}^H \cdot \mathbf{Y} \cdot \mathbf{a} + \mathbf{a}^H \cdot \mathbf{Y}^H \cdot \mathbf{Y} \cdot \mathbf{a} \quad (3-18)$$

Derivando la expresión anterior obtenemos:

$$\frac{\partial J(\mathbf{a})}{\partial \mathbf{a}} = -2 \cdot \mathbf{Y}^H \cdot \mathbf{x} + 2 \cdot \mathbf{Y}^H \cdot \mathbf{Y} \cdot \mathbf{a} \quad (3-19)$$

Una vez hemos obtenido la expresión de la derivada obtendremos un mínimo cuando la igualemos a 0, los valores obtenidos en los coeficientes al cumplir esta condición serán los deseados:

$$-2 \cdot \mathbf{Y}^H \cdot \mathbf{x} + 2 \cdot \mathbf{Y}^H \cdot \mathbf{Y} \cdot \mathbf{a} = 0 \quad (3-20)$$

$$\mathbf{Y}^H \cdot \mathbf{Y} \cdot \mathbf{a} = \mathbf{Y}^H \cdot \mathbf{x} \quad (3-21)$$

$$\mathbf{a} = (\mathbf{Y}^H \cdot \mathbf{Y})^{-1} \cdot \mathbf{Y}^H \cdot \mathbf{x} \quad (3-22)$$

Finalmente llegamos a la ecuación (3-22) que es idéntica a la (3-12) y se corresponde con la solución de mínimos cuadrados.

3.1.1 Algoritmo de Newton amortiguado

Este algoritmo [10] es una variante del algoritmo Least Mean Square (LMS). Busca actualizar parcialmente los coeficientes del predistorsionador \mathbf{a} en base a una estimación previa, forzando así cierta continuidad. Viene descrito por la siguiente ecuación:

$$\mathbf{a}_{p+1} = \mathbf{a}_p + \mu \cdot (\mathbf{Y}^H \cdot \mathbf{Y})^{-1} \cdot \mathbf{Y}^H \cdot \mathbf{e} \quad (3-23)$$

donde p es el índice de bloque, μ es un constante de “relajación” y \mathbf{e} el error definido como la diferencia entre la señal deseada y la estimada. Si $\mu = 1$ y $\mathbf{a}_0 = 0$ entonces el algoritmo converge a la solución de Least-Square en una sola iteración. Cuando $\mu < 1$, se introduce algo de memoria que restringe el valor que \mathbf{a} puede cambiar de un bloque a otro. Esta forma tiene la ventaja de ser robusta ya que el error provoca la actualización de μ por lo tanto siempre obtenemos una solución que minimiza el error promedio.

3.2 Problema del orden, complejidad y mal condicionamiento

Centrándonos en el caso presentado en la sección anterior (modelo MP) y tomando como método de resolución para el cálculo de los coeficientes el ajuste LS, el problema se reduce a solventar la ecuación (3-22). Observamos que \mathbf{Y} depende directamente del modelo de comportamiento del PA. Los modelos de comportamiento son series infinitas que se truncan para acabar encontrando cierta solución, como por ejemplo el MP el cual analizamos en esta sección. Este modelo, presentado en la ecuación (3-4) depende de los parámetros K y Q y el número de términos que se obtienen es $N \cdot (K + 1) \cdot (Q + 1)$. Conforme aumenta la complejidad del modelo, mejor será la estimación que se puede alcanzar, y por lo tanto mejor será la linealización, mayor el ACPR que se puede llegar a obtener y más se puede minimizar el NMSE.

La idea presentada anteriormente puede llevar a pensar que cuanto mayor sea el orden del modelo (depende del grado del polinomio y profundidad de la memoria), mejor será la linealización. Sin embargo, un orden demasiado alto, puede provocar efectos negativos, ya que al aumentar el orden del modelo aumentamos el coste computacional.

Para el modelo MP analizado y unos vectores \mathbf{y} , \mathbf{x} de entrada y salida del DPD de tamaño $N \times 1$ respectivamente, tenemos una matriz \mathbf{Y} de tamaño $N \cdot (K + 1) \cdot (Q + 1)$. El tamaño de la matriz crece exponencialmente con el orden del modelo, y hay que tener en cuenta que esta matriz ha de ser invertida para poder llevar a cabo la identificación de los coeficientes. Este hecho puede suponer en la práctica un problema de coste computacional y de tiempo de operación.

Aparte del problema del coste computacional, el aumentar el orden del modelo también puede suponer la aparición de problemas de mal condicionamiento.

Aumentar el orden mejora la linealización hasta cierto punto. Cuando se incluyen demasiados coeficientes se comete un error de sobreajuste y el modelo pierde su capacidad de predicción.

Así, el aumento del orden puede provocar una degradación del NMSE hasta llegar a cierto punto donde empieza a estancarse aunque sigamos aumentándolo. Tras llegar a este punto, si seguimos aumentando el orden, el NMSE empezará a empeorar y aumentar de forma descontrolada, este hecho se debe al mal condicionamiento de la matriz \mathbf{Y} . El mal condicionamiento de la matriz es debido a que entre los regresores (funciones base) de ésta existe cierto grado de correlación, es decir, no son completamente ortogonales.

3.3 Métodos considerados

Una vez conocidos los problemas que acarrea el uso de un orden del modelo demasiado alto en la inversión de la matriz \mathbf{Y} , vamos a buscar la manera de encontrar un conjunto de coeficientes reducido que represente al DPD fielmente, sin necesidad de realizar inversión de matrices para su cálculo. Para ello en esta sección se describirán dos de los algoritmos que junto a la solución LS, que es el utilizado por la función `pinv` de matlab, son los utilizados en este proyecto para obtener los coeficientes de los predistorsionadores presentados.

3.3.1 Cálculo de la inversa por bloques

Este método ha sido propuesto [3].

La clave en la estimación de los coeficientes del predistorsionador es calcular el valor de la pseudoinversa de \mathbf{Y} la cual denotaremos como \mathbf{Y}^+ . Con el clásico ajuste de Least-Square (LS) es fácil obtener la pseudoinversa mediante la siguiente expresión $\mathbf{Y}^+ = (\mathbf{Y}^H \cdot \mathbf{Y})^{-1} \cdot \mathbf{Y}^H$. Como podemos observar en la expresión anterior se realiza la inversa a una matriz. Esta operación tiene una serie de desventajas como es el aumento en la cantidad de cálculo, consecuencia de esto aumenta la carga computacional y ello dificulta la implementación de este algoritmo en FPGAs. Además está el problema de que la matriz que se trata de invertir tiene un número de condición muy alto, por lo que los errores numéricos pueden propagarse. Por lo tanto, el algoritmo propuesto tiene como objetivo obtener la pseudoinversa de \mathbf{Y} , \mathbf{Y}^+ , sin realizar la operación de inversión de matrices.

Consideramos la siguiente situación: \mathbf{A} y \mathbf{C} son las submatrices de \mathbf{Y} , cuya relación viene dada por la ecuación (3-24). El problema en cuestión consiste en calcular $(\mathbf{C}^H \cdot \mathbf{C})^{-1}$ a partir de $(\mathbf{A}^H \cdot \mathbf{A})^{-1}$

$$\mathbf{C} = [\mathbf{A}, \mathbf{B}] \quad (3-24)$$

donde \mathbf{B} es una columna que se añadirá en cada iteración. Estas columnas añadidas en cada una de las iteraciones son los regresores del modelo y deben ser introducidos uno a uno. Utilizando las propiedades de la matriz de bloques, la relación entre $(\mathbf{A}^H \cdot \mathbf{A})^{-1}$ y $(\mathbf{C}^H \cdot \mathbf{C})^{-1}$ puede representarse matemáticamente de la siguiente manera:

$$(\mathbf{C}^H \cdot \mathbf{C})^{-1} = \begin{bmatrix} \mathbf{A}^H \cdot \mathbf{A} & \mathbf{A}^H \cdot \mathbf{B} \\ \mathbf{B}^H \cdot \mathbf{A} & \mathbf{B}^H \cdot \mathbf{B} \end{bmatrix}^{-1} = \begin{bmatrix} (\mathbf{A}^H \cdot \mathbf{A})^{-1} + \boldsymbol{\gamma} \cdot \boldsymbol{\sigma} \cdot \rho & -\boldsymbol{\gamma} \cdot \rho \\ -\boldsymbol{\sigma} \cdot \rho & \rho \end{bmatrix} \quad (3-25)$$

donde

$$\boldsymbol{\gamma} = (\mathbf{A}^H \cdot \mathbf{A})^{-1} \cdot \mathbf{A}^H \cdot \mathbf{B} \quad (3-26)$$

$$\boldsymbol{\sigma} = \mathbf{B}^H \cdot \mathbf{A} \cdot (\mathbf{A}^H \cdot \mathbf{A})^{-1} \quad (3-27)$$

$$\rho = (\mathbf{B}^H \cdot \mathbf{B} - \mathbf{B}^H \cdot \mathbf{A} \cdot \gamma)^{-1} \quad (3-28)$$

Teniendo en cuenta que \mathbf{B} es un vector columna, el producto $\mathbf{B}^H \cdot \mathbf{B}$ es un número. ρ es el recíproco de un número. Esto significa que solo hay operaciones de multiplicación de matrices en el proceso de cálculo de $(\mathbf{C}^H \cdot \mathbf{C})^{-1}$ sobre la base $(\mathbf{A}^H \cdot \mathbf{A})^{-1}$. Por tanto, se trata de un algoritmo iterativo, que itera entre (3-25), (3-26), (3-27) y (3-28) hasta que todas las inversiones necesarias se lleven a cabo.

Además, en la primera iteración la submatriz \mathbf{A} solo tiene una columna por lo que $(\mathbf{A}^H \cdot \mathbf{A})^{-1}$ puede ser obtenida fácilmente.

3.3.2 Aproximación de la inversa por una matriz de Covarianza

El método propuesto en el artículo [4] viene definido para una configuración de aprendizaje directo como su propio título indica, pero en este Trabajo se ha adaptado y a continuación se explica para una configuración con aprendizaje indirecto.

En general, la precisión del modelo de comportamiento de un PA o de un predistorsionador puede mejorar con el aumento de N , es decir, con el número de muestras seleccionadas de las señales de entrada y salida utilizadas en la confección del modelo. Esto se debe a que hay una cierta desviación de la distribución entre muestras finitas e infinitas.

El ajuste tradicional de Least-Square (LS) tiene que calcular el producto de \mathbf{Y}^H por \mathbf{Y} y posteriormente su inversa. Estas dos operaciones tienen una complejidad computacional muy alta consumiendo por tanto muchos recursos. Cuando N tiende a infinito, y teniendo en cuenta que la señal de entrada se asume estacionaria y ergódica, la matriz variante en el tiempo $\mathbf{Y}^H \cdot \mathbf{Y}/N$ se convierte en la matriz de covarianza $\Delta_q = \lim_{N \rightarrow \infty} \left(\frac{\mathbf{Y}^H \cdot \mathbf{Y}}{N} \right)$. Esta matriz de covarianza es una matriz constante y está formada por la covarianza de las funciones base (regresores). Por lo tanto, usando una matriz constante \mathbf{V}_q (inversa de la matriz de covarianza) como sustituto de la matriz $\lim_{N \rightarrow \infty} \left(\frac{\mathbf{Y}^H \cdot \mathbf{Y}}{N} \right)^{-1}$ e incrementando el número de muestras en el modelado (N), eliminamos las dos operaciones de mayor complejidad computacional. Aunque finalmente se tenga que hacer una operación de inversión para calcular la inversa de la matriz de covarianza, esta nueva operación conlleva una menor complejidad computacional ya que la matriz de covarianza es una matriz constante a diferencia del caso inicial donde la operación de inversión se le realizaba a una matriz variable en el tiempo. Este nuevo enfoque conlleva a que la ecuación de actualización de los coeficientes pase de ser:

$$\mathbf{a}_{s+1} = \mathbf{a}_s + \mu \cdot (\mathbf{Y}^H \cdot \mathbf{Y})^{-1} \cdot \mathbf{Y}^H \cdot \mathbf{x} \quad (3-29)$$

A ser la siguiente:

$$\mathbf{a}_{s+1} = \mathbf{a}_s + \frac{\mathbf{V}_q \cdot \mathbf{Y}^H \cdot \mathbf{x}}{N} \quad (3-30)$$

donde \mathbf{V}_q es la inversa de la matriz de covarianza.

3.4 Figuras de mérito

En este apartado se presentan algunas las figuras de mérito que se utilizaran para evaluar los distintos métodos que se van a comparar. Éstas son el EVM, el NMSE y el ACPR.

3.4.1 EVM (Error Vector Magnitude)

Este parámetro se determina a la salida del filtro RRC del demodulador mediante la comparación entre el valor de símbolo recibido y el valor esperado idealmente. Se puede definir como el error RMS (Root Mean Square) de la diferencia entre los valores de símbolos recibidos y los esperados.

La EVM se define en la ecuación (3-31), donde I y Q son las componentes en fase y cuadratura de los símbolos. ΔI es la diferencia entre la componente I de la señal recibida y la esperada. N es el número total de símbolos en la medida. S es la potencia media de los símbolos esperados o de referencia.

$$EVM = \sqrt{\frac{\frac{1}{N} \cdot \sum_{j=1}^N (\Delta I_j^2 + \Delta Q_j^2)}{S_{media}^2}} [\%] \quad (3-31)$$

El valor de EVM nos da una medida de cómo la no linealidad afecta al proceso de detección. Se suele expresar en porcentaje y normalizado para permitir la comparación entre las distintas modulaciones.

La figura 3-2 muestra gráficamente el parámetro EVM [11].

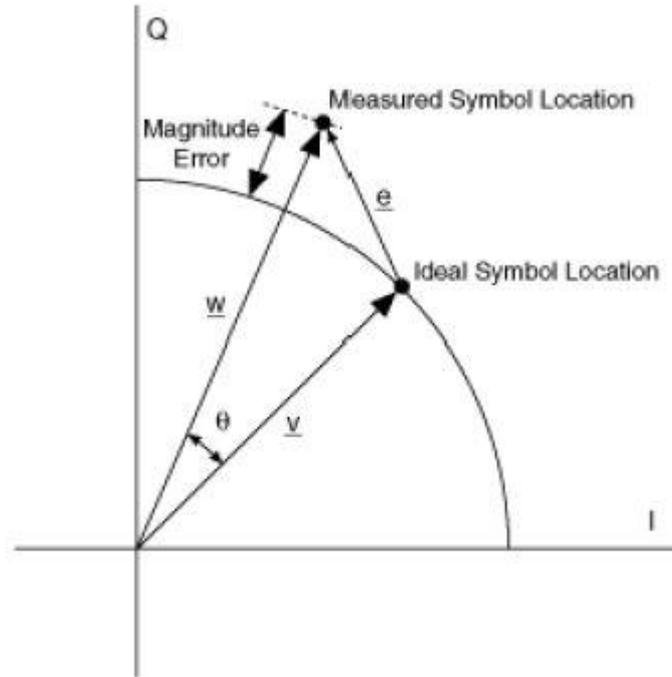


Figura 3-2. Representación gráfica del parámetro EVM.

