IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 28 April 2023, accepted 9 May 2023, date of publication 11 May 2023, date of current version 24 May 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3275434

RESEARCH ARTICLE

Physics Informed Spiking Neural Networks: Application to Digital Predistortion for Power Amplifier Linearization

SIQI WANG, PIETRO MARIS FERREIRA, (Senior Member, IEEE), AND AZIZ BENLARBI-DELAI

Sorbonne Université, CNRS, Lab. de Génie Electrique et Electronique de Paris, 75252 Paris, France
Université Paris-Saclay, CentraleSupélec, CNRS, Lab. de Génie Electrique et Electronique de Paris, 91192 Gif-sur-Yvette, France

Corresponding author: Pietro Maris Ferreira (maris@ieee.org)

ABSTRACT Recently, new emerging techniques of neuromorphic hardware render spiking neuron networks (SNN) promising as an energy-efficient solution for artificial intelligence (AI). With the idea of physics informed neural network, the structure can be simple while training data can be light. However its application in RF telecommunication system is still challenging. This paper, as the first time in the literatures, proposes a solution of SNN-based digital predistortion (SNN-DPD) for linearization of RF transmitters, such as power amplifiers (PA). A two-layer SNN is deployed in frequency domain to process the spectrum of the stimulus for a predistorted signal. The proposed technique is experimentally validated on a test bench with a real PA of different bias voltages. We also test the proposed SNN-DPD for multi-band linearization. The proposed method reaches the best performance of traditional DPD methods while owing advantages of the SNN, such as low power consumption and good biomimicry for AI.

INDEX TERMS Artificial intelligence (AI), digital predistortion, frequency domain, linearization, physics informed, power amplifiers, spiking neural networks.

I. INTRODUCTION

Along with the development of artificial intelligence (AI) in the past decades, different techniques based on neural networks have been used for PA linearization, such as artificial neural networks (ANN) [1], [2], [3], convolutional neural networks (CNN) [4], and support vector machine (SVM) [5]. These neuron-based techniques have been shown effective when both the number of neurons and the training dataset are large enough. In case where any knowledge of physical laws are known, the problems can be addressed more accurately with less training data thanks to physics informed neural networks (PINN) [6]. The concept of the PINN has also been generalized for different network structures, such as the CNN [7].

The spiking neural networks (SNN) have been considered as the third generation of neural network [8], which may have advantages on information processing of an AI system and on energy consumption. Different from conventional ANNs which process digital data, the SNN better mimics the biological behavior of the brain cortex which processes spike trains [9]. In other words, the event-driven SNN has more capacity to be inspired by the biological brain since its event-driven property helps to capture the rich dynamics of neurons within the brain. As the research on brain science is advancing rapidly, the process of inference and decision making of human brain can bring more and more inspiration to the AI.

Similar to the brain cortex, the neurons in SNN are excited by a current I_{ex} which brings an increase of membrane potential V_m. This waveform of V_m is in spike shape because it will be reset to a low potential V_r once it reaches a threshold V_s. The spike of V_m can be transmitted from one neuron to another through synapses. Not like neurons in ANN which are always kept active for data processing and memory access, a neuron in SNN shall be active only when it fires a spike. This event-driven property results in a giant reduction on computational consumption. The SNN is believed to have

The associate editor coordinating the review of this manuscript and approving it for publication was Junhua Li.

over 100 times higher efficiency on energy consumption than the ANN when implemented on a field-programmable gate array (FPGA) [10].

Moreover, compared with being developed on conventional computers and digital signal processing circuits [11], some recent studies found the SNN highly energy efficient on non-von-Neumann computing hardware [12], [13], [14]. With SNN-based neuromorphic chip hardware, an IBM platform TrueNorth [15] consumes only 10^{-5} of a conventional continuous-value network. Besides, Loihi [16] and Braindrop [17] also provide solutions of neuromorphic chips which save enormously energy compared with CPU-based processors.

In [18], an electronic neuron is designed with MOS transistors biased in weak inversion, which consumes around several hundred pico-Watt. With the technique of low power circuits design, the spiking neural networks exhibit a great interest on power efficiency compared with conventional continuous-value network of equivalent scale [19].

