A Modular 1D-CNN Architecture for Real-time Digital Pre-distortion

De Silva, Udara; Ma, Rui; Koike-Akino, Toshiaki; Yamashita, Ao; Nakamizo, Hideyuki

TR2022-007   January 27, 2022

Abstract

This study reports a novel hardware-friendly modular architecture for implementing one dimensional convolutional neural network (1D-CNN) digital predistortion (DPD) technique to linearize RF power amplifier (PA) real-time. The modular nature of our design enables DPD system adaptation for variable resource and timing constraints. Our work also presents a co-simulation architecture to verify the DPD performance with an actual power amplifier hardware-in-the-loop. The experimental results with 100 MHz signals show that the proposed 1DCNN obtains superior performance compared with other neural network architectures for real-time DPD application.

IEEE Radio and Wireless Symposium (RWS)

A Modular 1D-CNN Architecture for Real-time Digital Pre-distortion

Udara De Silva*, Toshiaki Koike-Akino*, Rui Ma*, Ao Yamashita†, and Hideyuki Nakamizo†

*Mitsubishi Electric Research Labs, Cambridge, MA, USA, rma@merl.com

†Mitsubishi Electric Corporation, Information Tech. R&D Center, Kanagawa, Japan

Abstract—This study reports a novel hardware-friendly modular architecture for implementing one dimensional convolutional neural network (1D-CNN) digital predistortion (DPD) technique to linearize RF power amplifier (PA) real-time. The modular nature of our design enables DPD system adaptation for variable resource and timing constraints. Our work also presents a co-simulation architecture to verify the DPD performance with an actual power amplifier hardware-in-the-loop. The experimental results with 100 MHz signals show that the proposed 1D-CNN obtains superior performance compared with other neural network architectures for real-time DPD application.

Index Terms—DPD, power amplifier, CNN, neural network

I. Introduction

The fifth-generation (5G) new radio (NR) is designed to deliver enhanced mobile broadband with higher data rates up to 20 Gbps. The growth in data rates is made possible partly by carrier aggregation, which has increased the total supported bandwidth from 100 MHz used in long-term evolution advanced (LTE-A) to over 1 GHz in 5G-NR [1]. The increase in bandwidth leads to a significant challenge in linearization of power amplifier (PA) over a wide bandwidth. Digital pre-distortion (DPD) has been widely adopted to improve PA linearity [2]. The conventional DPD is mainly based on the Volterra series [3] or its simplified versions like memory polynomial [4]. Recently, researchers have introduced a deep neural network (DNN) framework for DPD operation [5]–[7]. Most DNN-based DPD systems are implemented using software models written using popular machine learning libraries in python. Although these models perform well on offline data, adopting them to real-time hardware designs such as FPGA is challenging due to timing and resource constraints. These challenges could be addressed by using either a radio-frequency (RF) analog accelerator [8] or a digital accelerator [9]. This paper presents a novel modular implementation of a one-dimensional convolutional neural network (1D-CNN) that can operate in real-time. The modular nature of the architecture increases the adaptability of the system to different neural network configurations. The synthesis results show that the presented DNN-DPD system can operate in greater than 100 MHz bandwidth which can be used to pre-distort wideband orthogonal frequency-division multiplexing (OFDM) signals used in 5G-NR.

II. DNN-based DPD

In practice, applying DNN to linearize RF power amplifier in radio transmitter demands special attention, due to the strict
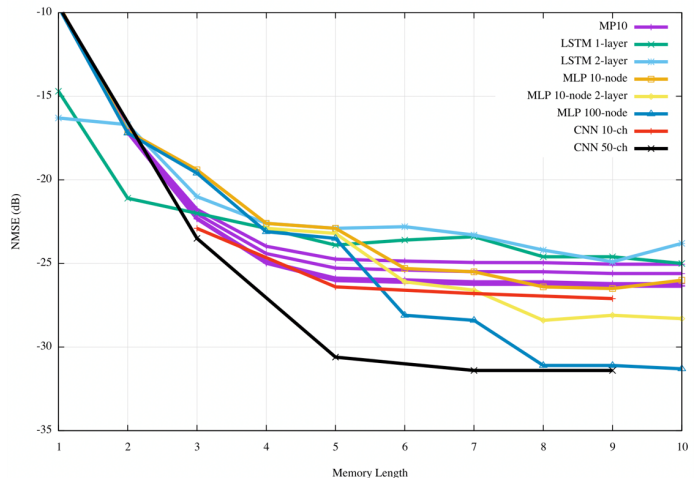

Fig. 1. NMSE performance of different NN architectures for PA modelling.

latency and resource constraints. Thus several modifications to conventional DNN should be made to adopt them to DPD. A commonly used simple approach for this is the real-valued time-delayed neural network (RVTDNN) [10]. It was further modified by the augmented real-valued time-delayed neural network (ARVTDNN) [5] which extends the input layer by adding pre-calculated envelope and its higher-order terms.