3.4.2 ACPR (Adjacent Channel Power Ratio)

El ACPR nos da una medida de la potencia de la señal fuera de la banda, normalizada con respecto a la potencia de la señal dentro de la banda. Así pues, caracteriza cómo las no linealidades de la señal afectan a los canales adyacentes. La potencia en el canal adyacente debida a la señal en banda del canal considerado será vista como una señal interferente para la señal en banda del canal adyacente. El ACPR viene dado por la ecuación (3-32):

$$ACPR(dB) = 10 \cdot \log_{10} \sum_{k=1}^K \left| \frac{Y^{out-band}[k]}{Y^{in-band}[k]} \right|^2 \quad (3-32)$$

Donde, $Y^{out-band}$ es el espectro de la señal fuera de la banda e $Y^{in-band}$ es el espectro de la señal dentro de la banda. Cuando nos referimos al espectro, estamos hablando de la Transformada Discreta de Fourier (DFT).

3.4.3 NMSE (Normalised Mean Square Error)

El NMSE nos da una medida del parecido entre dos señales en el dominio del tiempo, normalizado por la potencia de la señal utilizada como referencia. En el contexto de linealización de amplificadores, se utilizará para valorar el parecido entre la salida normalizada del amplificador linealizado y su entrada. Su expresión viene dada por la ecuación (3-33):

$$NMSE(dB) = 10 \cdot \log_{10} \frac{1}{N} \cdot \sum_{n=1}^N \left| \frac{y[n] - x[n]}{x[n]} \right|^2 \quad (3-33)$$

donde, \mathbf{x} son las muestras de la señal a la entrada, \mathbf{y} son las muestras de la señal a la salida, normalizada, y N es el número total de muestras.

4 MATERIALES UTILIZADOS

Este capítulo se divide en dos secciones: la primera de ellas consiste en una descripción de las herramientas utilizadas en la realización de las pruebas mientras que en la segunda sección se explica el código generado para la generación de los resultados.

Tan sólo se ha requerido de las siguientes herramientas para abordar nuestro cometido:

- Plataforma WebLab.
- Un ordenador portátil con el software matemático Matlab.

4.1 Plataforma WebLab

La predistorsión digital (DPD) es hoy en día un método comúnmente utilizado para la linealización de transmisores de RF. Sin embargo, aún se necesitan más desarrollos. Por esta razón, en 2014 se propuso la competición de diseño de estudiantes "Linealización de un Amplificador de Potencia mediante Predistorsión Digital" en el International Microwave Symposium (IMS) del IEEE. La competición también sirve como punto de referencia entre algoritmos porque ahora es posible evaluar los diferentes algoritmos en un sistema común, por medio del acceso remoto.

Para ofrecer a los participantes la posibilidad de probar sus algoritmos de linealización por adelantado, se ha creado una página web, mantenida por la Universidad Tecnológica de Chalmers, llamada dpdcompetition.com. Esta página web es la utilizada en este proyecto para llevar a cabo las pruebas. En ella podemos acceder de forma remota al sistema de medición (WebLab) que incorpora un generador de señal, un amplificador de potencia en tecnología GaN y un analizador de señal. Esta configuración permite probar los predistorsionadores considerados. La configuración de medición está basada en un chasis PXI (PXIe-1082) con un PC host integrado de National Instruments. El chasis está equipado con un transceptor de señal vectorial (PXIe-5646R VST) que permite probar señales con un ancho de banda instantáneo de 200 MHz. La señal generada en el VST alimenta a un amplificador lineal (driver) antes de pasar al PA sobre el cual se realizan las medidas. A la salida del dispositivo bajo prueba (DUT) se coloca un atenuador de 30 dB y la salida de éste se conecta al receptor del VST. El PC embebido en el chasis PXI es el encargado de controlar los instrumentos y de cargar y descargar los archivos de datos a petición de los usuarios. El DUT se complementa con un módulo de fuente de alimentación (PXI-4130) que también mide el consumo de corriente del amplificador de potencia. La figura 4-1 ilustra lo explicado anteriormente:

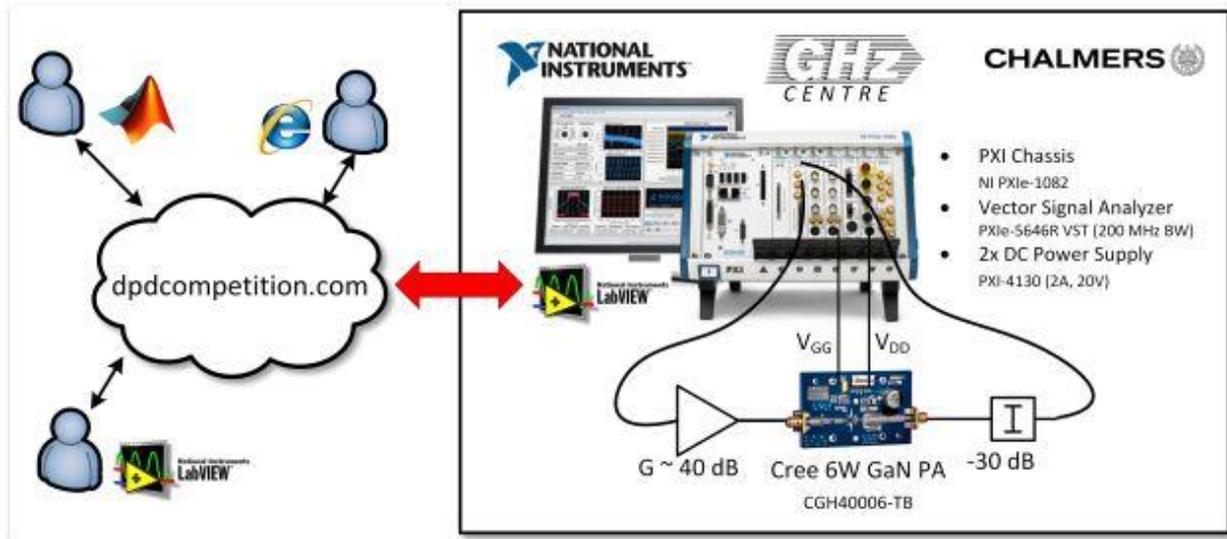


Figura 4-1. Configuración del RFWebLab [12].

El nivel máximo de potencia de salida del PA se limita a aproximadamente 6 W, al establecerse una restricción en la potencia de salida permitida del generador de señal. Los ajustes en el analizador de señal (nivel de referencia y atenuación de entrada) se configuran de tal manera que no distorsionen la señal medida, incluso para el nivel de potencia de pico más alto. Esto significa que experimentará una degradación en el rendimiento de ruido cuando se consideren niveles de potencia inferiores al máximo permitido.

Este acceso remoto a una configuración de medición de última generación se diseñó inicialmente para los participantes en la competición de estudiantes del IMS, pero gracias al patrocinio de equipos de medición propietarios de National Instruments, la plataforma se encuentra permanentemente disponible para todos. Utilizar estos recursos posibilita que aquellos grupos de investigación que no tienen acceso a equipos de medida puedan realizar mediciones utilizando una configuración experimental de última generación y, por lo tanto, aumenten la comprensión de las imperfecciones del hardware en la calidad de la señal en los transmisores modernos de los sistemas de comunicaciones.

Otra de las ventajas de este sistema de medición remoto es que no es necesario registrarse para su uso. Existen dos opciones para cargar los archivos de datos:

- Usar cualquiera de las funciones de Matlab proporcionadas por el grupo de Chalmers: PA_meas_Pin.m o PA_meas_xs.m.
- Usar LabVIEW VI.

La utilizada en este proyecto será la primera. Con el fin de proteger el sistema remoto de medición WebLab, nuestro código y el de cualquier usuario que desee acceder a él debe cumplir las siguientes restricciones:

Para proteger el amplificador y el sistema contra niveles excesivamente altos, se imponen limitaciones a la potencia máxima de salida y a la potencia de salida rms del generador de señal vectorial. Se supone que el sistema tiene una impedancia de referencia de 50 ohm. La potencia máxima permitida del generador de señal es -8 dBm. El nivel máximo permitido de potencia rms ($P_{in, RMS}$) depende de la relación de potencia pico a promedio (PAPR) de la señal de entrada y asegurará que los picos de las señales de entrada permanezcan durante la mayor parte del tiempo por debajo de -8 dBm. En este sentido, se permite incrementar ligeramente la potencia

RMS cuando se trabaja con señales de PAPR baja. También se establece una relación máxima de potencia de pico a promedio de la señal de datos de BB de 20 dB. Si se excede cualquiera de estos límites, el sistema no realizará la medición. El resultado en estos casos es un archivo de datos que contiene un único valor "-inf".

Se acepta como entrada un máximo de 1 000 000 de muestras de datos de BB. Si se carga un archivo con más de 1 000 000 de muestras, solo se utilizan las primeras 1 000 000 de muestras. El número de muestras en el archivo de datos de salida (csv o dat) es de la misma longitud que la señal de entrada (o 1 000 000 de muestras si la señal de entrada era más larga).

La velocidad máxima de muestreo tanto en el generador como en el analizador es de 200 MHz. El ancho de banda máximo útil es de 160 MHz. Esto significa que cualquier componente de señal que se coloque fuera del rango de frecuencia [-80 80] MHz se elimina antes de ser enviado desde el generador de señal. La señal que se recibe del analizador de señal también se limita al ancho de banda de 160 MHz, es decir, utiliza solo las componentes de señal en el rango [-80 80] MHz porque todo lo que esté fuera de este rango se distorsiona por un filtrado.

4.2 Matlab

Es un entorno de cálculo técnico de altas prestaciones para el cálculo numérico y visualización. Entre sus prestaciones básicas destacan el análisis numérico, el cálculo matricial, el procesamiento de señales y la representación de gráficos.

Es un entorno intuitivo y fácil de usar, donde los problemas y las soluciones son expresados como se escriben matemáticamente, sin la programación tradicional. Es un sistema interactivo cuyo elemento básico de datos es una matriz que no requiere dimensionamiento. Esto permite resolver muchos problemas numéricos en una fracción más pequeña del tiempo que llevaría hacerlo en otros lenguajes de programación como C, Basic y Fortran. Entre sus prestaciones básicas se hallan: la manipulación de matrices, la representación de datos y funciones, la implementación de algoritmos y la comunicación con programas en otros lenguajes y con otros dispositivos hardware.

Matlab también proporciona una serie de soluciones específicas denominadas Toolboxes. Estas son muy importantes para la mayoría de los usuarios ya que son conjuntos de funciones que permiten extender el entorno de Matlab para resolver determinados problemas como:

- Procesamiento de señales.
- Diseño de sistemas de control.
- Simulación de sistemas dinámicos.

Para el desarrollo de este proyecto se emplean dos Toolboxes: el primero de ellos es "Signal Processing ToolBox" el cual es un paquete que proporciona funciones y aplicaciones para generar, medir, transformar, filtrar y visualizar señales. Además, incluye algoritmos para remuestrear, suavizar y sincronizar señales, diseñar y analizar filtros, estimar espectros de potencia y medir picos, ancho de banda y distorsión. El segundo es el "Communications System Toolbox" caracterizado por proporcionar algoritmos y aplicaciones para el análisis, diseño, simulación extremo a extremo, y la verificación de los sistemas de comunicaciones en Matlab.

4.3 Consideraciones sobre el código generado

En este proyecto se han implementado dos tipos de predistorsionadores, el primero de ellos basado en el modelo sin memoria (ML) y el segundo basado en el Memory Polynomial (MP). Para cada uno de los predistorsionadores se han utilizados tres algoritmos distintos para el cálculo de sus coeficientes. Estos tres algoritmos han sido presentados en el tercer capítulo y son: ajuste LS, cálculo de la inversa por bloques y aproximación de la inversa por una matriz de covarianza. A continuación, primeramente se presenta la señal de entrada con la que trabajaremos y posteriormente se describe el funcionamiento de cada uno de los archivos que componen nuestro código.

4.3.1 Señal de entrada

La señal de entrada simulada para la generación de los resultados es una señal OFDM con 512 subportadoras y una constelación 128-QAM. Las suportadoras están espaciadas para producir un ancho de banda de aproximadamente 10 MHz. La señal OFDM está centrada en el origen $[-5 \ 5]$ MHz (BB), y es filtrada utilizando un filtro de coseno alzado. Tiene una frecuencia de muestreo fijada a 200 MHz y una duración de 500 μ s. El nivel de potencia RMS de la señal de entrada es de -22 dBm; éste es el valor máximo de potencia permitido que cumple con las especificaciones de seguridad establecidas por la plataforma, pues la PAPR típica de la señal de entrada es del orden de 9 dB.

A continuación, en las figuras 4-2 y 4-3 se representan el voltaje de la parte real e imaginaria frente al tiempo y el espectro de densidad de potencia respectivamente de la señal OFDM simulada:

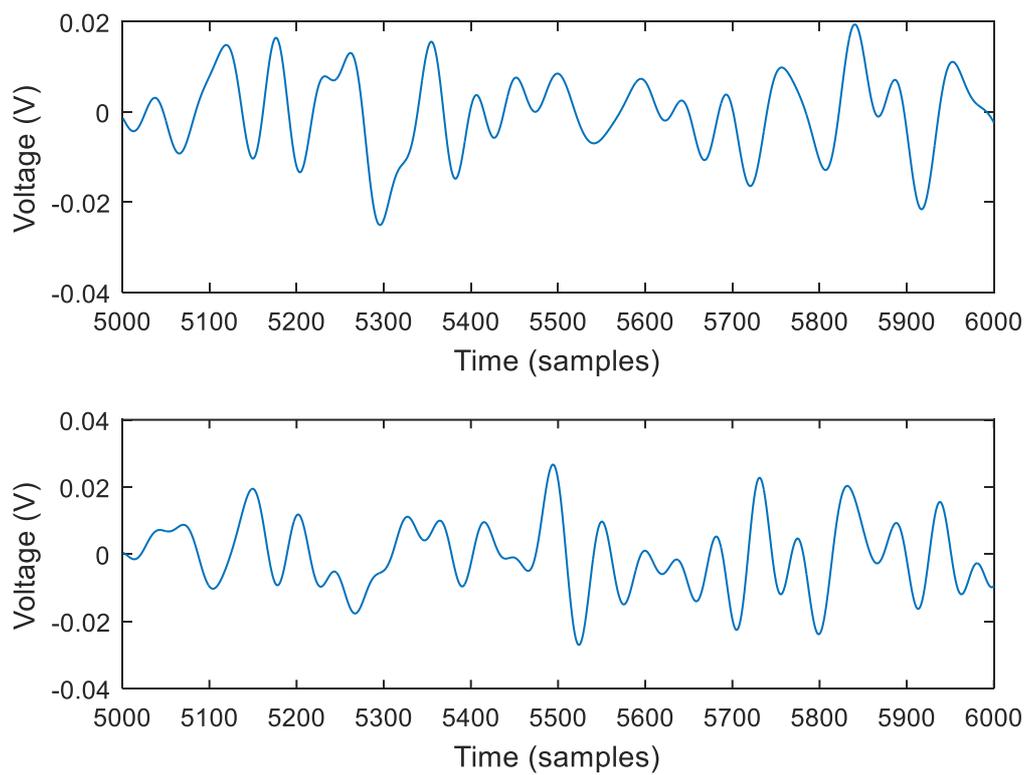


Figura 4-2. Voltaje frente a tiempo de la parte real e imaginaria de la señal de entrada.