Power consumption has always been a crucial factor in wireless communication systems. As the development of modern 5G and beyond communication systems, the number of base stations goes up exponentially. Improving the power efficiency of the system becomes more and more important. Power amplifiers (PA) are ones of the most power consuming devices in the system. However, the energy efficiency of a PA is restricted by its nonlinearity [20].

The digital predistortion (DPD) is one of the most common approaches for PA linearization [21]. The distortion of a PA is mainly due to its nonlinearity and memory effects. In order to compensate for these distortions, various DPD models have been developed based on Volterra series, such as memory polynomial (MP) [22], generalized memory polynomial (GMP) [23], dynamic-deviation-reduction (DDR) model [24], and decomposed vector rotation (DVR) model [25]. Block-oriented non linear (BONL) systems [26] have also been studied. Some techniques of machine learning [27], [28] are recently found of interest in PA linearization. The DPD can be implemented for both time-domain (TD) data and frequency-domain (FD) data. In [29], an FD-DPD has been proposed to process only the subcarriers of a wideband signal, which largely reduces the number of processed samples and avoids the restriction of the high sampling rate. However, this FD-DPD has huge computational burden in constructing nonlinear basis functions since the multiplications in time domain is converted to convolutions in frequency domain. With this physical knowledge, applying the SNN under the framework of the PINN to address this problem is promising but challenging. Some studies on using the nonlinear behaviors of spiking neurons for off-line PA modeling have recently been reported [30].

In this paper, we explore to use the SNN technique on the DPD modeling. Since the SNN uses the frequency of its spike trains to present a constant value during a certain period, it is difficult to process the orthogonal frequency division multiplexing (OFDM) signal in time domain. Thus we present an SNN-based DPD (SNN-DPD) as a solution for both single-band and multi-band linearization, which processes the subcarriers of OFDM frames in frequency domain. Its input data is the spectrum of the stimulus magnitude, and its output data gives the spectrum of the magnitude of the predistorted signal. The SNN contains two layers of neurons, with the input layer of neurons excited by the encoded currents, and the second layer outputs a series of firing rates f_{spike} which can be decoded to corresponding subcarriers of the predistorted signal. To the best of authors' knowledge, this is the first study to employ the emerging technique of SNN in the PA linearization. In this paper, the proposed method is validated with experimental measurements on the testbench of Chalmers WebLab [31] for both single-band and multi-band cases, which confirms the effectiveness of its linearization performance. The Chalmers WebLab is accessible to global public via Internet so that researchers can fairly compare different techniques under similar conditions.

This paper is organized as follows. In Section II, the SNN and its training method are explained. The proposed SNN-DPD is presented in Section III. We analyze the complexity of classical techniques and the proposed technique in Section IV. In Section V, the test bench and the corresponding experimental results are presented and discussed. Finally, the conclusion is given in Section VI.

II. SNN AND NETWORK TRAINING

The SNN is inspired by the biological behavior of the brain cortex. The information in the SNN is transmitted and processed in the form of spike trains which are pulses of membrane voltage. In the cortex neuron, a biological current I_{ex} is generated by the passing of Sodium (Na^+) and Potassium (K^+) ions through the membrane [32]. Under the excitation of I_{ex}, the membrane voltage V_m keeps on increasing. The neuron fires a spike once V_m exceeds a given threshold V_T. However, it cannot go to infinite. After exceeding the threshold, V_m falls down quickly back to the rest value. This electric behavior in the neuron has been described by the Hodgkin-Huxley (HH) model [33] and Morris and Lecar (ML) model [34], which enables the design of an electronic neuron to repeat the procedure of firing spikes. Some simplified models have been proposed, such as Izhikevich model [35] and leaky integrate and fire (LIF) model [36].

The information processing of a neuron is illustrated in Fig. 1. A neuron generates spikes of membrane voltage V_m (1) when it is excited by a current I_{ex} as shown in Fig. 1(a); (2) or when a synapse connected to it is stimulated by a spike train as shown in Fig. 1(b). The output spike train from a neuron in Fig. 1(a) or (b) can have different frequencies, as shown in Fig. 1(c). This spike train can carry the information in different ways, such as by its frequency f_{spike}. A counter is connected to the neuron to read out f_{spike}. This frequency information can also be converted to the current information, if necessary, through a predefined look-up table (LUT) since the values of f_{spike} and I_{ex} are somehow one-to-one mapping.

