Multi Component Carrier, Sub-Band DPD and GNURadio Implementation

Chance Tarver[1], Mahmoud Abdelaziz[2], Lauri Anttila[2], and Joseph R. Cavallaro[1]
[1]Rice University, Department of Electrical and Computer Engineering, Houston, TX
[2]Tampere University of Technology, Department of Electronics and Communications Engineering, Tampere, Finland

***Abstract*—Digital predistortion (DPD) is an effective way of mitigating spurious emission violations without the need of a significant backoff in the transmitter, thus providing better power efficiency and network coverage. In this paper, the IM3 sub-band DPD, proposed earlier by the authors, is extended to more than two component carriers (CCs) through a sequential learning solution. The DPD learning is iterated over each spurious emission generated by each pair and trio of CCs. We train and apply the DPD coefficients for the intermodulation distortion (IMD) products until a satisfactory performance is achieved. The algorithm is tested in simulations using MATLAB and in a novel, real-time implementation on a CPU via a software version of the algorithm using GNURadio.**

***Keywords*—Adaptive filters, carrier aggregation, digital predistortion, nonlinear distortion, power amplifier, software-defined radio, spectrally-agile radio, spurious emission, GNURadio.**

I. INTRODUCTION

Digital predistortion (DPD) is an attractive solution for transmitter linearization, and has received substantial attention in the past 20 years (see [1] and the references therein). The vast majority of the previous works on DPD has concentrated on single-carrier transmissions. However, noncontiguous spectral allocations are gradually becoming common in many wireless communication standards such as 3GPP LTE-A and the 802.11 family. In scenarios where a single power amplifier (PA) is used to amplify several component carriers (CCs), two challenges arise for the DPD design: (1) The introduction of intermodulation distortion (IMD) between the noncontiguous CCs which may lead to a violation of the transmitter emission masks [2], [3]. This is especially true in multi-carrier modulations where the peak-to-average power ratio (PAPR) is high such as in OFDMA [1]. And (2) the increased DPD parameter estimation and filtering complexity, especially when the spacing between the CCs is large. Consequently, the DPD complexity and power consumption quickly rise, and the additional complexity due to DPD may become unjustifiable, especially for mobile devices.

Although the large majority of the DPD work that has been done only considers single-carrier transmissions, the work that does exist on DPD for multiple CCs is mostly limited to two carriers with the exception of [4], [5]. This is no longer sufficient as the 3GPP LTE-Advanced standard allows for 5 CCs [6] and modern system on chips such as the recently released Qualcomm Snapdragon 821 support 3 CCs for the downlink signal [7]. In [5], a fast ADC is needed to capture and linearize the entire band, and in [4], they are able to use a spectral extrapolation technique to use a slow ADC. However, both works face a similar issue in that the processing and application of their DPD is still complex.

The authors introduced a novel, sub-band DPD that used a block-based method for real systems and implemented it on an FPGA in [8] demonstrating real-time processing of the adaptive DPD learning solution. Moreover, in [9], higher nonlinearity orders were introduced in addition to the third-order nonlinearity processing in [8]. In [10], the authors did an iterative method that allowed for the suppression of multiple spurs that also accounted for the mutual effect of one spur on another. In this paper, we now introduce a new, sub-band system for scenarios when there exists more than two CCs.

When broadcasting more than two CCs, especially when one or more are noncontiguous, the output spectrum quickly begins to appear complicated with multiple spurs appearing in what may seem like odd shapes. However, this composite PA output can be decomposed into individual spurs created by intermodulation between each pair and trio of CCs as shown in Figure 1. By using superposition, the analysis of the scenario and the DPD algorithm can be simplified in that it allows us to use our previous sub-band algorithm for DPD learning and IMD suppression.