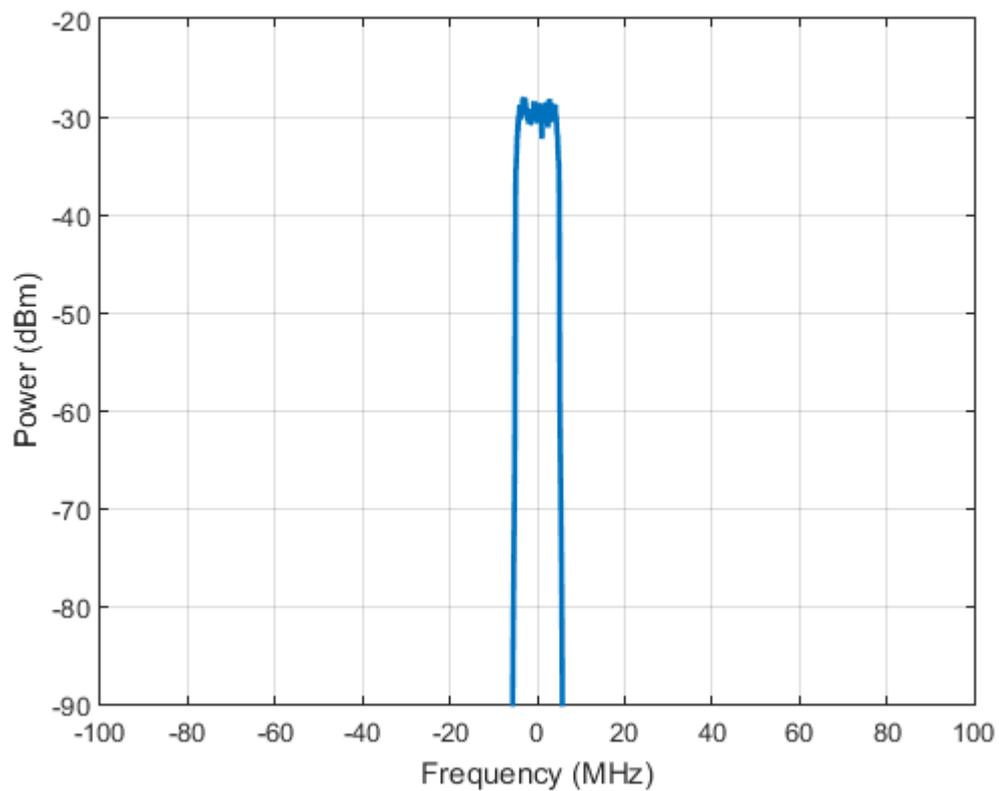


Figura 4-3. Espectro de densidad de potencia de la señal de entrada.

4.3.2 Descripción del código generado

- `IMS2016_generate_signal`: El objetivo de esta función es generar la señal presentada en el apartado 4.3.1, que posteriormente será enviada al PA para realizar las simulaciones. Recibe como parámetro una semilla utilizada para la generación de los datos aleatorios que conforman la señal OFDM. Devuelve el vector `x_ofdm`, que contiene las muestras de la envolvente compleja discreta de la señal generada, la variable `fs` que representa la frecuencia de muestreo y la variable `f_ofdm` que representa la frecuencia central de la señal OFDM. Las características de la señal generada han sido expuestas en el apartado anterior.
- `RFWebLab_PA_meas_v1_1`: Esta función es la encargada de enviar la señal generada por `IMS2016_generate_signal` al amplificador de potencia remoto. Recibe la señal OFDM generada (muestras en BB) y su potencia en dBm. Además, antes de enviar la señal al amplificador comprueba que cumpla con todos los requisitos expuestos en el apartado Plataforma WebLab de este capítulo. Esta función devuelve las muestras BB de la señal a la salida del amplificador de potencia, la potencia en dBm de ésta así como la corriente y voltaje medidos. La señal devuelta tiene la misma longitud que la enviada pero debemos tener en cuenta que el muestreo se inicia de forma completamente aleatoria, por lo que se debe realizar una sincronización antes de utilizar los datos de salida.
- `Mysynch_IMS2016`: Recibe la señal enviada al amplificador (generada por `IMS2016_generate_signal`) y la señal a la salida de éste. Devuelve ambas señales sincronizadas, resolviendo de este modo el problema de sincronización ya mencionado al describir la función `RFWebLab_PA_meas_v1_1`. Además, devuelve la potencia media de salida, el ACPR de la señal OFDM y el NMSE entre la entrada y la salida del amplificador.
- `Espectro`: Recibe como parámetros de entrada la envolvente compleja discreta de una señal y su frecuencia de muestreo. Devuelve su espectro de densidad de potencia. En primer lugar, se crea una ventana de Kaiser con un ancho de banda de 8 kHz y una atenuación de lóbulo lateral de 50 dB. Utilizando la ventana anterior, se estima la densidad espectral de potencia a través del periodograma de Welch, obtenido mediante la función `pwelch` implementada en Matlab, utilizando 8000 puntos. Finalmente, la densidad espectral de potencia se centra en el origen utilizando la función `fftshift`.
- `Test_MLfinal`: Este script implementa el predistorsionador basado en el polinomio sin memoria. Como utilizamos la configuración de aprendizaje indirecto comenzamos obteniendo las señales sincronizadas a la entrada y salida del amplificador de potencia. Para ello se hace uso de las funciones expuestas anteriormente. Continuamos con el cálculo de la ganancia de compresión para poder modelar correctamente la señal de entrada al predistorsionador que será la señal de salida del amplificador de potencia dividida por la ganancia de compresión. Una vez tenemos las señales de entrada y salida bien dimensionadas calculamos los coeficientes del predistorsionador resolviendo la ecuación $\mathbf{w} = \mathbf{Y}^{-1} \cdot \mathbf{x}$, donde \mathbf{Y}^{-1} representa la pseudoinversa de la matriz \mathbf{Y} . Esta ecuación podrá ser resuelta de tres formas distintas: mediante la función `pinv` implementada en Matlab la cual utiliza el algoritmo de LS, mediante la estimación modificada Least-Square implementada en la función `calpseudoinv` o mediante el método de la matriz de covarianza implementado por la función `pcovarianza`. Tras haber obtenidos los coeficientes se comprueba el sistema completo, es decir, la cascada de predistorsionador y amplificador de potencia. Este proceso se repite 2 veces para obtener mejores resultados en la linealización con la única diferencia de que en las siguientes iteraciones la señal de entrada al predistorsionador, para la estimación de sus coeficientes, continuará siendo la salida del amplificador de potencia, pero su salida ahora será la salida del predistorsionador de la iteración anterior la cual hemos denominado por `z`.
- `Test_MPfinal`: Este script implementa el predistorsionador basado en el Memory Polynomial. Tiene la misma estructura que el `test_MLfinal` explicado antes con la única salvedad de que este predistorsionador tiene memoria. Para introducir la memoria se ha tenido que modificar la longitud del vector de entrada al predistorsionador añadiendo `Q` muestras más que en el vector de salida. El valor de `Q` representa la profundidad de la memoria.
- `Calpseudoinv`: Esta función recibe la matriz formada por las muestras de entrada al predistorsionador (\mathbf{Y}) y el número de regresores del modelo. Devuelve la pseudoinversa de dicha matriz obtenida

mediante la estimación modificada Least-Square explicada en el capítulo 3.

- **Pcovarianza:** Recibe la matriz formada por las muestras de entrada, el número de muestras utilizado para modelar los vectores de entrada y salida, la señal de salida del predistorsionador y el número de regresores del modelo. Devuelve los coeficientes del predistorsionador hallados mediante el método de la matriz de covarianza. Hace uso de una función propia de Matlab denominada cov que cuando recibe dos vectores de igual longitud devuelve una matriz 2x2 con la varianza del primero vector colocada en la posición (1,1), la varianza del segundo vector colocada en la posición (2,2) y con la covarianza de los dos vectores colocadas en la posición (1,2) y (2,1). Elegimos cualquiera de estas dos últimas posiciones para modelar la matriz de covarianza que como se explicó en el capítulo anterior está compuesta por la covarianza de los regresores.

5 RESULTADOS

En este capítulo se expondrán los resultados obtenidos al aplicar los modelos de predistorsión presentados en el capítulo 2 al PA remoto descrito en el capítulo 4, poniéndose de manifiesto los efectos beneficiosos que introduce la predistorsión. Se analizarán los siguientes casos:

- Caso sin DPD.
- ML con algoritmo LS.
- MP con algoritmo LS.
- ML con algoritmo de cálculo de la inversa por bloques.
- MP con algoritmo de cálculo de la inversa por bloques.
- ML con algoritmo de aproximación de la inversa por una matriz de covarianza.
- MP con algoritmo de aproximación de la inversa por una matriz de covarianza.

Para todos los casos la señal de entrada será común, siendo ésta una señal OFDM en BB como la presentada en 4.3.1. Su potencia es de -22 dBm que es el valor máximo que puede tomar sin violar las especificaciones de seguridad establecidas por la plataforma RF WebLab.

Con el fin de poder evaluar cada uno de los casos, se representan en todos ellos las siguientes gráficas: espectro de la señal de salida del PA, característica AM/AM, ganancia frente a potencia de entrada y característica AM/PM. Además, se calculará el valor del ACPR y NMSE en cada uno de los casos para denotar cómo afecta en éstos el uso de DPD o el tipo de algoritmo utilizado.

5.1 Caso sin DPD

En este caso no se utilizará predistorsionador, simplemente la señal es generada y enviada directamente al PA sin ningún tipo de preprocesado. A continuación, se muestran los cambios generados por el PA en la señal de entrada presentada en 4.3.1 (véanse las figuras 5-1 a 5-4):

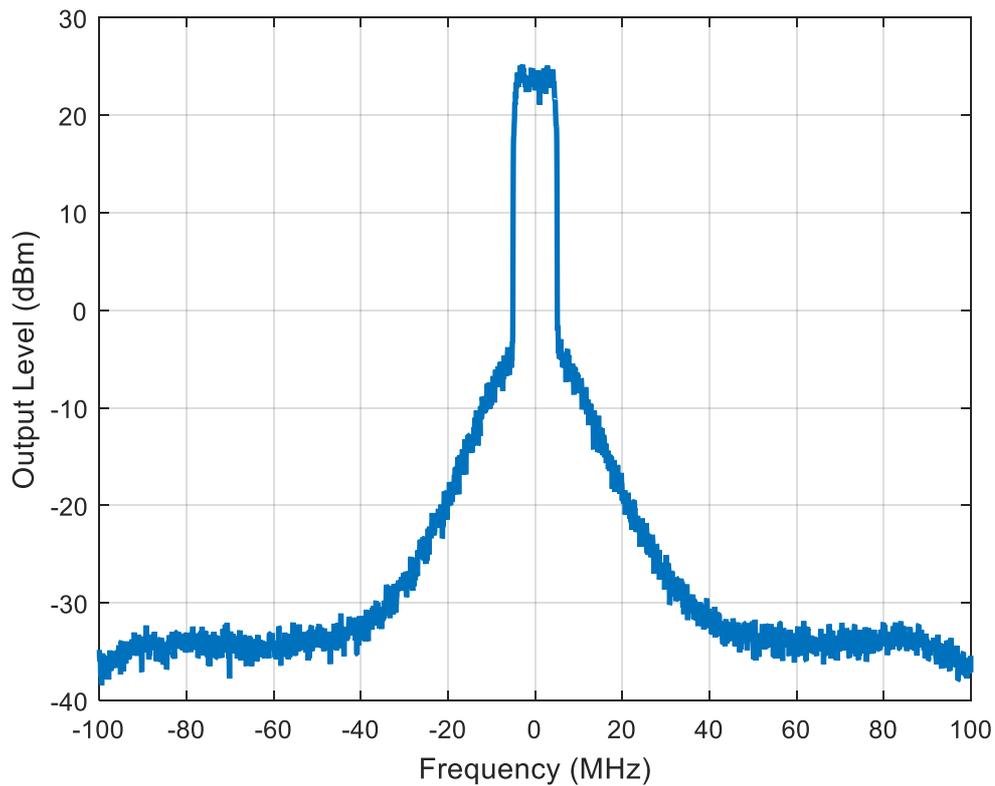


Figura 5-1. Espectro de la señal de salida del PA sin DPD.

En la figura 5-1 se puede observar como el nivel de potencia de la señal ha aumentado tras su paso por el PA, sin embargo, este hecho ha provocado también un recrecimiento espectral. Vemos que para niveles de potencia inferiores a los -5 dBm el espectro comienza a ensancharse, ocupando un ancho de banda superior a su inicial e interfiriendo en los canales adyacentes. Este recrecimiento, como se indicó en capítulos anteriores tiene una serie de consecuencias negativas y por ello debemos mitigarlo. Para dicho cometido, en los apartados siguientes se mostrará como el uso del DPD permite disminuirlo.

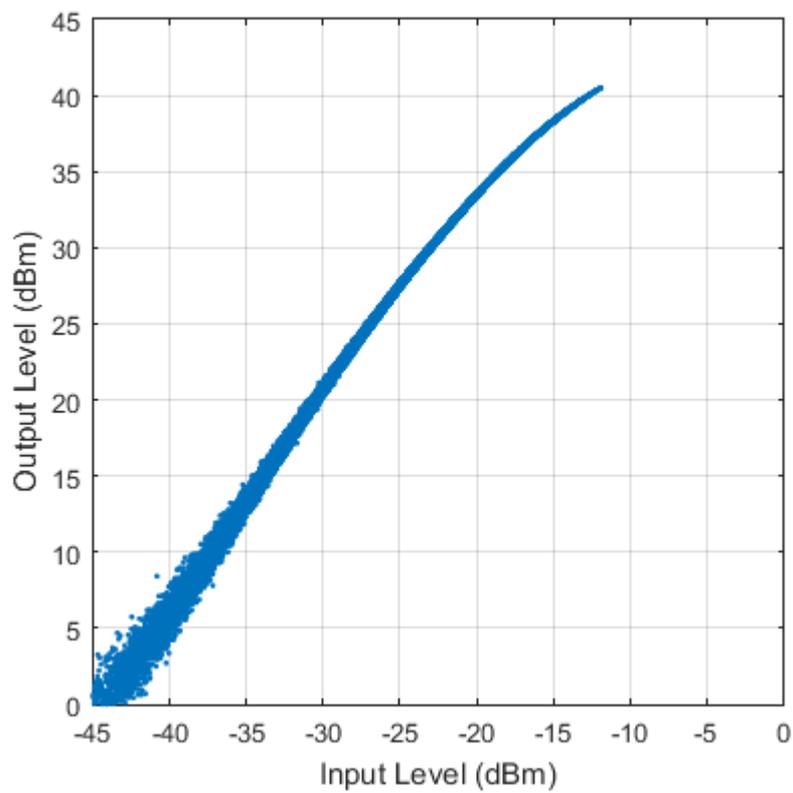


Figura 5-2. Característica AM/AM sin DPD.

En la figura 5-2 se observa una zona donde la potencia de salida crece linealmente con la potencia de entrada, esta zona es conocida como zona lineal. Está comprendida entre los -45 y -24 dBm, para potencias de entrada superiores a -24 dBm la potencia de salida ya no crece linealmente sino que satura. Esta zona se conoce como zona de saturación y es en la que se trabajará para obtener la mayor eficiencia por parte del PA.