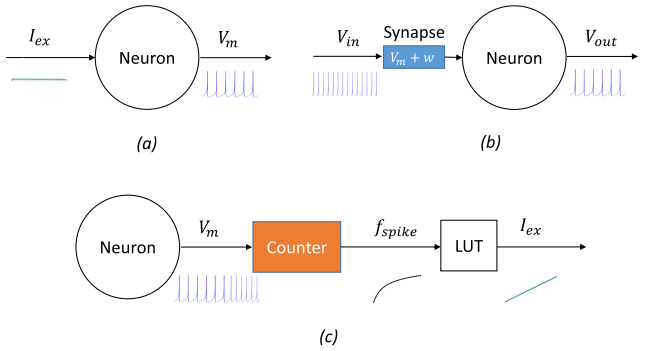

FIGURE 1. Scheme of a spiking neuron: (a) spiking neuron excited by a current; (b) spiking neuron receives a spike train as input; (c) readout of information from spike rates as neuron's output and transfer to current information if necessary.

Their relation depends on the realization model of the neuron circuit and the model parameters.

A. SPIKING NEURON MODELS

The variation of membrane voltage V_m as a function of excitation current I_{ex} can be described by HH model:

$$\begin{aligned}\frac{dV_m}{dt} &= \frac{1}{C_m}\big(I_{\text{ex}} - G_{Na}\cdot m^3\cdot h\cdot(V_m - E_{Na})\\ &\quad - G_K\cdot n^4\cdot(V_m - E_K) - G_L(V_m - E_L)\big)\\ \frac{dn}{dt} &= \alpha_n(1-n) - \beta_n n\\ \frac{dm}{dt} &= \alpha_m(1-m) - \beta_m m\\ \frac{dh}{dt} &= \alpha_h(1-h) - \beta_h h,\end{aligned} \tag{1}$$

where C_m is the membrane capacitance, G_{Na} and G_{K} are maximum conductances of channels for Na and K ions respectively, G_L is the leakage conductance, E_{Na} and E_K are Nernst potentials for Na and K ions respectively, E_L is the leakage potential, m, n, and h are activation coefficients, the variables (α_m, α_n, α_h, β_m, β_n, β_h) are functions of V_m.

We can see in this models that the differential equation of membrane voltage V_m shows its gradient is negatively proportional to its own value, which means V_m increases slowly when its value is large. However, under the excitation by a current I_{ex} which is large enough, the gradient of V_m is kept positive. The membrane voltage of a neuron keeps increasing till a threshold voltage and then is abruptly reset to the rest voltage. This procedure generates a spike. With the existence of I_{ex}, the neuron keeps on generating spike train with a frequency related to the value of I_{ex}.

Fast Spiking (FS) neurons are explored in [14] and [18] which fire high-frequency tonic spikes with relatively constant period, as well as the Low-Threshold Spiking (LTS) neurons which fire tonic spikes with pronounced spike frequency adaptation (decreasing) and rebound spikes due to post-inhibitory effect. In Fig. 2, we illustrate the I_{ex}-f_{spike} curves with 4 different datasets from the post-layout simulations of neurons redesigned as presented in [37]:

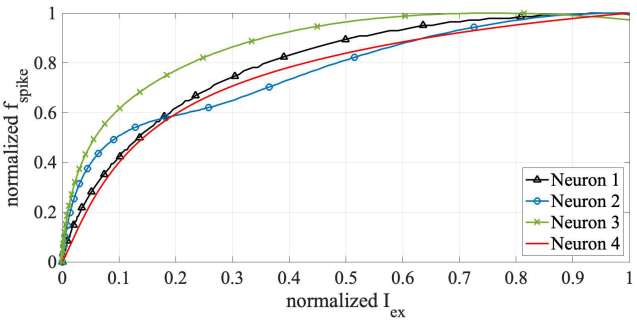

FIGURE 2. Normalized I_{ex} vs. f_{spike} curves of 4 neurons.