Much of the previous DPD work for carrier aggregation exists in simulation or in a limited implementation. In [8], an FPGA implementation was created by the authors, but it lacked the flexibility to easily accommodate all possible carrier aggregation (CA) scenarios and DPD processing. Other recent implementations include GPU [11]. For a frequency-agile, flexible system, a software solution may be desirable. This is especially attractive for emerging CloudRAN technologies where much of the baseband processing will be computed at a central server [12] and for cognitive radio where the transmission scenario may be rapidly changing. To target these scenarios the authors develop a software-based, sub-band DPD solution for DPD training and application using GNURadio in this paper.

In summary, the main novelties of this paper include the introduction of a sequential version of the previously proposed sub-band DPD to account for more than 2 CCs

This work was supported by the Finnish Funding Agency for Technology and Innovation (Tekes) under the project "Future Small-Cell Networks using Reconfigurable Antennas (FUNERA)." This work was also supported in part by the US National Science Foundation under grants ECCS-1408370, ECCS-1232274, and CNS-1265332 in the WiFiUS program. The work was also funded by the Academy of Finland under the projects 288670 "Massive MIMO: Advanced Antennas, Systems and Signal Processing at mm-Waves," 284694 "Fundamentals of Ultra Dense 5G Networks with Application to Machine Type Communication," and 301820 "Competitive Funding to Strengthen University Research Proles," and by the Linz Center of Mechatronics (LCM) in the framework of the Austrian COMET-K2 programme.

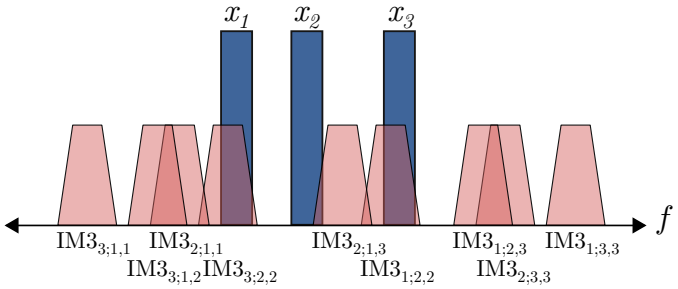

Fig. 1: Example power spectral density of a carrier aggregated signal with three CCs after being applied through a nonlinear PA. The locations of the CCs in this example are arbitrary, and only IM3 spurious emissions are presented as they are the main focus of this work. As can be seen, there are multiple IM3 spurs present in the nearby frequencies of the CCs which often overlap to varying extents. Here we ignore the intermodulations that are directly on the main carriers of the form $\text{IM3}_{i;i,k}$ and $\text{IM3}_{i;i,i}$ to focus on those in the spurious region.

and a GNURadio implementation of the sub-band DPD solution demonstrating effective real-time performance of the previously developed solution using real CPUs. We present a general, mathematical analysis followed by a variety of simulations of different possible scenarios on different platforms to highlight the versatility of our approach.

This paper is organized as follows. In Section II, the modeling of the spurious emissions (spurs) at the third-order intermodulation distortion (IM3) sub-bands is presented. A simulation of an actual LTE CA scenario where the sub-band DPD can be applied is presented in Section III. In Section IV, the GNURadio implementation is presented. In Section V, the paper is concluded.

II. Spurious Component Modeling and Sequential IM3 Sub-band DPD Processing

To illustrate the presence of the spurs, a mathematical model is presented. Unlike previous work, here we consider an arbitrary number of CCs being sent through a nonlinear PA. This analysis will provide a theoretical foundation and motivation for our work. For simplicity of the presentation, we restrict our analysis in this section to the third-order, PA nonlinearity and the corresponding IM3 spurs. However, this analysis can generalize to account for memory effects, higher order nonlinearities, and higher order intermodulation spurs by following the analysis presented in [9].