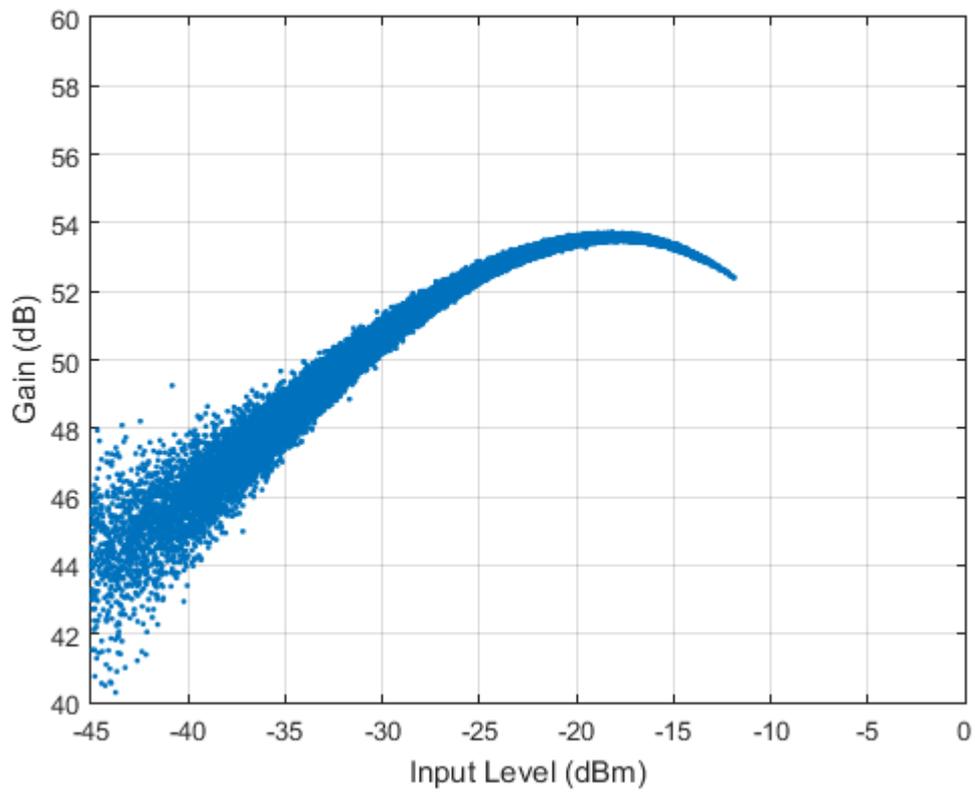


Figura 5-3. Ganancia frente a potencia de entrada sin DPD.

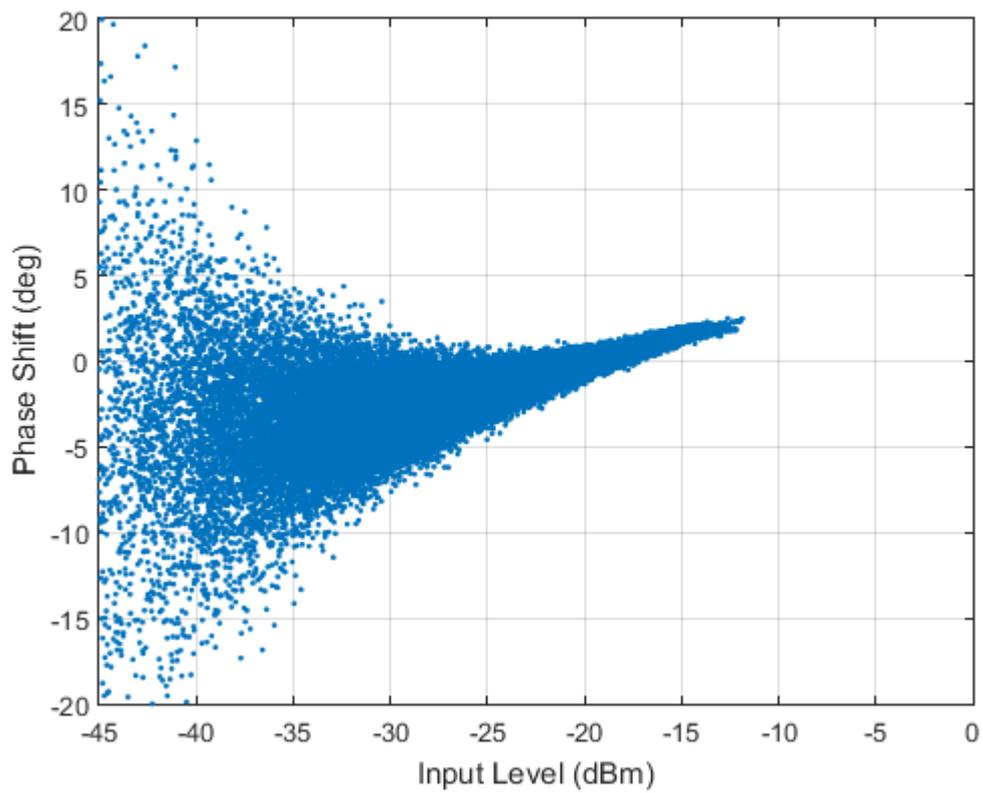


Figura 5-4. Característica AM/PM sin DPD.

Para el caso sin DPD analizado, los valores de ACPR y NMSE a la salida del PA son 31.6 dB y -24.0 dB respectivamente.

5.2 ML con algoritmo LS

Este predistorsionador está basado en el modelo sin memoria presentado en la ecuación (2-4), el número de muestras seleccionado para modelar las señales de entrada y salida es de 10000. El orden del polinomio que describe el comportamiento del DPD es $P=13$.

Una vez obtenido el valor de los coeficientes del DPD se realizan dos nuevas iteraciones siguiendo el mismo proceso, con la salvedad de que las señales de entrada y salida utilizadas para modelar los nuevos coeficientes del DPD son las de la iteración anterior. Estas iteraciones son realizadas con el fin de mejorar el resultado de la linealización.

Los resultados obtenidos se muestran en las figuras 5-5 a 5-8:

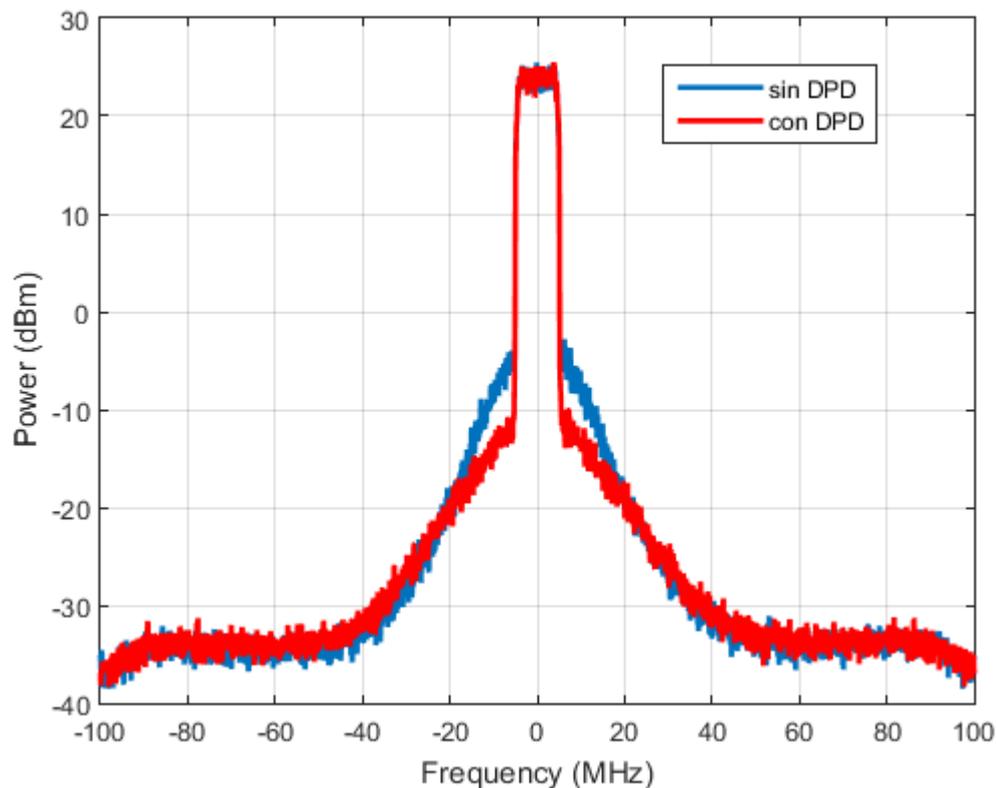


Figura 5-5. Espectro señal de salida (ML algoritmo LS).

Como podemos observar en la figura 5-5, el uso del DPD ha mitigado el ensanchamiento espectral. Aunque dicho ensanchamiento no se ha erradicado, con el DPD se produce para niveles de potencia inferiores lo cual es beneficioso ya que el nivel de potencia interferente en el canal adyacente es menor.

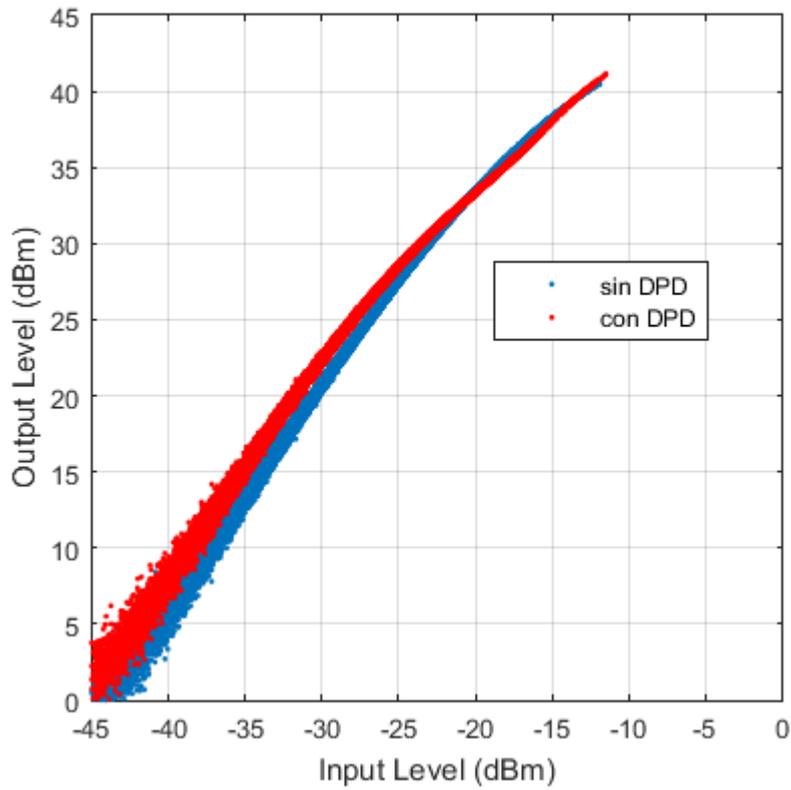


Figura 5-6. Característica AM/AM (ML algoritmo LS).

Con el uso del DPD para un mismo nivel de potencia de entrada obtenemos un nivel de potencia de salida mayor. Además, se ha conseguido aumentar el rango de la zonal lineal.

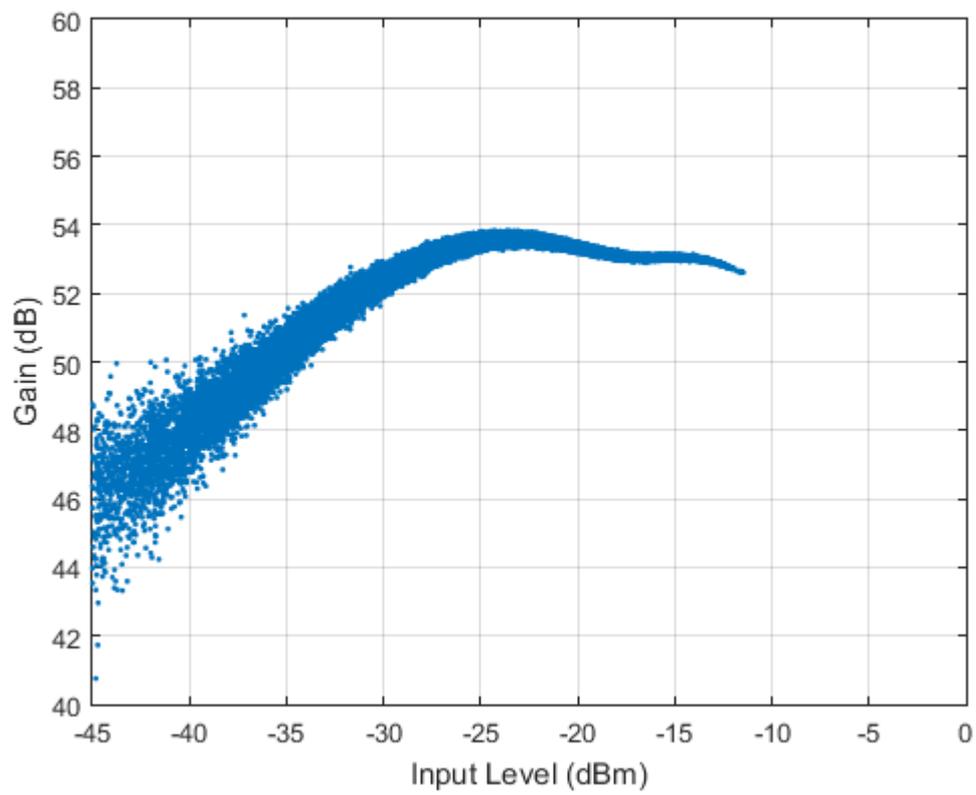


Figura 5-7. Ganancia frente a potencia de entrada (ML algoritmo LS).

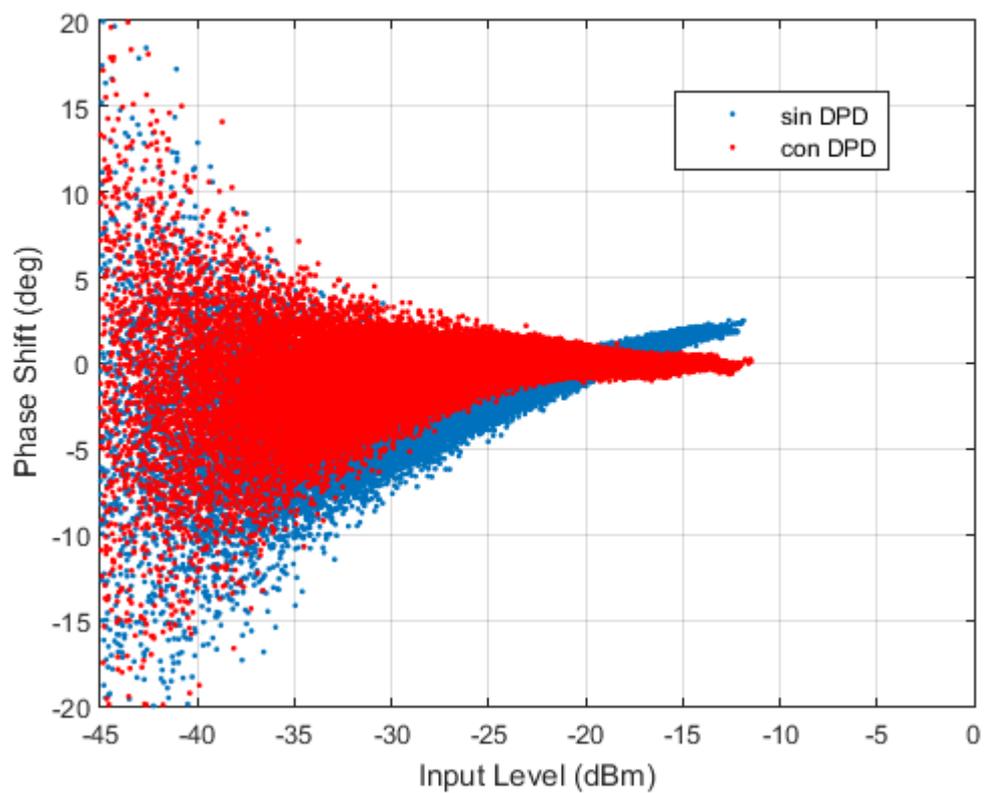


Figura 5-8. Característica AM/PM (ML algoritmo LS).