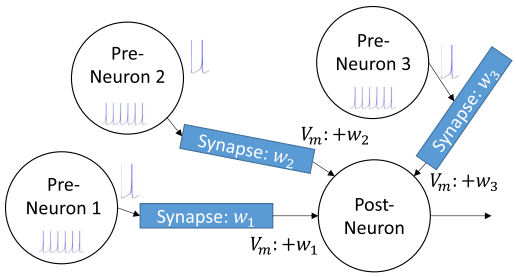

FIGURE 3. System scheme of two neurons connected by synapses.

1) Neuron 1: FS neuron with LIF model designed in [14].
2) Neuron 2: FS neuron with ML model designed in [14].
3) Neuron 3: FS neuron with ML model designed in [18].
4) Neuron 4: LTS neuron with ML model designed in [18].

For better visualization, in Fig. 2, the datasets of f_{spike} and I_{ex} are offset and normalized between the interval [0,1]. We can see that, though different types of neuron with different designs have different characteristics but with similar trends, the value of I_{ex} can always be determined if we know f_{spike}, and vice versa. In other words, with a constant input I_{ex}, a spiking neuron can produce a spike train with stable frequency. This enables the connection between the samples of signals and the SNN.

B. SYNAPSE MODELS

The neurons are connected with synapses which transmit the spikes of the pre-neuron to the post-neuron as illustrated in Fig. 3. The synapses are labeled with different weights w_i. The spikes fired by the pre-neurons are transmitted through synapses to the post-neuron. For a neuron of HH or ML model, the membrane voltage V_m is increased by w_i once it receives a spike from the synapses:

$$\begin{aligned}\frac{dV_m}{dt} &= \frac{I_{\text{ex}} - (G_{Na}\cdot m^3\cdot h\cdot(V_m - E_{Na})}{C_m}\\ &\quad - \frac{G_K\cdot n^4\cdot(V_m - E_K) + G_L(V_m - E_L)}{C_m}\\ &\quad + \sum_i w_i \sum_s \delta(t - t_{i,s}),\end{aligned} \tag{2}$$

where $t_{i,s}$ represents the s-th spiking time of the i-th pre-neuron.

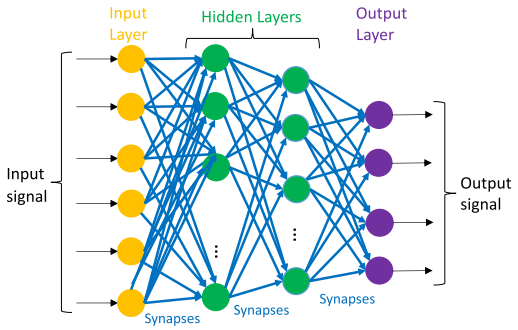

FIGURE 4. Structure of SNN with hidden layers.

A simple structure of an SNN is illustrated in Fig. 4. A network can be composed of at least 2 layers: the input layer and the output layer. Between these 2 layers, some hidden layers can be inserted for better variety of connections. Each layer consists of a certain number of neurons. The input signal can be some spike trains or some values of excitation current I_{ex}, and the output signal can be measured directly in form of spiking rate f_{spike} or their equivalent current I_{ex}.

C. NETWORK TRAINING BY STDP

An effective way to train the SNN is the spike-timing dependent plasticity (STDP) mechanism [38]. In the brain cortex, the strength of connections between neurons is dynamic and is adjusted by the STDP process. In the artificial synapses, we adopt the same process for the SNN training.