Convolution neural networks (CNNs) have been also investigated for DPD application [6], where the input data and its envelop terms are stacked together for two-dimensional CNN. Our simulation results in Fig. 1 shows that one-dimensional (1D) CNN can outperform several other configurations, including RVTDNN based on multi-layer perceptron (MLP), recurrent neural networks based on long short-term memory (LSTM), and conventional memory polynomial. Motivated by this result, we investigate a hardware-friendly 1D-CNN design for the DPD application.

III. Design of modular 1D-CNN architecture

Our architecture adopts the modularity and parametric design principles for the adoptability of our design to other DNN architectures. The overall 1D-CNN architecture is shown in Fig. 2. The ConvBlock is a parameterized implementation of the 1D convolution defined with a kernel size K, an input channel C_{in} and an output channel C_{out} for specification of kernel weights and biases. A system that converts an input
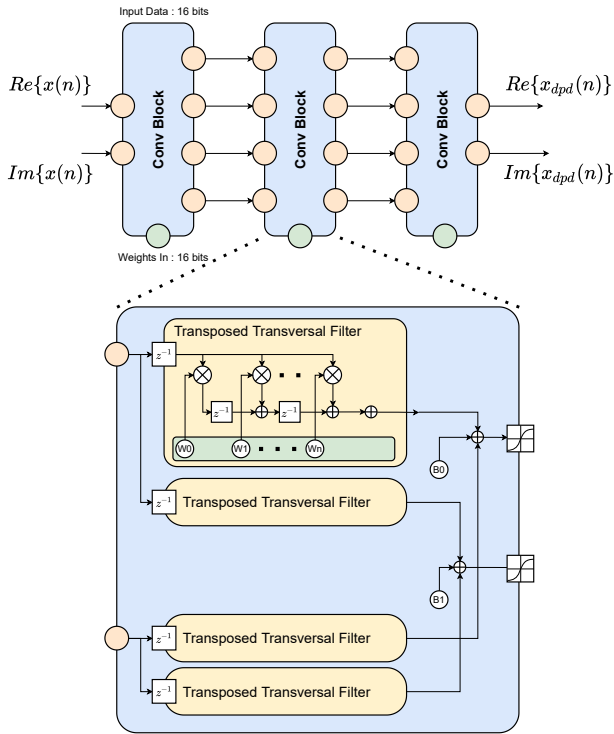
Fig. 2. System architecture of 1D-CNN processor.

Fig. 3. Verification architecture.

with C_{in} channels to an output with C_{out} channels should have $C_{\mathrm{in}} \times C_{\mathrm{out}}$ number of 1D filters. The design uses the 'generate' construct in Verilog to allow the design to support a user-specified number of input/output channels.

The 1D filters are implemented using the transposed finite-impulse response (FIR) filter structure [11] since it is known that the transposed form filters can outperform the direct form filters when the filters are large [12]. In the transposed FIR filters, multipliers can be avoided entirely if the coefficients are powers of two (PoT) [12] which leads to more hardware-friendly designs.

In order to update weights without modifying any design files, the weights are declared in a separate file and included in the Conv1d implementation. This design pattern requires resynthesizing every time the weights are updated. Alternatively, weights can be stored in registers which can be updated by a processor.

The design supports various nonlinear activation function such as sigmoid. Our experiment showed that rectified linear unit 6 (ReLU6) outperforms sigmoid activation in 1D CNN-based DPD systems. The ReLU6 implementation is also hardware-friendly compared to most other activation functions as it just needs two cooperators.

IV. VERIFICATION USING CO-SIMULATION

A co-simulation strategy is used to verify the design. The verification setup uses the ModelSim as the register-leverl transfer (RTL) simulator and Python to communicate with RFWebLab [13] and simulate the software-based 1D CNN.
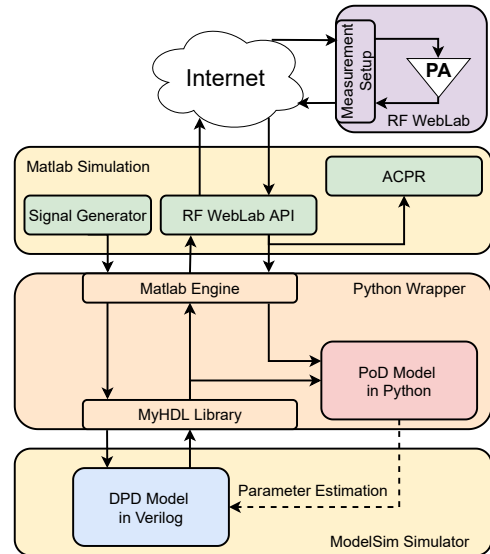
The complete verification architecture is shown in Fig. 3. MyHDL [14] and Matlab Engine are used through Python libraries that communicate with ModelSim and Matlab. OFDM signals are generated in MATLAB and provided as the input to the DPD system running in ModelSim simulator. The output is sent to RFWebLab, an online accessible testbed with a real PA. The output from PA is returned to MATLAB, to measure the improvement.