Con respecto a los valores de ACPR y NMSE, el uso del DPD ha provocado mejoras. Inicialmente, se ha calculado el NMSE en ausencia del DPD y se ha obtenido un valor de -23.8 dB. Tras introducir el DPD y realizar 2 iteraciones, como se ha indicado al principio, se ha conseguido disminuir el NMSE obteniéndose un valor de -29.6. De igual manera también ha mejorado el valor del ACPR, incrementándose desde un valor inicial de 31.8 dB (sin DPD) hasta un valor final de 37.1, obtenido tras las 2 iteraciones en presencia del DPD.

5.3 MP con algoritmo LS

En este caso se trabaja con el DPD basado en el modelo MP presentado en la ecuación (2-5). El número de muestras seleccionado para modelar las señales de entrada y salida es de 10000. La profundidad de la memoria es de $Q=3$ y el orden del polinomio $P=13$.

Como en el caso del ML se realizan dos iteraciones una vez estimado el valor de los coeficientes con el fin de optimizar este valor y obtener una mejora en la linealización. El algoritmo utilizado para la estimación de estos es el LS. Los resultados obtenidos con el modelo descrito se representan en las figuras 5-9 a 5-12 y la tabla 5-1:

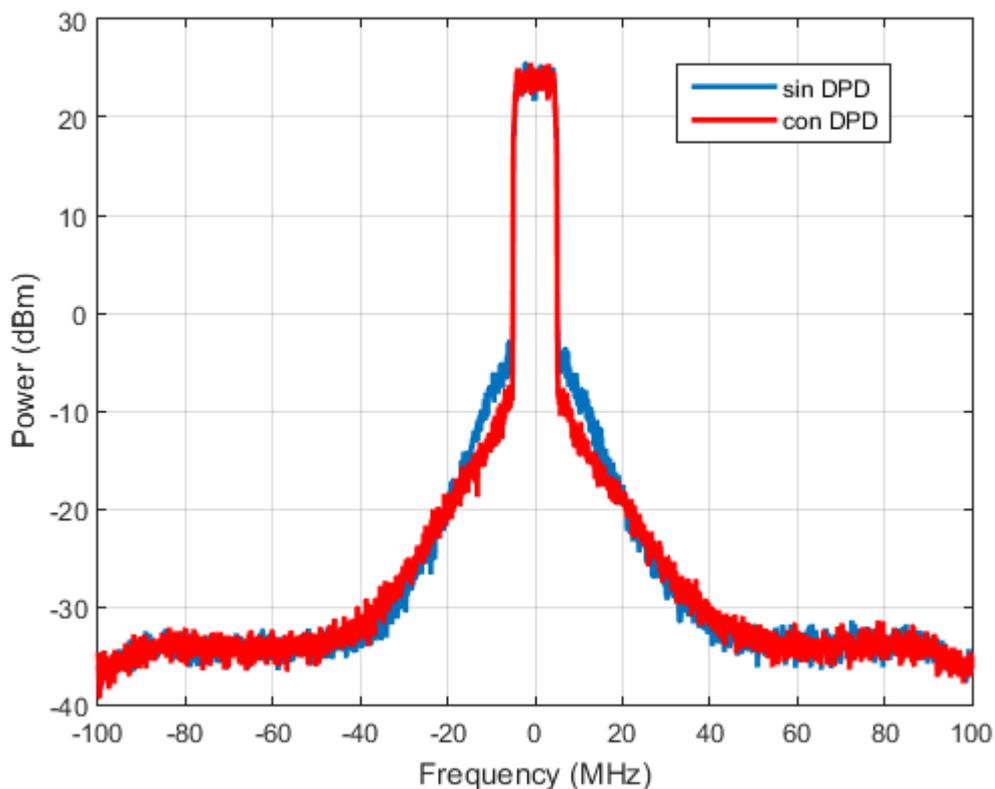


Figura 5-9. Espectro señal de salida (MP algoritmo LS).

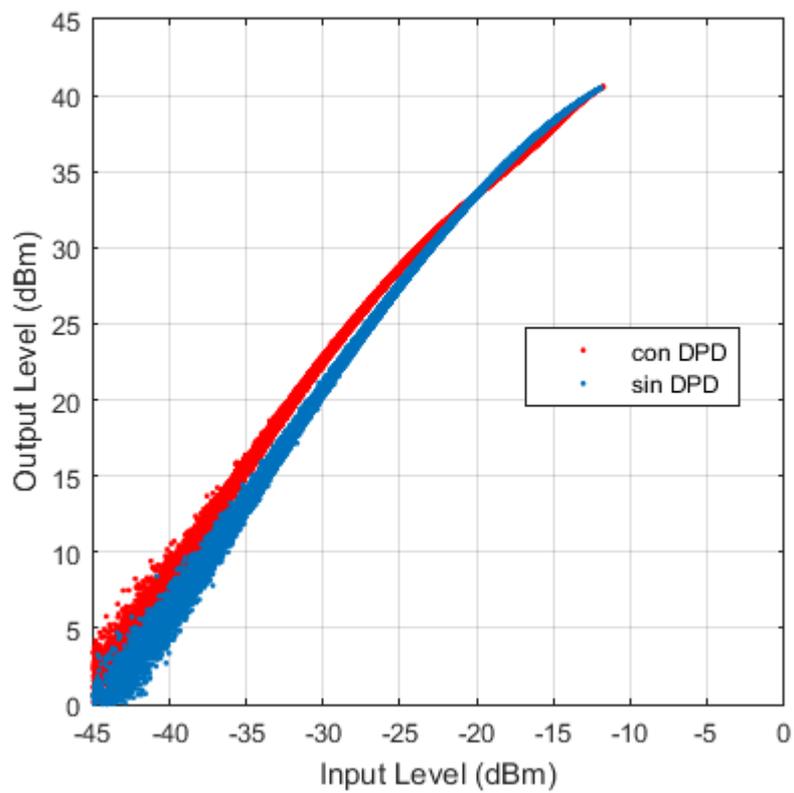


Figura 5-10. Característica AM/AM (MP algoritmo LS).

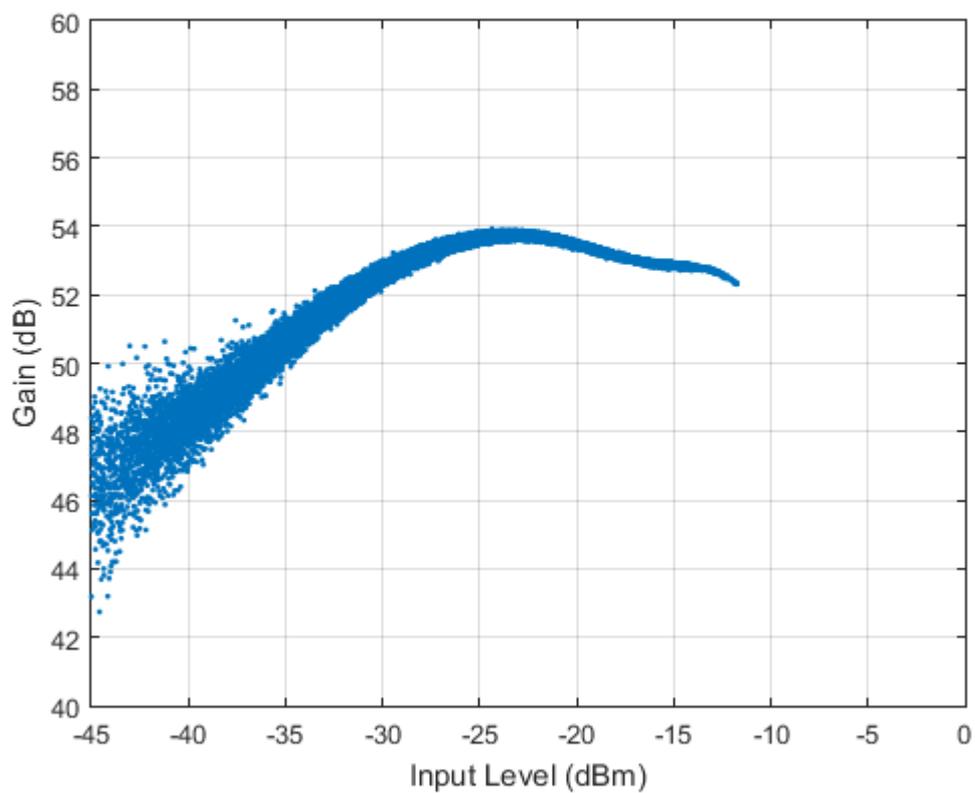


Figura 5-11. Ganancia frente a potencia de entrada (MP algoritmo LS).

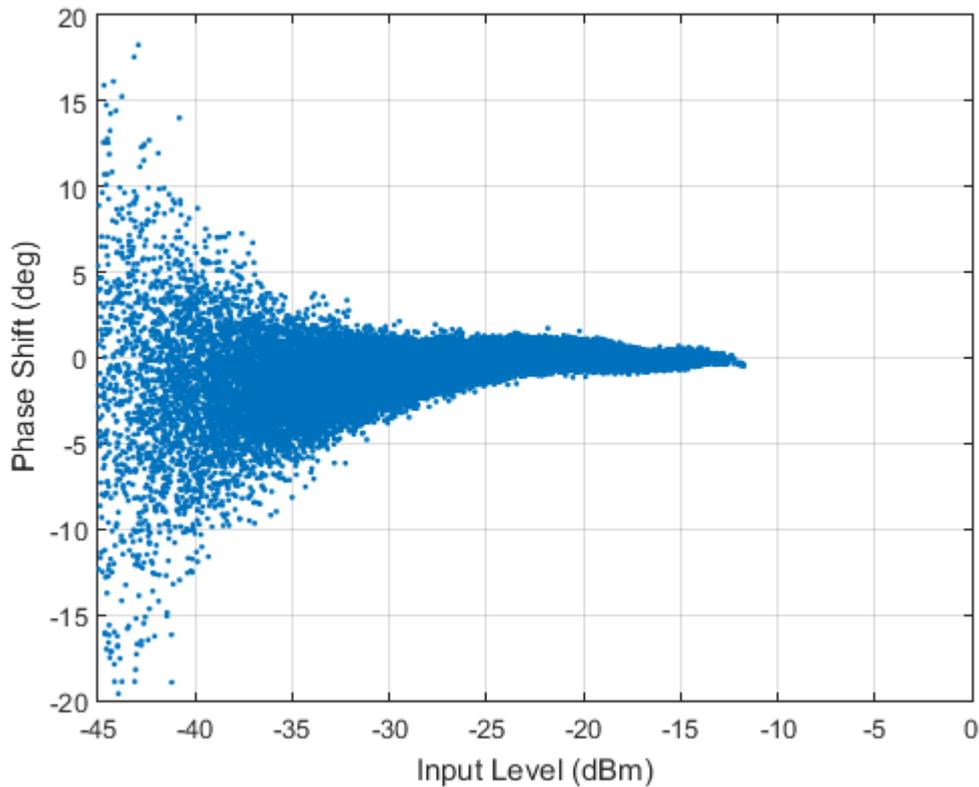


Figura 5-12. Característica AM/PM (MP algoritmo LS).

Tabla 5-1. Valores ACPR y NMSE con MP algoritmo LS

Casos	ACPR (dB)	NMSE (dB)
Sin DPD	31.8	-23.9
Con DPD	36.0	-27.7
DPD 1 iteración	36.5	-28.2
DPD 2 iteraciones	36.0	-27.8

Analizando la tabla 5-1 podemos ver que los parámetros no mejoran progresivamente sino que obtienen su valor óptimo tras la primera iteración. En la iteración final los valores tanto de ACPR como NMSE se estabilizan alcanzando un buen valor que mejora el caso sin iteración pero están un poco por debajo del caso con una sola iteración.

5.4 ML con algoritmo de cálculo de la inversa por bloques

El DPD presentado tiene las mismas dimensiones que el DPD del apartado 5.2, es decir, mismo número de muestras utilizado en el modelado y mismo orden. También se realizan dos iteraciones adicionales para optimizar la linealización. La única diferencia es que la inversa de la matriz del sistema es obtenida mediante el algoritmo presentado en [3]. Por simplicidad en las gráficas hemos denotado algoritmo de cálculo de la inversa por bloques como IB. Los resultados obtenidos se muestran en las figuras 5-13 a 5-16 y la tabla 5-2:

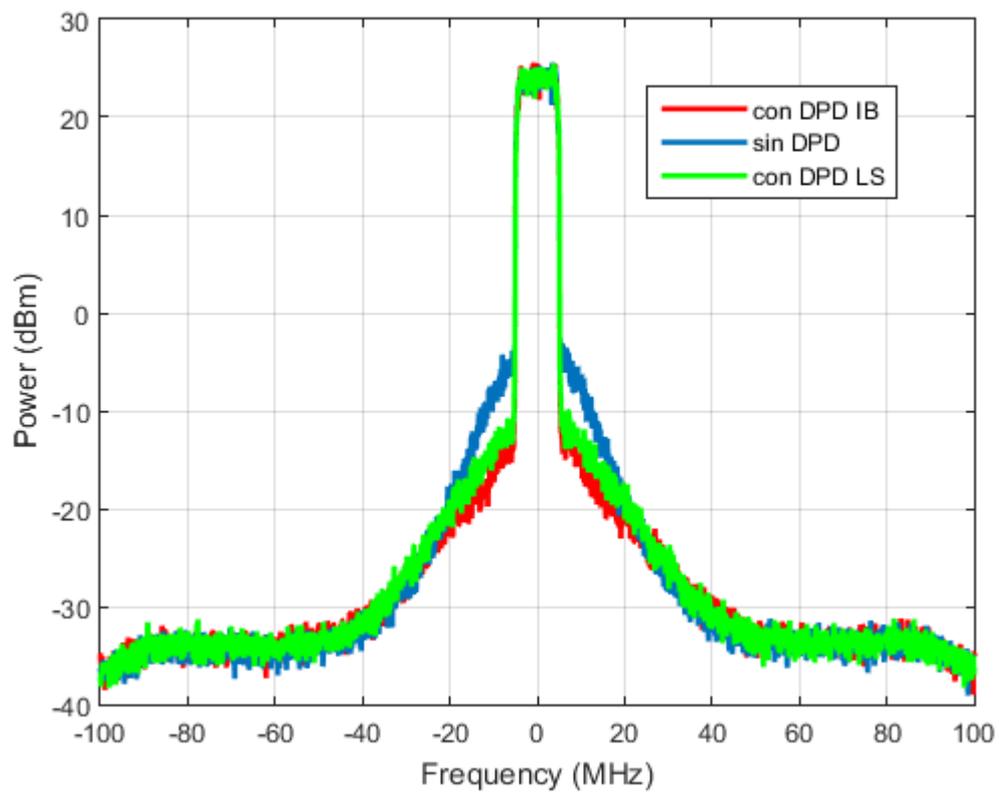


Figura 5-13. Comparativa de espectros de señal de salida.