The strength of connections between neurons is represented by the weight of synapse w_i in (2). The main factor in the STDP is the spiking times of the pre-neuron and post-neuron on the two sides of the synapse. In case where the pre-neuron fires a spike before the post-neuron, we shall increase the weight because there is a causal connection between the pre-neuron spike and that of the post-neuron. In case where the pre-neuron fires a spike after the post-neuron, the causal connection does not exist anymore and we shall decrease the weight. The weight variation Δw_i depends on the spiking time difference Δt between the pre-neuron and the post-neuron. When their spiking times are close, their connection should be adjusted rapidly. Thus for the i-th synapse, we have

$$\Delta w_i = \sum_{s_{\text{pre}}} \sum_{s_{\text{post}}} F(\Delta t_{s_{\text{pre}}, s_{\text{post}}}), \tag{3}$$

where s_{pre} and s_{post} are the index of spike train fired by the pre-neuron and the post-neuron of the i-the synapse, respectively. The function $F(\Delta t)$ can be expressed as

$$F(\Delta t) = \begin{cases} A_{\text{pre}} e^{-\frac{\Delta t}{\tau_{\text{pre}}}}, & \text{if } \Delta t > 0, \\ -A_{\text{post}} e^{-\frac{\Delta t}{\tau_{\text{post}}}}, & \text{otherwise}, \end{cases} \tag{4}$$

where A_{pre} and A_{post} are positive constants to determine the limit of $F(\Delta t)$, τ_{pre} and τ_{post} are time constants to scale Δt, and $\Delta t = t_{s_{\text{post}}} - t_{s_{\text{pre}}}$ is positive when the pre-neuron fires a spike before the post-neuron.

D. POWER CONSUMPTION OF SNN

The SNN can be implemented on three kinds of materials: 1. classical digital circuits, such as Intel CPU or FPGA; 2. neuromorphic manycore processor, such as TrueNorth [15], Loihi [16] and Braindrop [17]; 3. low-power neuromorphic circuits with weak-inversion CMOS, such as in [14] and [18]. We analyze their corresponding power consumption separately.

1) CLASSICAL DIGITAL CIRCUITS

With classical digital circuits, the spike is represented by bit 1 while bit 0 represents non-spiking state. The power consumption of an FPGA is proportional to its sampling frequency according to [39]:

$$P = V_{\text{dc}}^2 \cdot f \cdot C_l \cdot \alpha \tag{5}$$

where α is the switching activity of bits which corresponds to number of flops (floating-point operations per second), C_l the load capacitance, V_{dc} the supply voltage and f is the system frequency directly related to the sampling frequency f_s. The power consumed by a classic DPD is usually in order of several hundreds of mW and up to 2 W according to [40].

2) NEUROMORPHIC MANYCORE PROCESSOR

The neuromorphic processors are specifically designed for synaptic operations. Its power consumption is counted per synaptic operation. According to [15], TrueNorth consumes 26 pJ per synaptic event which renders around 400 billion synaptic operations per second (SOPS) per watt. Loihi in [16] consumes 23.6 pJ per synaptic event and Braindrop needs only 381 fJ per synaptic event according to [17].

3) LOW-POWER NEUROMORPHIC CIRCUITS

Recently some electrical neurons (e-neuron) have been proposed with CMOS transistors working in weak-inversion mode which permits the circuit to work with very low current. The e-neuron in [18] needs 3.6 fJ for each spike. Moreover, the circuit is only activated in very short time when there is a spike. At other time, the circuit is equivalent to being deactivated. The power consumption of an SNN on different material are listed in Table 1. In [37], authors have measured the power of e-Neuron based on ML model designed in [14] and obtain 5 nW for the silicon core with an area of 0.025 μm^2, which is negligible in front of classical digital circuits.

III. PROPOSED SNN-DPD

A. CHALLENGES IN APPLICATION OF SNN

As the SNN technique has a great advantage on energy efficiency and is promising for AI development thanks to its biomimicry, we propose to apply it as a DPD for PA linearization. As described in the previous section, if the input of a spiking neuron is a constant, its output is a spike train with a constant frequency. Otherwise, the frequency of the spike train at the neuron output is also varying as shown in Fig. 2.

TABLE 1. Comparison of hardware consumption per spike for SNN.

	FPGA [40]	TrueNorth [15]	Loihi [16]	Braindrop [17]	e-neuron [18]
Year	2014	2014	2018	2019	2021
E(pJ)	>2000	26	23.6	0.38	0.0036

We need to wait for an observation window to read out the information of f_{spike}. The length of this observation window determines the bit rate of each neuron. Since the neuron spike rate is stable when excited by a constant current, we can count the time delay between two neighbored spikes and take its reciprocal as f_{spike}. With the e-Neuron in Braindrop in [17] and circuits designed in [18], the length of the observation window needs several microseconds to cover this time delay.