The model is developed at the composite baseband equivalent level using a memoryless, parallel-Hammerstein PA. We model the i_{th} component carrier as the baseband centered signal $x_i(n)$ that will exist at frequency f_i in the composite baseband signal. Thus, the composite baseband equivalent PA input and output signals, $x(n)$ and $y(n)$, read

$$x(n) = \sum_{i=1}^{N} x_i(n) e^{2\pi j n \frac{f_i}{f_s}}, \tag{1}$$

$$y(n) = \beta_1 x(n) + \beta_3 |x(n)|^2 x(n), \tag{2}$$

where β_1 and β_3 are unknown complex coefficients, f_s is the system sampling frequency, and N is the total number of component carriers being transmitted. Through direct substitution of (1) in (2), the baseband equivalent IM3 terms and their frequencies, where i, j, and k each go from 1 to N, are

$$y_{IM3:i;j,k}(n) = x_i^*(n) x_j(n) x_k(n), \tag{3}$$

$$f_{IM3:i;j,k} = -f_i + f_j + f_k. \tag{4}$$

The idea in [8], for suppressing the IMD at an IM3 sub-band for example, is to inject a proper additional low-power cancellation signal to (1), located at f_{IM3}, such that the spurious emission at the IM3 sub-band of the PA output is suppressed. Stemming from the signal structure in (3), the injection signals are of the form $x_i^*(n)x_j(n)x_k(n)$ but should be scaled properly with a complex DPD coefficient denoted here by α. Thus, incorporating such DPD processing, the composite baseband equivalent PA input signal now reads

$$\begin{aligned}\tilde{x}(n) = &\sum_{i=1}^{N} x_i(n) e^{2\pi j n \frac{f_i}{f_s} n} \\ &+ \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{k=1}^{N} \alpha_{i;j,k}\; x_i^*(n) x_j(n) x_k(n) e^{2\pi j n \frac{-f_i+f_j+f_k}{f_s}}.\end{aligned} \tag{5}$$

The above analysis accounts for all of the IM3 terms created. The idea can be extended to account for additional terms created by higher-order nonlinearities. However, it is not always necessary to account for higher-order (or even all of the third-order) terms. Depending on the RF frontend architecture and the relative location of the CCs in their band, some of these IMD terms may fall in the stopband of a TX band-pass filter. In these cases, the authors propose focusing the algorithm to only target the IMD spurs that cause a violation in the spurious emission requirements. By focusing our efforts on only the problematic spurs, we can reduce complexity in the DPD training and application.

This superposition method of suppressing multiple spurs relies on training to learn each of these α coefficients. This is done by the adaptive LMS method that has been shown in [8]–[10]. As can be seen in Figure 1, some of the IM3 terms fall in the in-band region near the component carriers. Although we focus on the spurious region in this paper, the same methods could be applied to reduce in-band emissions by targeting the corresponding spurs.

III. Simulations

We test the algorithm by implementing a practical LTE carrier aggregation scenario in MATLAB. For this, we use an LTE downlink scenario with three, 20MHz component carriers that match scenario CA_41C-41A. For this, the spurious region begins 25 MHz to the right of the highest carrier and the emission limit is -30 dBm [6]. In our simulation, we assume the two contiguous carriers are oriented to the lower portion of the band and the non-contiguous carrier is separated from the other carriers by 30 MHz. We also assume a TX band pass filter on the output of the PA. We use a ninth-order, memoryless PA model developed from a true mobile PA with a transmit power of +23 dBm in [9].

In this scenario, there is only a violation of the spurious emission requirements for the spurious region to the right of

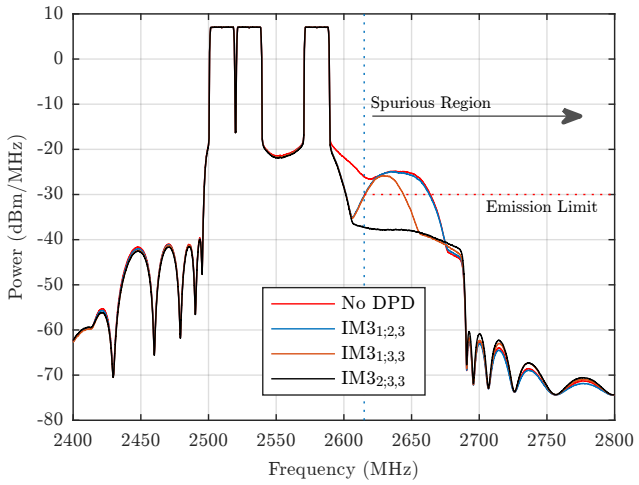

Fig. 2: Simulation of a CA scenario with three CCs. We sequentially suppress additional IM3 spurs until we suppress the three right-most spurs. By suppressing these three, the PA output power in the spurious region falls by approximately 14 dB so that it is below the emission limit.