The DPD is trained using the indirect learning architecture (ILA) shown in Fig. 4(a). A post-distortion (PoD) model is used in ILA, which has the same architecture as the DPD model. The PoD model is trained using the PA output y and PA input x_{dpd}. The error is calculated by $e(n) = x_{\mathrm{pod}}(n) - x_{\mathrm{dpd}}(n)$. The PoD model is trained until the normalized mean squared error is minimized. Fig. 4(b) shows that PoD model output $x_{\mathrm{pod}}(n)$ closely predicts the PA input $x_{\mathrm{dpd}}(n)$ when the error is minimized. After the training, the parameters of the PoD model are passed on to the DPD model.

We use 50 MHz OFDM signals to test the DPD performance in terms of adjacent channel power ratio (ACPR) [15]. The experimental results showed that DPD improves the ACPR from -32.92 dB to -38.88 dB. The power spectral density (PSD) comparison with and without the DPD is also shown in Fig. 4(c).

The hardware design is synthesized targeting the Xilinx ZCU104 FPGA board. Table I shows the timing and area reports of the design synthesis for different configurations. All configurations use the fixed point number representation with 16 total bits and 10 fractional bits. The ACPR results shown in Fig. 4(c) is taken from the design with two hidden layers, 20 neurons per layer, and a kernel size of 5. The performance could be increased by increasing them at the cost of more hardware resources. The timing results in Table I show that our design meets the requirement to run at least 100 MHz for 50 MHz bandwidth.
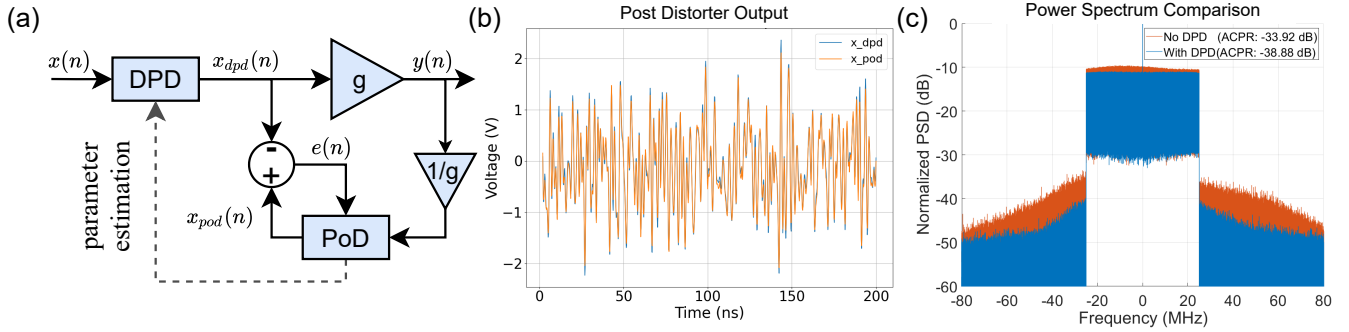
Fig. 4. DPD verification using co-simulation: (a) In-direct learning architecture; (b) Post distorter output ($x_{pod}(n)$ comparison with PA input ($x_{dpd}(n)$)) after training; (c) Power spectral density of the transmit signal without using DPD (shown in red) which has an ACPR of -33.92 dB and transmit signal after using DPD (shown in blue) which has an ACPR of -38.88 dB.