Since the DPD working in frequency domain has lower complexity and submits to less constraints on the sampling rate of the input signal according to [29], in this work, we use the SNN to process the subcarriers of the orthogonal frequency division multiplexing (OFDM) signal before it goes to time domain. The system of an orthogonal frequency division multiplexing (OFDM) transmitters (TX) with TD DPD and FD DPD is illustrated in Fig 5(a) and 5(b), respectively. A frame of OFDM data is first created in frequency domain and is split into parallel subcarriers through a serial to parallel (S/P) converter. The time-domain signal $x(n)$ for the PA is then prepared by inverse fast Fourier transform (IFFT). The TD DPD processes the signal after the IFFT and FD DPD processes the signal before the IFFT. There is no additional complexity brought by the FD DPD. In this figure, we omit the digital-to-analog converter (DAC) and the modulator which upconvert the baseband signal $x(n)$ to radio-frequency (RF) before feeding the signal to the PA.

B. PHYSICS INFORMATION OF FD SIGNAL PROCESSING

A conventional TD DPD is modeled by the GMP as expressed in [23]:

$$\begin{aligned} x(n) = &\sum_{k=0}^{\mathcal{K}_a-1}\sum_{l=0}^{\mathcal{L}_a-1} a_{kl}u(n-l)|u(n-l)|^k \\ &+ \sum_{k=1}^{\mathcal{K}_b}\sum_{m=1}^{\mathcal{M}_b}\sum_{l=0}^{\mathcal{L}_b-1} b_{kml}u(n-l)|u(n-l-m)|^k \\ &+ \sum_{k=1}^{\mathcal{K}_c}\sum_{m=1}^{\mathcal{M}_c}\sum_{l=0}^{\mathcal{L}_c-1} c_{kml}u(n-l)|u(n-l+m)|^k, \end{aligned} \tag{6}$$

where the input signal is $u(n)$, k is the index for nonlinearity, m and l are the indices for memory, and a_{kl}, b_{kml} and c_{kml} are the complex coefficients. The extraction of DPD model coefficients using direct or indirect learning architecture (DLA/ILA) has been well described in [41] and [42]. If we denote $U(\omega)$ and $X(\omega)$ as the Fourier transform of $u(n)$ and $x(n)$ with sampling rate f_s in spectrum $[-\frac{f_s}{2},\frac{f_s}{2}]$ respectively,

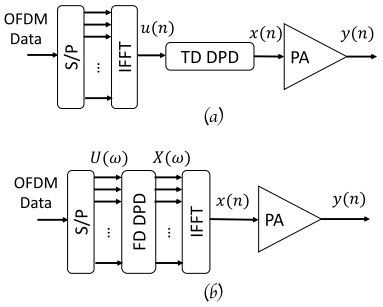

FIGURE 5. Diagram of system with (a) TD DPD and (b) FD DPD.

an FD-GMP model can be deduced from (6) for the FD-DPD:

$$\begin{aligned} &X(\omega) \\ &= \sum_{k=0}^{\mathcal{K}_a-1}\sum_{l=0}^{\mathcal{L}_a-1} a_{kl}e^{-j2\pi\frac{\omega}{f_s}l}U(\omega) * \Phi_k\big(U(\omega)\big) \\ &\quad + \sum_{k=1}^{\mathcal{K}_b}\sum_{m=1}^{\mathcal{M}_b}\sum_{l=0}^{\mathcal{L}_b-1} b_{kml}e^{-j2\pi\frac{\omega}{f_s}l}U(\omega) * \Phi_k\big(U(\omega)e^{-j2\pi\frac{\omega}{f_s}(l+m)}\big) \\ &\quad + \sum_{k=1}^{\mathcal{K}_c}\sum_{m=1}^{\mathcal{M}_c}\sum_{l=0}^{\mathcal{L}_c-1} c_{kml}e^{-j2\pi\frac{\omega}{f_s}l}U(\omega) * \Phi_k\big(U(\omega)e^{-j2\pi\frac{\omega}{f_s}(l-m)}\big), \end{aligned} \tag{7}$$