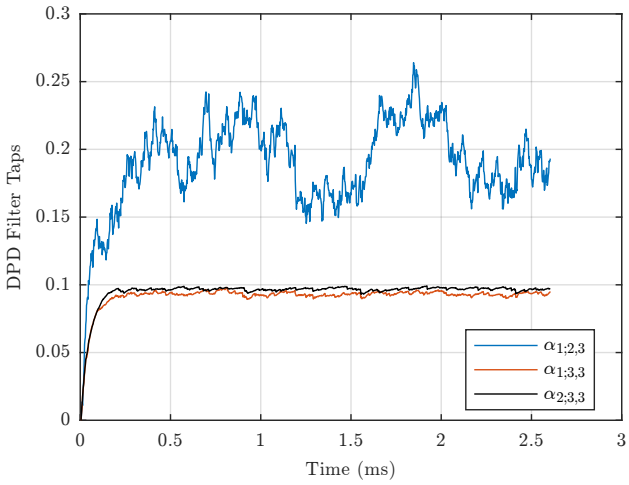

Fig. 3: Convergence of the absolute value of the complex DPD coefficients $\alpha_{1;2,3}$, $\alpha_{+1;3,3}$, and $\alpha_{2;3,3}$ during DPD training. Here, the $\alpha_{1;2,3}$ has a greater mean-square error because it is located in close proximity to CC3. As seen in Figure 2, it still contributes approximately 10 dB of suppression at its corresponding spur. The other coefficients converge smoothly.

the highest frequency CC, so we choose to do training to combat only the contributing spurs. These spurious emissions are from the intermodulation of the first component carrier with the third, the second with third, and the combined intermodulation of all three. We begin by training for $\alpha_{1;2,3}$. The training of this coefficient is shown as the blue curve in Figure 3. If we attempt to use this final, converged coefficient to suppress the $IM3_{1;2,3}$ spur, there is some suppression as shown by the blue line in Figure 2. We then train for the intermodulation between the first and third CC to learn $\alpha_{1;3,3}$. The combined effect of applying both coefficients is shown by the orange plot in the same figure. Training is performed for the final intermodulation product in the region, IM3$_{2;3,3}$ to get $\alpha_{2;3,3}$. This is applied with the other sub-band processing to get the final, black curve in Figure 2. Now the output is suppressed to be below the spurious emission requirement so we can stop training. The learning process of the coefficients is shown in Figure 3. This learning is done individually in a sequential manner, but is shown combined in this figure. The total time learning in this simulation is approximately 7.5 ms. The learning occurs quickly and can be sped up additionally by applying the techniques from [10].

IV. GNURADIO IMPLEMENTATION

DPD has been implemented in various technologies such as GPU [11] and FPGA [8]. A disadvantage that is common to these include a time consuming development and/or a lack of flexibility in the deployed solution. Using CPUs to do the DPD training and application is an alternative that is relatively easy to develop for and provides for incredible flexibility. One potential possibility for such system is in the upcoming CloudRAN systems [12]. From a central server, the network provider could perform DPD training whenever necessary for basestation RF equipment and then apply the DPD in their baseband processing. Moreover, such a system could be desirable in cognitive radio applications. Here, with this software implementation, we are easily and dynamically able to train and apply DPD even as the number and placement of CCs changes.

We developed an implementation of the sub-band DPD solution for multiple CCs using GNURadio. GNURadio is an open source SDR framework that operates in realtime. It provides many blocks for common SDR related functions and a scheduler that handles the dataflow in an efficient, parallel manner [13]. The blocks are implemented using C++ for performance, and they are connected using Python to facilitate ease of development. In our DPD flowgraph, we use many of the preexisting blocks but specially develop a LMS block for performing the DPD training, a frequency-shift block for shifting signals in the frequency domain, and a parallel-Hammerstein PA block for simulating the flowgraph.