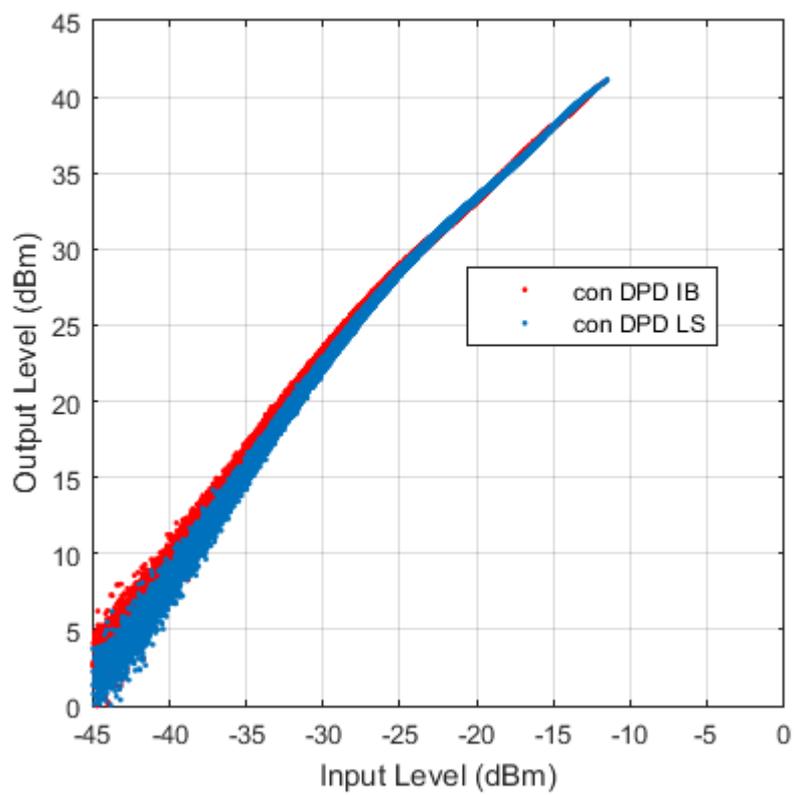


Figura 5-14. Característica AM/AM (ML algoritmo IB).

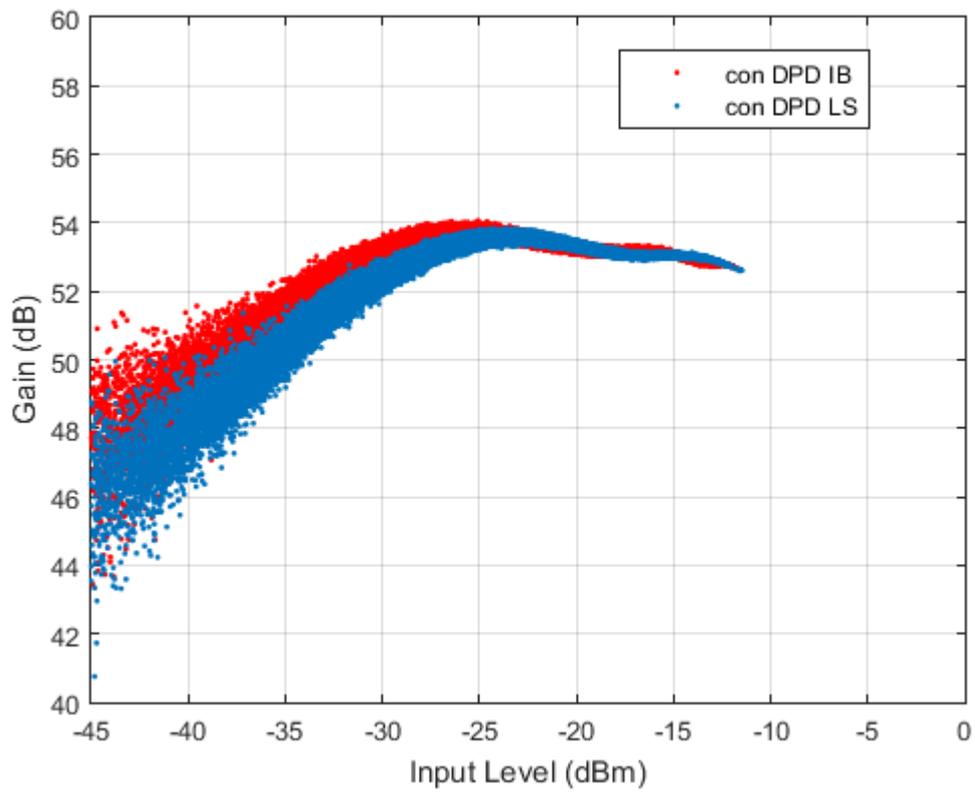


Figura 5-15. Ganancia frente a potencia de entrada (ML algoritmo IB).

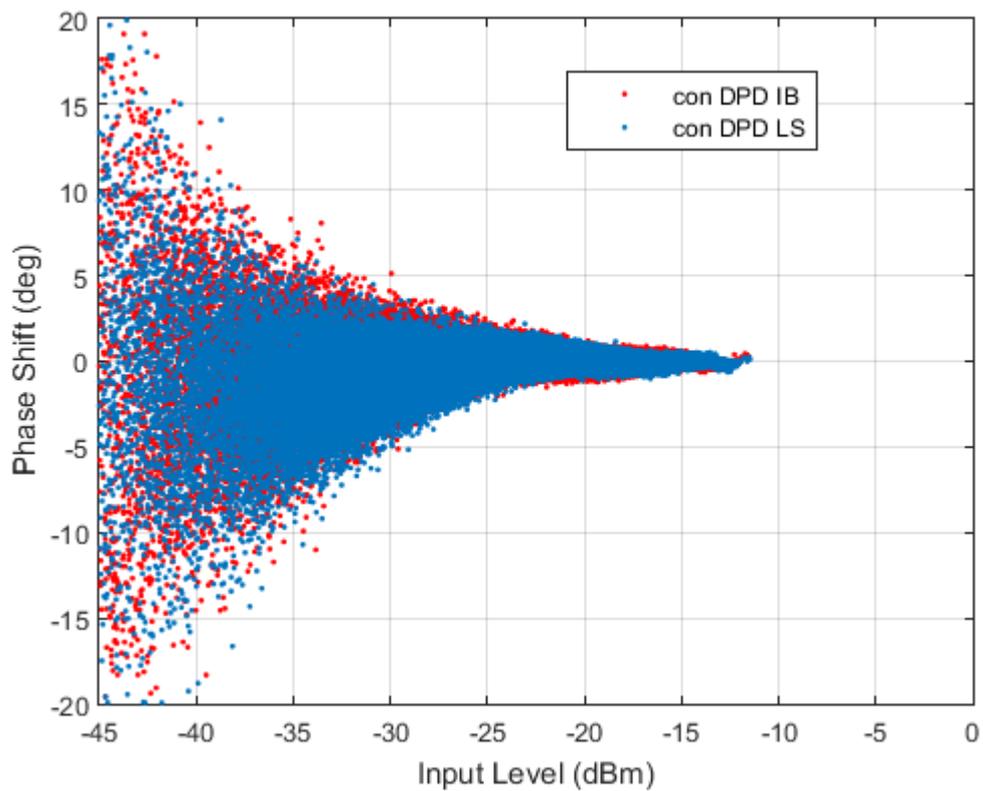


Figura 5-16. Característica AM/PM (ML algoritmo IB).

Tabla 5-2. Valores ACPR y NMSE con ML algoritmo IB

Casos	ACPR (dB)	NMSE (dB)
Sin DPD	31.7	-23.9
Con DPD	38.9	-30.5
DPD 1 iteración	40.1	-31.9
DPD 2 iteraciones	39.2	-30.9

La tabla 5-2 representa el mismo tipo de comportamiento que el analizado en la tabla 5-1. En ella los parámetros no crecen progresivamente con cada iteración sino que obtienen su valor óptimo tras la primera iteración. Los resultados obtenidos en la tabla 5-2 constituyen los mejores valores obtenidos para los distintos algoritmos analizados en el modelo ML.

5.5 MP con algoritmo de cálculo de la inversa por bloques

El método utilizado para calcular la inversa en este caso es el presentado en [3]. El dimensionamiento del DPD es idéntico al de la sección 5.3, y de nuevo se ha adoptado la abreviatura IB para referirnos al algoritmo utilizado. Se han obtenido los resultados que se muestran en las figuras 5-17 a 5-20 y la tabla 5-3:

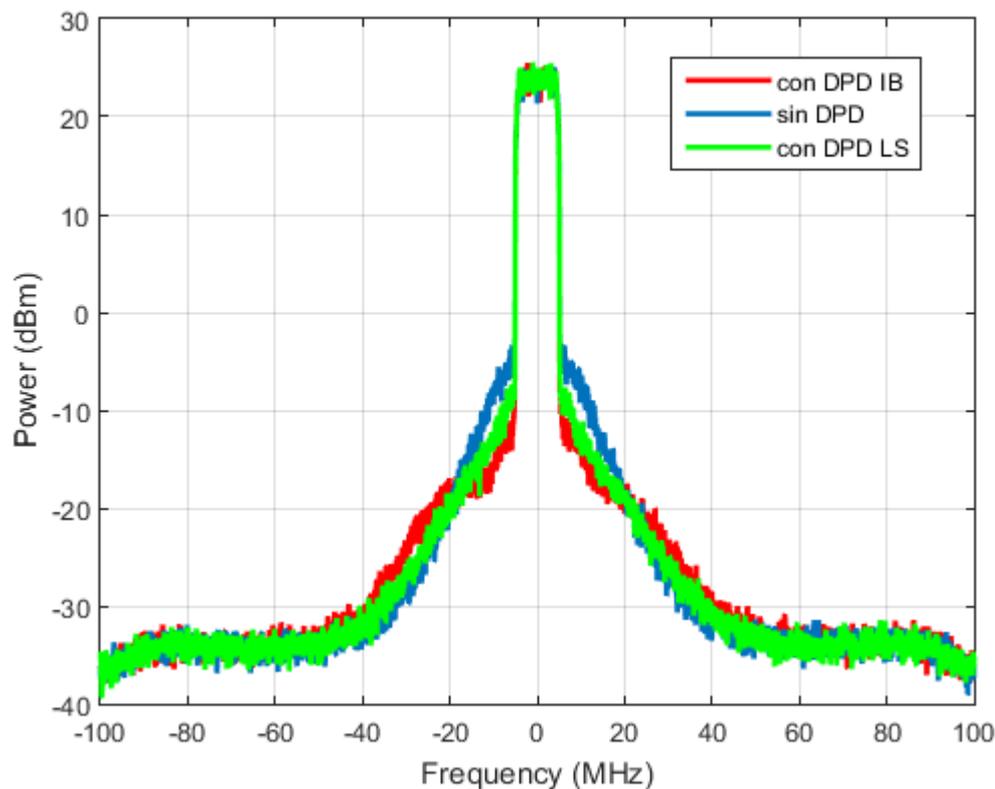


Figura 5-17. Comparativa de espectros de la señal de salida.

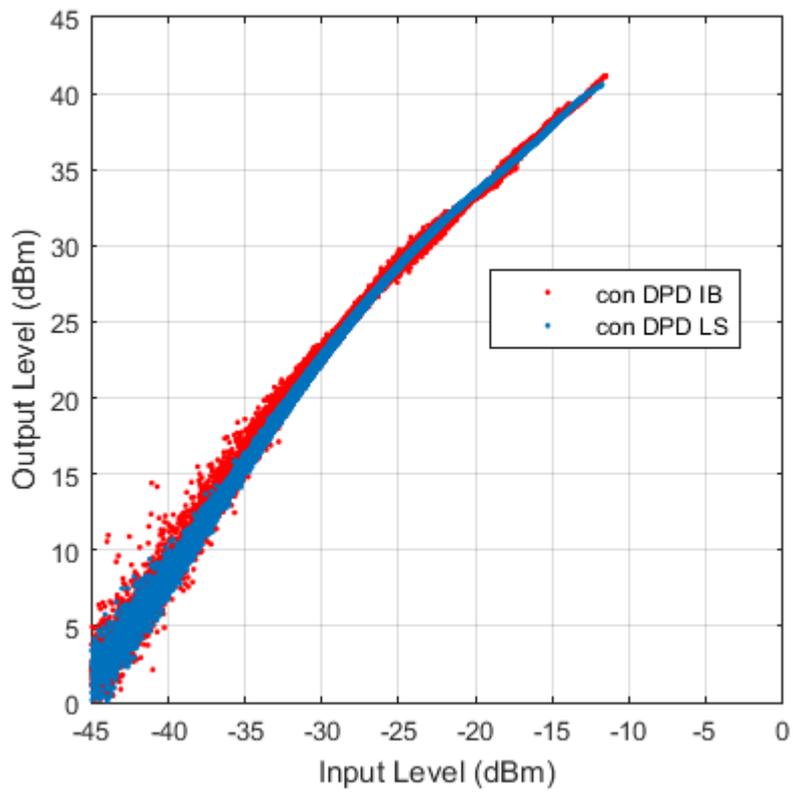


Figura 5-18. Característica AM/AM (MP algoritmo IB).

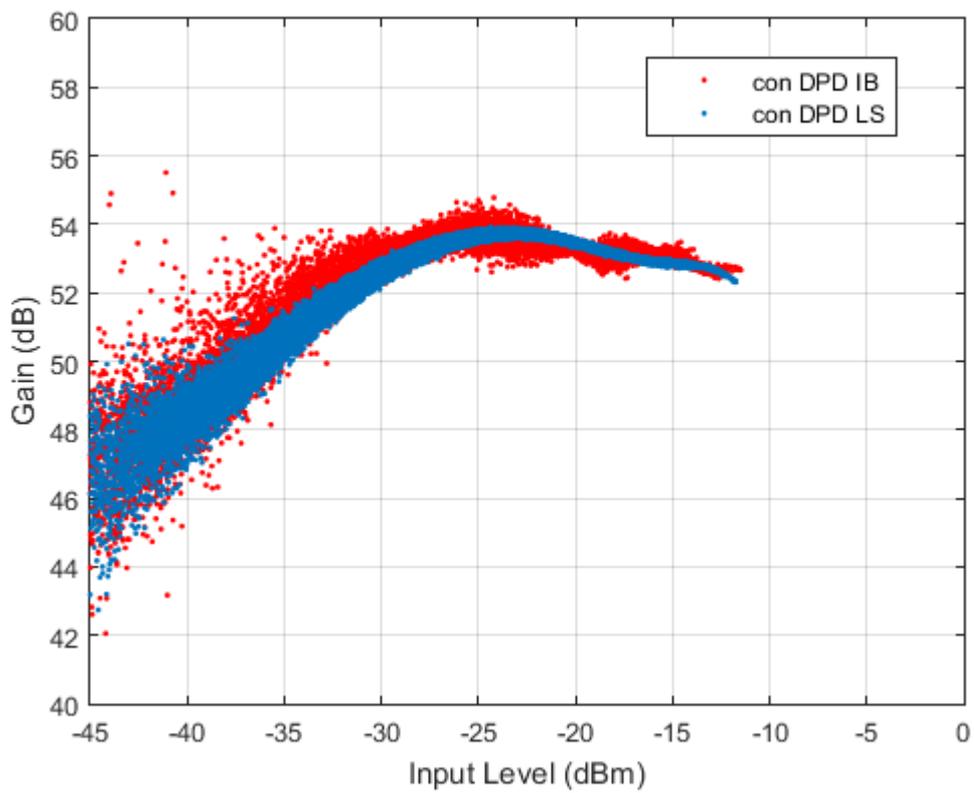


Figura 5-19. Ganancia frente a potencia de entrada (MP algoritmo IB).