TABLE I
TIMING AND RESOURCE UTILIZATION OF HARDWARE DESIGN FOR
DIFFERENT NETWORK CONFIGURATIONS

Design			Timing	Area		
hidden layers	neurons per layer	kernel size	Critical path (ns)	LUTs	Regs	DSPs
1	20	5	1.541	~4k	~1k	400
1	20	9	1.541	~4k	~1k	400
2	20	5	3.218	~55k	~46k	914
1	40	5	1.541	~5k	~7k	510
2	40	5	4.040	~76k	~59k	1k

The synthesis is targeted for Xilinx MPSoC FPGA in ZCU104 board which has 230k LUTs, 460k Regs and 1.7k DSP blocks

V. CONCLUSION

This paper presents a hardware-friendly modular architecture of a 1D-CNN-based DPD system. Due to the modular nature and parametric design, this architecture is easily adaptable to different DNN configurations to accommodate different timing and resource constraints. The design is verified using a co-simulation strategy that measures the ACPR improvement for an FPGA. The performance of the DPD system can be approximated in significantly less time since it avoids the bit-stream generation time. The initial experimental results showed around a 5 dB reduction in ACPR for two-layer 1D-CNN with 20 neurons per hidden layer. This performance can be further improved by using a deeper/wider network configuration at the cost of increased resource utilization. A software-based post-distortion model is used to extract the parameters for the DPD model. In future work, we will investigate direct learning architecture to update the DPD model parameters on the fly. The current design uses DSP blocks in the FPGA to perform multiplication. In the future, we will investigate PoT approximations for the model parameters and the improvement in resource utilization by replacing DSP blocks with shift and addition operations.

REFERENCES

[1] H. Huang, J. Xia, and S. Boumaiza, "Parallel-processing-based digital predistortion architecture and FPGA implementation for wide-band 5G transmitters," in *2019 IEEE MTT-S International Microwave Conference on Hardware and Systems for 5G and Beyond (IMC-5G)*, 2019, pp. 1–3.

[2] L. Ding, "Digital predistortion of power amplifiers for wireless applications," Ph.D. dissertation, 2004, copyright - Database copyright ProQuest LLC; ProQuest does not claim copyright in the individual underlying works; Last updated - 2021-05-18. [Online]. Available: https://www.proquest.com/dissertations-theses/digital-predistortion-power-amplifiers-wireless/docview/305191006/se-2?accountid=10901

[3] C. Yu *et al.*, "Band-limited Volterra series-based digital predistortion for wideband RF power amplifiers," *IEEE Transactions on Microwave Theory and Techniques*, vol. 60, no. 12, pp. 4198–4208, 2012.

[4] D. Morgan *et al.*, "A generalized memory polynomial model for digital predistortion of RF power amplifiers," *IEEE Transactions on Signal Processing*, vol. 54, no. 10, pp. 3852–3860, 2006.

[5] D. Wang, M. Aziz, M. Helaoui, and F. M. Ghannouchi, "Augmented real-valued time-delay neural network for compensation of distortions and impairments in wireless transmitters," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, no. 1, pp. 242–254, 2019.

[6] X. Hu *et al.*, "Convolutional neural network for behavioral modeling and predistortion of wideband power amplifiers," *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–15, 2021.

[7] M. Tanio, N. Ishii, and N. Kamiya, "Efficient digital predistortion using sparse neural network," *IEEE Access*, vol. 8, pp. 117 841–117 852, 2020.

[8] U. De Silva, S. Mandal, A. Madanayake, J. Wei-Kocsis, and L. Belostotski, "RF-rate hybrid CNN accelerator based on analog-CMOS and Xilinx RFSoC," in *2020 IEEE International Symposium on Circuits and Systems (ISCAS)*, 2020, pp. 1–5.

[9] N. P. Jouppi *et al.*, "In-datacenter performance analysis of a tensor processing unit," in *2017 ACM/IEEE 44th Annual International Symposium on Computer Architecture (ISCA)*, June 2017, pp. 1–12.

[10] *Neural Network Based Models*. John Wiley & Sons, Ltd, 2015, ch. 7, pp. 133–152. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/9781119004424.ch7

[11] A. Oppenheim and R. Schafer, *Discrete-time Signal Processing*, ser. Prentice-Hall signal processing series. Prentice Hall, 1989. [Online]. Available: https://books.google.com/books?id=bPhSAAAAMAAJ

[12] "Transposed form FIR filters." [Online]. Available: http://ebook.pldworld.com/_semiconductors/XILINX/DataSource CD-ROM/Rev.8 (Q1-2003)/appnotes/xapp219.pdf

[13] "RF WebLab - online RF measurements." [Online]. Available: http://dpdcompetition.com/rfweblab/

[14] K. Jaic and M. C. Smith, "Enhancing hardware design flows with MyHDL," ser. FPGA'15. New York, NY, USA: Association for Computing Machinery, 2015, p. 28–31. [Online]. Available: https://doi.org/10.1145/2684746.2689092

[15] M. Rawat and F. M. Ghannouchi, "A mutual distortion and impairment compensator for wideband direct-conversion transmitters using neural networks," *IEEE Transactions on Broadcasting*, vol. 58, no. 2, pp. 168–177, 2012.