In the flowgraph, component carriers are loaded from a binary file that was previously generated in MATLAB. They are each multiplied by a value from our frequency-shift block so that they are spaced as desired in the frequency domain. They are then added together and passed to the PA block for simulating a PA output. We simultaneously generate basis functions by connecting blocks to emulate Equation 3. The PA output is shifted to center the spur we wish to suppress. This shifted signal is passed through a low-pass filter to isolate the spur. We pass this and the corresponding basis function into our LMS block. Here, we generate the corresponding α. This is then multiplied by the corresponding basis function and added to the PA input as per Equation 5. Over time, α converges as the correlation between the error signal and the basis function goes to zero. This has the effect of suppressing the corresponding spur.

We have also developed an insightful visualization tool in this framework. Figure 4 shows a screenshot of this during a simulation. The spectrum, convergence of the coefficients, and various parameters are shown to the user. From this tool, we can change the position and amplitude of the carriers, change the parameters of the PA model, and turn on/off additional

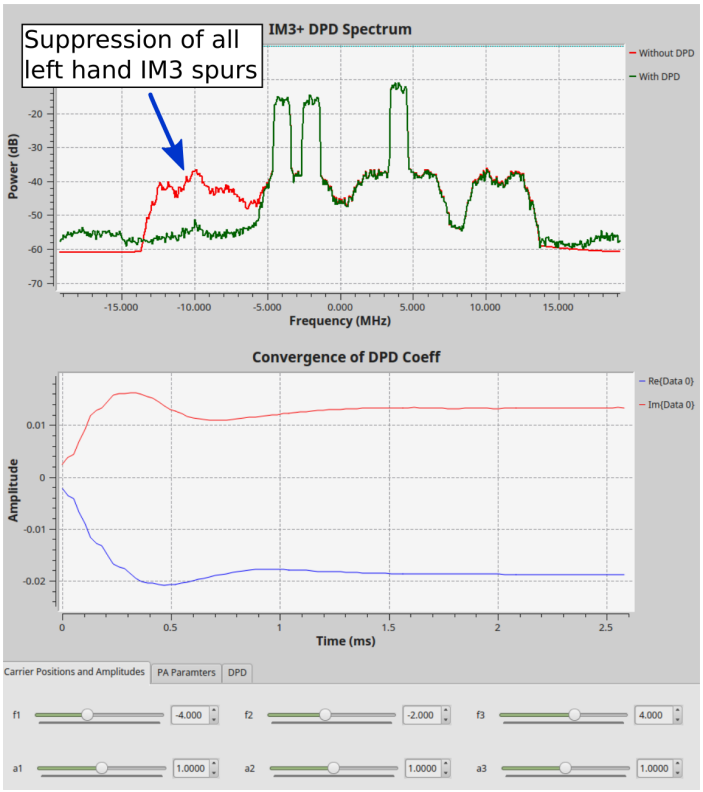

Fig. 4: GNURadio simulation of the sub-band DPD targeting the three left-hand IM3 spurs. The GNURadio implementation is flexible to where any or all spurs can be targeted depending on what is required. Here, an IM3 suppression of over 15 dB is achieved, while the DPD coefficient, α, converges smoothly and rapidly. The red curve in the first subplot is without DPD and the green is with DPD.

DPD processing. This allows a user to tweak the models and instantly see how the spectrum is effected. In the simulation shown, we take three, 1.4 MHz carriers separated by a few MHz and broadcast them through a nonlinear, simulated PA. A total of nine IM3 spurs are generated in the nearby frequency domain. As previously discussed, it is not always necessary to linearize the entire band. We imagine a scenario where we need to suppress only the left hand spurious emissions. From the tool, we can set it to train and apply DPD to only those spurs.