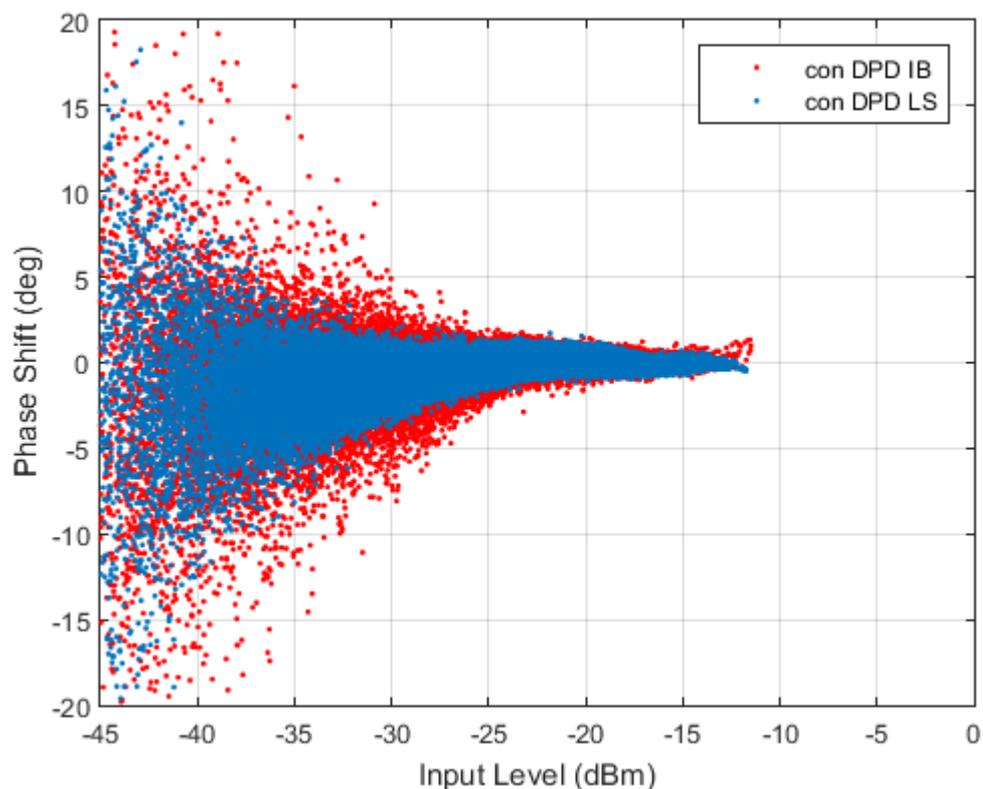


Figura 5-20. Característica AM/PM (MP algoritmo IB).

Tabla 5-3. Valores ACPR y NMSE con MP algoritmo IB

Casos	ACPR (dB)	NMSE (dB)
Sin DPD	31.7	-23.9
Con DPD	37.9	-29.4
DPD 1 iteración	38.1	-29.8
DPD 2 iteraciones	38.1	-29.8

La tabla 5-3 representa un comportamiento del modelo distinto al de las tablas anteriores. En este caso los valores tanto de ACPR como de NMSE obtienen su valor óptimo tras la primera iteración, pero éste se mantiene al realizar la segunda a diferencia de los casos anteriores donde empeora.

5.6 ML con algoritmo de aproximación de la inversa por una matriz de covarianza

Por simplicidad en la notación denotamos el método como CV. Para el mismo dimensionamiento utilizado en 5.2 pero utilizando el método presentado en [4] para el cálculo de la inversa los resultados obtenidos se muestran en las figuras 5-21 a 5-24 y la tabla 5-4:

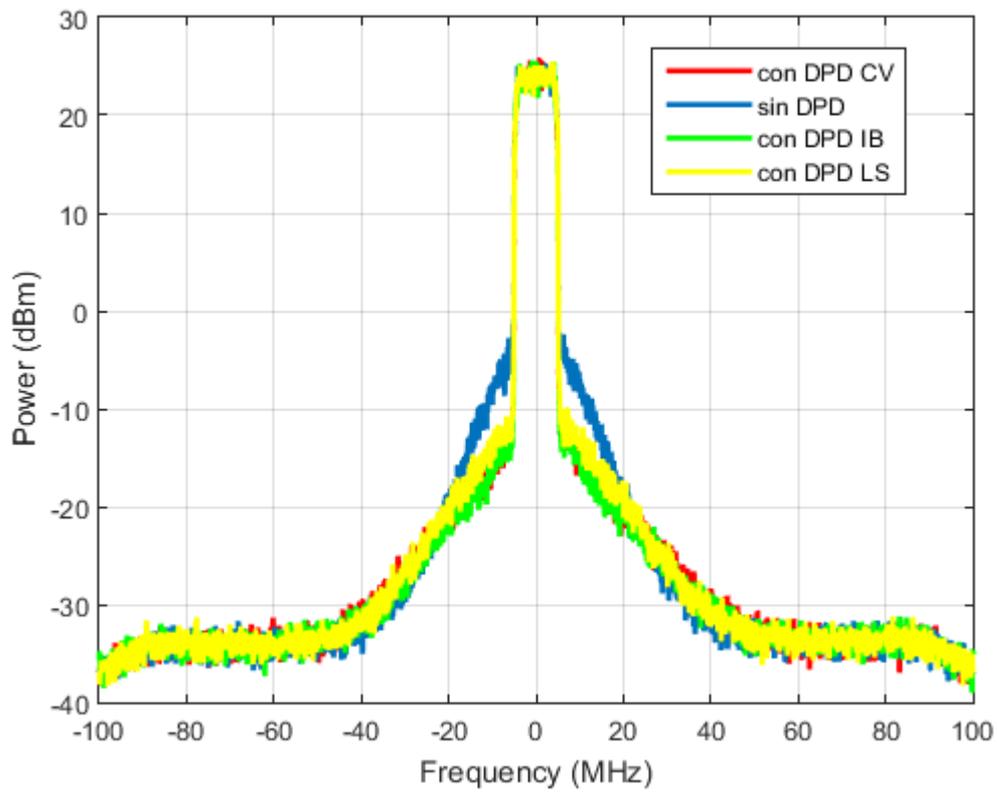


Figura 5-21. Comparativa de los espectros de salida con diferentes algoritmos para el modelo ML.

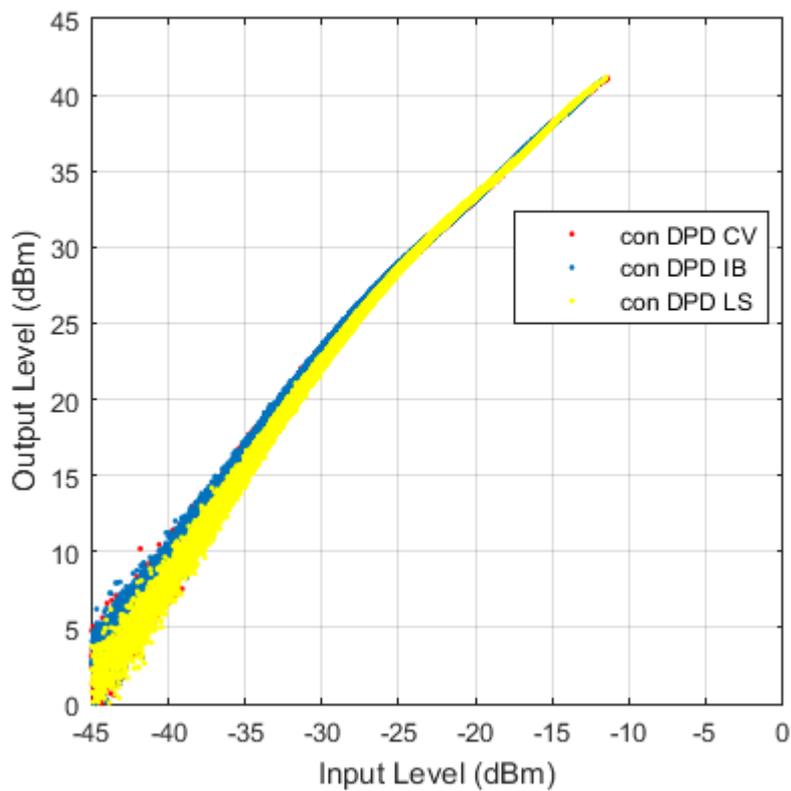


Figura 5-22. Comparativa de la característica AM/AM para los diferentes algoritmos para el modelo ML.

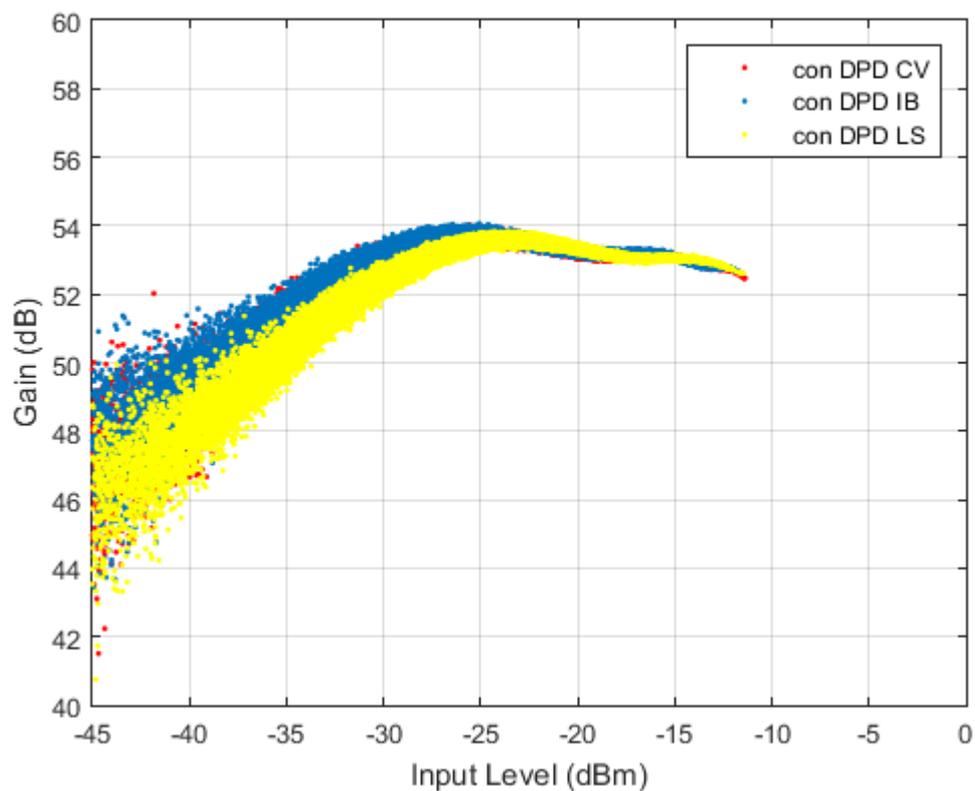


Figura 5-23. Comparativa de la ganancia frente a la potencia de entrada de los distintos algoritmos para el modelo ML.

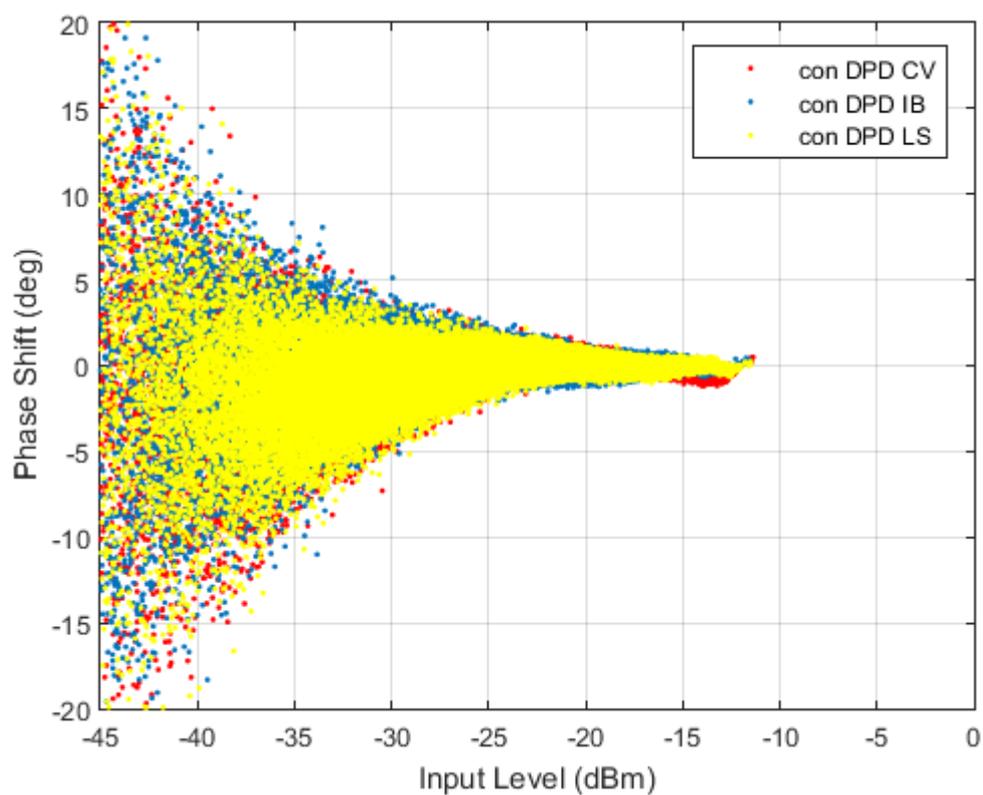


Figura 5-24. Comparativa de la característica AM/PM de los distintos algoritmos para el modelo ML.

Tabla 5-4. Valores ACPR y NMSE con ML algoritmo CV

Casos	ACPR (dB)	NMSE (dB)
Sin DPD	31.5	-23.6
Con DPD	37.2	-30.0
DPD 1 iteración	31.9	-23.6
DPD 2 iteraciones	38.3	-30.5

La tabla 5-4 representa un comportamiento distinto al analizado anteriormente. Aunque la mejora de los parámetros sigue sin ser progresiva (vemos que los valores empeoran tras realizar la primera iteración con respecto a los iniciales obtenidos mediante la aplicación del DPD), en este caso si se obtienen los mejores valores de ACPR y NMSE tras la última iteración.

5.7 MP con algoritmo de aproximación de la inversa por una matriz de covarianza

Manteniendo la profundidad de memoria y el número de muestras utilizado para modelar las señales de entrada y salida de los apartados anteriores, los resultados obtenidos al reemplazar el algoritmo LS por el presentado en [4] se muestran en las figuras 5-25 a 5-28 y la tabla 5-5:

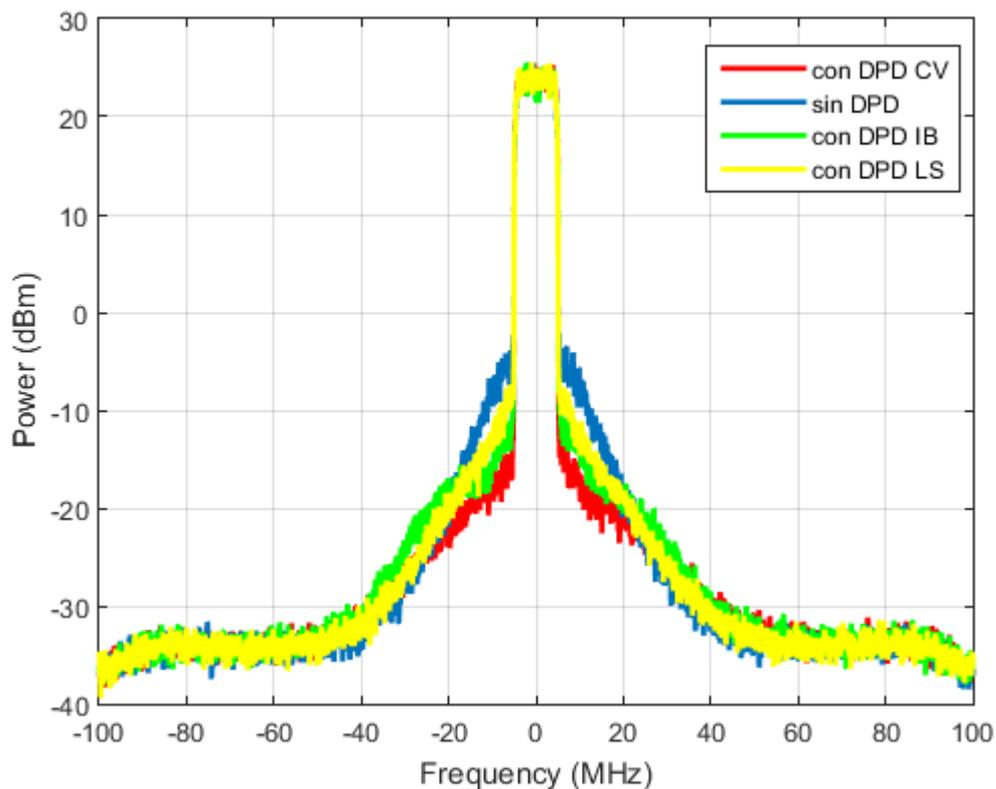


Figura 5-25. Comparativa de los espectros de salida para los distintos algoritmos en el modelo MP.