where $*$ is the convolution function, and $\Phi_k\big(U(\omega)\big)$ can be defined as in [29]:

$$\Phi_k\big(U(\omega)\big) = \begin{cases} U(\omega)^{*\frac{k}{2}}\bar{U}(-\omega)^{*\frac{k}{2}}, & \text{if } k \text{ is even,} \\ U(\omega)^{*\frac{k+1}{2}}\bar{U}(-\omega)^{*\frac{k-1}{2}}, & \text{otherwise,} \end{cases} \tag{8}$$

where $\bar{U}$ represents the conjugate of U, and $(\cdot)^{*k}$ represents the convolution power.

The FD-DPD can also be extended for linearization of multi-band case. According to the 2D-MP model in [43]:

$$\begin{aligned} x_s(n) = &\sum_{k=0}^{\mathcal{K}_s-1}\sum_{r=0}^{k}\sum_{l=0}^{\mathcal{L}_s-1} a_{krl}^{(1)}u_s(n-l) \\ &\times |u_s(n-l)|^{k-r}|u_{\bar{s}}(n-l)|^r, \end{aligned} \tag{9}$$

where u_s and x_s (s=1,2) denotes input and the predistorted signal of the s-th band respectively, $x_{\bar{s}}$ represents the input signal other than the s-th band, n is the index of the signal sample, $\mathcal{K}_s$ is the nonlinearity order for the s-th band, and $\mathcal{L}_s$ is the memory depth for the s-th band.

The FD-2D DPD model can then be deduced as

$$\begin{aligned} X_s(\omega) = &\sum_{k=0}^{\mathcal{K}_s-1}\sum_{r=0}^{k}\sum_{l=0}^{\mathcal{L}_s-1} a_{krl}e^{-j2\pi\frac{\omega}{f_s}l}U_s(\omega) \\ &* \Phi_{k-r}\big(U_s(\omega)\big) * \Phi_r\big(U_{\bar{s}}(\omega)\big). \end{aligned} \tag{10}$$

The subcarriers of the predistorted signal depends thus on the subcarriers of signals in both two bands.

The reasoning above gives a physics information for the SNN-DPD that, if we take the subcarriers of signal

spectrum $U(\omega)/X(\omega)$ as the SNN's input/output data, each output neuron should be connected to all input neurons. In the proposed SNN-DPD, an SNN is expected to replace the highly power-consuming convolutions with low-power spike processing.

C. THE STRUCTURE OF SNN-DPD

According to the physics information of given in the previous section and the study in [37], we propose an SNN-DPD which realizes the functionality of the FD-DPD with a full connected two-layer SNN. The system structure is depicted in Fig. 6. Since the SNN processes only positive real data, the complex subcarriers values of the OFDM data are split into real-part group and imaginary-part group by functions real$(\cdot)$ and imag$(\cdot)$ respectively. These values of real-part and imaginary-part are then offset to positive and are rescaled according to the maximal value of neuron's I_{ex}. For a stimulus of bandwidth B, if the subcarriers of the stimulus spectrum can be expressed with discrete $\omega \in [-\frac{B}{2}, \frac{B}{2}]$ with a step of $\Delta\omega$ as:

$$U(\omega) = U_r(\omega) + iU_i(\omega), \tag{11}$$

the input of the i-th neuron in the input layer is

$$I_{\text{ex}}(i) = \begin{cases} (\dfrac{U_r(i)}{\max|U(\omega)|} + 0.5)I_{\text{ex},max}, & \text{if } i < \dfrac{B}{\Delta\omega}, \\ (\dfrac{U_i(i-N)}{\max|U(\omega)|} + 0.5)I_{\text{ex},max}, & \text{otherwise}, \end{cases} \tag{12}$$