Our GNURadio DPD solution is capable of performing training on CPUs in realtime. Once training is complete, the application of the DPD to the baseband processing has a relatively low overhead. For example, with the memoryless, third-order case of suppressing a single IMD spur, the application of the DPD only consists of four additional complex multiplications and one addition per sample making it feasible to apply sub-band DPD on most SDR platforms. This complexity scales linearly with the number of spurs we linearize.

V. Conclusion

In this paper, we present extensions to previous work on sub-band DPD processing. Firstly, the algorithm is developed to handle more than two CCs by doing training for each pair and trio of CCs and then applying superposition. Secondly, we present a new software implementation that shows promise for rapid development and flexibility using GNURadio.

For future work, the authors would like to further develop the GNURadio implementation to handle higher order non-linearities and memory effects. The authors have previously proven the sub-band method on software defined radio platforms using real PAs. We hope to verify the case for more than two CCs in a similar manner.

References

[1] F. M. Ghannouchi and O. Hammi, "Behavioral Modeling and Predistortion," *IEEE Microw. Mag.*, pp. 52–64, Dec. 2009.

[2] Nokia, "R4-121205 Way forward for non-contiguous intraband transmitter aspects. Available at: http://www.3gpp.org/," tech. rep., 3GPP, Feb. 2013.

[3] T. Lähteensuo, "Linearity Requirements in LTE Advanced Mobile Transmitter.," Master's thesis, Tampere University of Technology, Tampere, Finland., May 2013.

[4] Y. Ma, Y. Yamao, Y. Akaiwa, and K. Ishibashi, "Wideband Digital Predistortion Using Spectral Extrapolation of Band-Limited Feedback Signal," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 61, pp. 2088–2097, July 2014.

[5] M. Younes, A. Kwan, M. Rawat, and F. M. Ghannouchi, "Linearization of Concurrent Tri-Band Transmitters Using 3-D Phase-Aligned Pruned Volterra Model," *IEEE Transactions on Microwave Theory and Techniques*, vol. 61, pp. 4569–4578, Dec 2013.

[6] *LTE; Evolved Universal Terrestrial Radio Access (E-UTRA) User Equipment (UE) radio transmission and reception, 3GPP TS 36.101 V13.2.1 (Release 13)*, May 2016.

[7] Qualcomm, "Snapdragon 821 processor specs." https://www.qualcomm.com/products/snapdragon/processors/821.

[8] M. Abdelaziz, C. Tarver, K. Li, L. Anttila, M. Valkama, and J. Cavallaro, "Sub-band digital predistortion for noncontiguous transmissions: Algorithm development and real-time prototype implementation," in *49th Asilomar Conf.*, Nov. 2015.

[9] M. Abdelaziz, L. Anttila, C. Tarver, K. Li, J. R. Cavallaro, and M. Valkama, "Low-Complexity Subband Digital Predistortion for Spurious Emission Suppression in Noncontiguous Spectrum Access," *IEEE TMTT*, vol. 64, pp. 3501–3517, 2016.

[10] C. Tarver, M. Abdelaziz, L. Anttila, M. Valkama, and J. R. Cavallaro, "Low-complexity, Sub-band DPD with Sequential Learning: Novel Algorithms and WARPLab Implementation," in *2016 IEEE Workshop on Signal Processing Systems (SiPS)*, 2016.

[11] K. Li, A. Ghazi, C. Tarver, J. Boutellier, M. Abdelaziz, L. Anttila, M. J. Juntti, M. Valkama, and J. R. Cavallaro, "Parallel digital predistortion design on mobile GPU and embedded multicore CPU for mobile transmitters," *CoRR*, vol. abs/1612.09001, 2016.

[12] A. Checko, H. L. Christiansen, Y. Yan, L. Scolari, G. Kardaras, M. S. Berger, and L. Dittmann, "Cloud RAN for Mobile Networks; A Technology Overview," *IEEE Communications Surveys Tutorials*, vol. 17, pp. 405–426, Firstquarter 2015.

[13] GNU Radio Website. http://www.gnuradio.org, 2016.