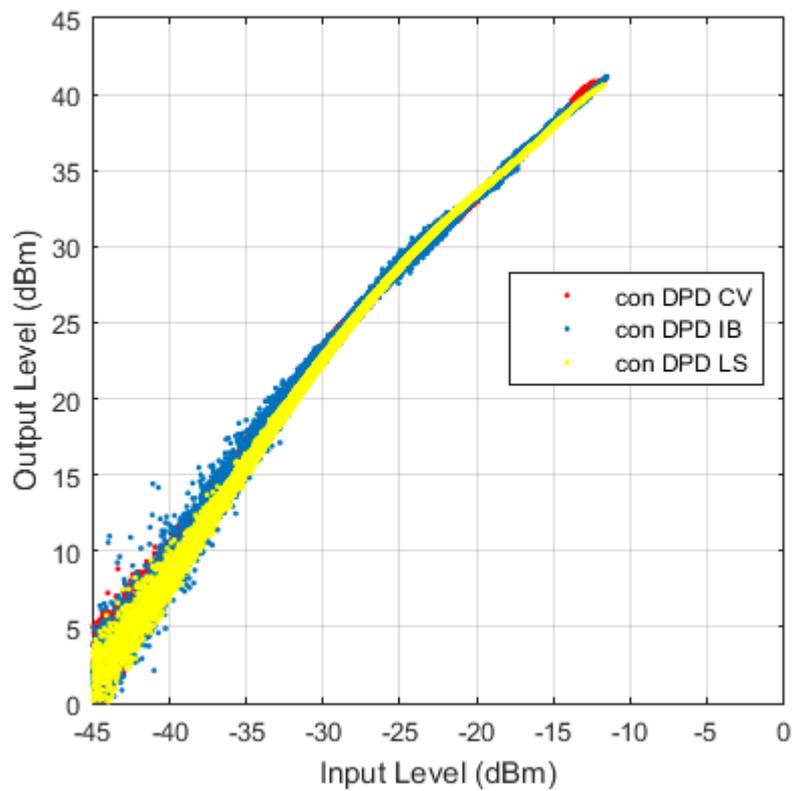


Figura 5-26. Característica AM/AM para los distintos algoritmos del modelo MP.

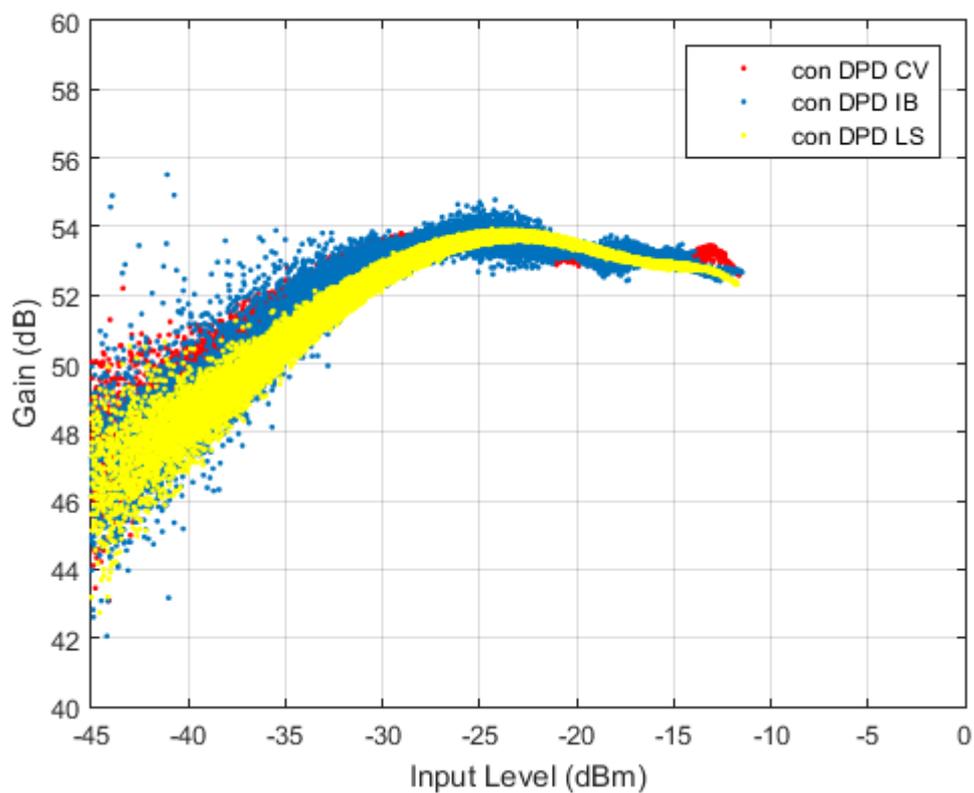


Figura 5-27. Ganancia frente a potencia de entrada para los distintos algoritmos del modelo MP.

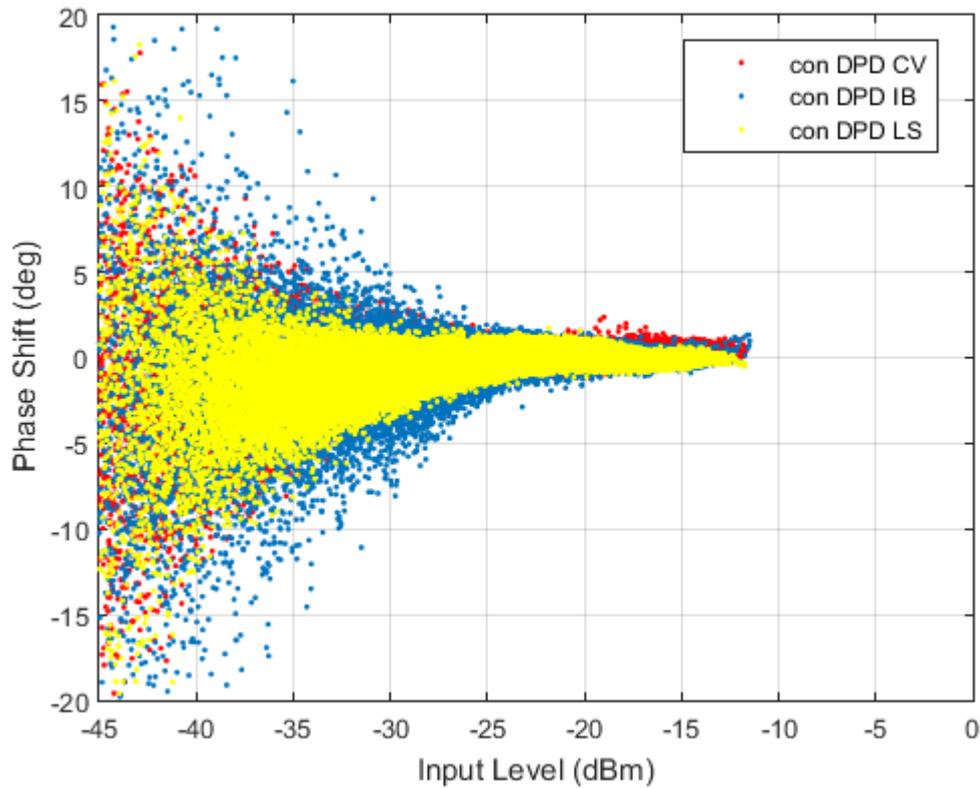


Figura 2-28. Característica AM/PM para los distintos algoritmos del modelo MP.

Tabla 5-5. Valores ACPR y NMSE con MP algoritmo CV

Casos	ACPR (dB)	NMSE (dB)
Sin DPD	31.5	-23.9
Con DPD	35.1	-25.8
DPD 1 iteración	40.1	-31.7
DPD 2 iteraciones	40.8	-32.4

El modelo MP con algoritmo de aproximación de la inversa por una matriz de covarianza obtiene los mejores resultados en cuanto a NMSE y ACPR de todos los casos analizados. Como podemos observar en la tabla 5-5 el modelo presenta una mejora progresiva de estos valores con cada iteración, obteniéndose el mejor de ellos tras la segunda iteración.

6 CONCLUSIONES Y LÍNEAS FUTURAS DE INVESTIGACIÓN

6.1 Conclusiones

El presente Trabajo surgió con el objetivo de linealizar uno de los dispositivos no lineales más importantes y usados en los sistemas de comunicaciones actuales como es el PA. Inicialmente se han revisado varias técnicas de linealización como son el Feedback, Feedforward, LINC, EE&R, CALLUM y la predistorsión digital. Siendo esta última, predistorsión digital, en la que hemos profundizado.

Los efectos de las no linealidades son nocivos tanto dentro como fuera de banda por ello se han implementado modelos que realizan una predistorsión digital sobre la señal y no sobre los datos, teniendo en cuenta de este modo la distorsión fuera de banda. Algunos de los principales efectos son:

- Distorsión fuera de banda: Ensanchamiento del espectro introduciéndose en los canales adyacentes.
- Distorsión dentro de banda: Se produce una distorsión en la constelación de la señal de salida. El valor del parámetro EVM cuantifica esta distorsión.

Por tanto, para evitar los efectos anteriormente citados se han implementado dos tipos de DPD basados en series de Volterra. El primero de ellos está basado en el modelo sin memoria (ML) y el segundo en el Memory Polynomial (MP), que sí tiene en cuenta los efectos de memoria. Estos efectos asociados a la memoria provocan una distorsión tanto en frecuencia como en magnitud, haciendo que las curvas AM/AM y AM/PM dejen de ser funciones estáticas y pasen a ser dependientes de la frecuencia.

El cálculo de los coeficientes que modelan al DPD es otro de los aspectos analizados en este Trabajo. Como se analizó en el capítulo 3, el algoritmo de estimación de los coeficientes basado en LS realiza la operación de inversión de matrices, lo cual conlleva un gran coste computacional. Además, como se ha señalado en capítulos anteriores la matriz del sistema por normal general está mal condicionada ya que existe cierta correlación entre los regresores que la componen, este mal condicionamiento provoca la propagación de errores. Debido a este motivo se han propuesto dos métodos de estimación alternativos que estiman el valor de los coeficientes sin necesidad de realizar la operación de inversión de matrices.

Analizando los resultados obtenidos en el capítulo 5, se puede observar que en el caso del DPD basado en el modelo sin memoria (ML) los mejores valores, en cuanto a NMSE y ACPR se refiere, se obtienen con el algoritmo de cálculo de la inversa por bloques. En el caso del DPD basado en el modelo Memory Polynomial (MP) los mejores resultados son obtenidos con el algoritmo de aproximación de la inversa por una matriz de covarianza, siendo estos a su vez los mejores de todas las pruebas realizadas.

Del análisis anterior se pueden extraer dos conclusiones. La primera de ellas es que los algoritmos alternativos presentados en [3] y [4] no solo son más eficientes computacionalmente, sino que también proporcionan mejores

resultados sin la necesidad de realizar inversión de matrices. La segunda conclusión es que el hecho de tener en cuenta los efectos de memoria en el modelado del DPD juega un papel importante en los resultados obtenidos por la aplicación de éste. Como se muestra en el capítulo 5 los mejores resultados se obtienen con el DPD basado en el modelo MP.

6.2 Líneas futuras de investigación

En un futuro sería recomendable probar los DPD implementados en este trabajo con otras señales de entrada para probar su validez y poder comparar los resultados obtenidos. Además, en el caso del MP se podría probar a aumentar la profundidad de la memoria para ver si con ello se obtiene una mejora de los resultados a cambio de incrementar la complejidad computacional.

Por otro lado, los códigos desarrollados tanto para la implementación de los DPD como para la implementación de los métodos de cálculo de los coeficientes podrían ser depurados con el fin de hacerlos más eficientes en términos de carga computacional.

REFERENCIAS

- [1] R. Negra, T.D. Chu, M. Helaoui, S. Boumaiza, G.M. Hegazi, K. Ghannouchi, "Switch-based GaN HEMT model suitable for highly-efficient RF power amplifier design," *IEEE/MTT-S 2007 International Microwave Symposium Digest*, pp. 795 -798, 2007.
- [2] J. C. Pedro, S.A. Maas, "A comparative overview of microwave and wireless poweramplifier behavioral modeling approaches," *IEEE Transactions on Microwave Theory and Techniques*, vol. 53, no. 4, pp. 1150-1163, 2005.
- [3] J. Peng, S. He, Z. Dai, B. Wang, "A simplified sparse parameter identification algorithm suitable for power amplifier behavioral modeling," *IEEE Microwave and Wireless Components Letters*, vol. 27, no. 3, pp. 290-292, 2017.
- [4] Z. Wang, W. Chen, G. Su, F. M. Ghannouchi, Z. Feng, Y. Liu, "Low computational complexity digital predistortion based on direct learning with covariance matrix," *IEEE Transactions on Microwave Theory and Techniques*, vol. 65, no. 11, pp. 4274-4284, 2017.
- [5] M. S. Vázquez Rodríguez, "Implementación en DSP de un predistorsionador digital para la linealización de amplificadores de potencia," Proyecto Fin de Carrera, Universidad Politécnica de Cataluña, 2006.
- [6] A. Katz, J. Wood, D. Chokola, "The evolution of PA linearization: from classic feedforward and feedback through analog and digital predistortion," *IEEE Microwave Magazine*, vol. 17, no. 2, pp. 32-40, 2016.
- [7] A. Zozaya, "Aportación a la linealización de amplificadores de potencia mediante la teoría de la hiperestabilidad," Tesis Doctoral, Departamento de Teoría de la Señal y Comunicaciones, Universidad Politécnica de Cataluña, 2002.
- [8] D. K. Su, W. J. McFarland, "An IC for linearizing RF power amplifiers using envelope elimination and restoration," *IEEE Journal of Solid-State Circuits*, vol. 33, no. 12, pp. 2252-2258, 1998.
- [9] J. M. Mozos Ruiz, "Técnicas de reducción del modelo para linealización de amplificadores de potencia mediante DPD," Trabajo Fin de Grado. Universidad Politécnica de Cataluña, 2016.
- [10] D. R. Morgan, Z. Ma, J. Kim, M. G. Zierdt, J. Pastalan, "A generalized memory polynomial model for digital predistortion of RF power pmplifiers," *IEEE Transactions on Signal Processing*, vol. 54, no. 10, pp. 3852-3860, 2006.
- [11] D. Hall, M. Anderson, "Understanding RF & microwave specifications – Part 2", National Instruments, 2007.
- [12] <http://www.dpdcompetition.com/> , último acceso: febrero de 2018.