where N is the number of subcarriers $U(\omega)$, $\Delta\omega{=}\frac{1}{N}f_s$, and $i{=}[1,\ldots,2N]$, $I_{\text{ex},max}$ is the maximum of excitation current for the spiking neuron. We connect the neurons of output layer directly with the input layer without any hidden layer. The output of the neurons in the output layer are spike trains of different f_{spike} which corresponds to the real part and imaginary part of the frequency components of spectrum of the desired predistorted signal. The values of f_{spike} can be read out and translated to current information as shown in Fig. 1. To be noticed, this translation of output information from f_{spike} to the current format is just because our target data in network training is in form of current I_{ex}, which will be explained in the following subsection. They are regrouped according to the order of neuron index to restore the complex value for the IFFT. The predistorted signal $x(n)$ converted from $X(\omega)$ is then fed to the PA. As discussed in [29], the FD-DPD is implemented before the OFDM signals are converted into time domain signals by IFFT. Therefore, there is no additional complexity of IFFT to be considered compared with the time-domain DPD techniques.

D. TRAINING OF 2-LAYER SNN-DPD

The identification of the proposed SNN-DPD model can be in different ways. In this paper, we first extract an ideal predistorted signal $x_d(n)$ using the augmented iterative learning control (AILC) technique [3], [44].

The AILC is an iterative process by adjusting the PA input signal to make the PA output signal converge towards a desired signal $y_d(n)$, where $y_d(n){=}G \cdot u(n)$ with G denotes the linear gain of the PA [45]. The AILC at the k-th iteration is depicted in Fig. 7. A Gaussian white noise $d(n)$ is considered at the PA input. The initial input of PA is set as $x_0(n){=}u(n)$. At each iteration, $x(n)$ is updated according to the error between the measured PA output $y(n)$ and the desired signal $y_d(n)$:

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + L \cdot \mathcal{F}\{\boldsymbol{e}_k\}, \tag{13}$$

where $\boldsymbol{x}_k = [x_k(1), \ldots, x_k(N)]$, N is the number of signal samples, L is the learning matrix, $\mathcal{F}\{\cdot\}$ represent low pass filtering, $\boldsymbol{e}_k = [y_d(1) - y_k(1), \ldots, y_d(N) - y_k(N)]$. After the system converges within several iterations, we obtain the updated signal $x(n)$ as the desired predistorted signal $x_d(n)$.

The SNN is then trained with $U(\omega)$ and $X_d(\omega)$ which is the FFT of $x_d(n)$ as illustrated in Fig. 8. The target dataset in the SNN training is biased and rescale $X_d(\omega)$. We rescale the subcarriers of main band spectrum and adjacent band spectra differently since its power in adjacent bands is much lower than that in the main band. For the purpose of not making the figure too complicated, in Fig. 8 we simply demonstrate the training process with main band of $X_d(\omega)$ which has the same number of subcarriers as $U(\omega)$. The training for adjacent bands are similar with the same input data $U(\omega)$ while only the target data are of different bands.

According to (2), we can see that a neuron fires spikes when it is excited by a current I_{ex} or when it receives enough spikes transmitted by synapses. In the proposed SNN-DPD, the neurons in the input layer can be excited by certain values of currents or by certain spike trains. Either of them should correspond to the input signal $U(\omega)$. In contrast, the neurons in the output layer only receive information transmitted from the input layer through synapses. Thus, in Fig. 8, we create a supervisor layer (in red) to enable the supervised training.

We first encode the real part and imaginary part of subcarriers $U(\omega)$ to excitation currents I_{ex} of the neurons in the input layer. The spike trains fired by these input layer then represent the information of $U(\omega)$. In the same way, we create spike trains from neurons in the supervisor layer with excitation currents which represent the desired predistorted signal $X_d(\omega)$.

The remote supervised method (ReSuMe) in [46] is adopted for the 2-layer SNN training. The neurons in the supervisor layer are connected to the neurons in the input layer in the same way as red synapses in Fig. 6.

The training is an iterative procedure as describe in Algorithm 1 where at each iteration we separately compute the bias of two groups of weight Δw_i^o and Δw_i^d according to (3) and (4). The weight of the i-th synapses at the k-th iteration is then updated by

$$w_{i,k} = w_{i,k-1} + \Delta w_i^o - \Delta w_i^d. \tag{14}$$

Then, by running the training system in Fig. 8 for a necessary period, the spiking time differences $\Delta t_{s_{\text{input}},s_{\text{output}}}$ between the input and output layers will converge to the spiking time

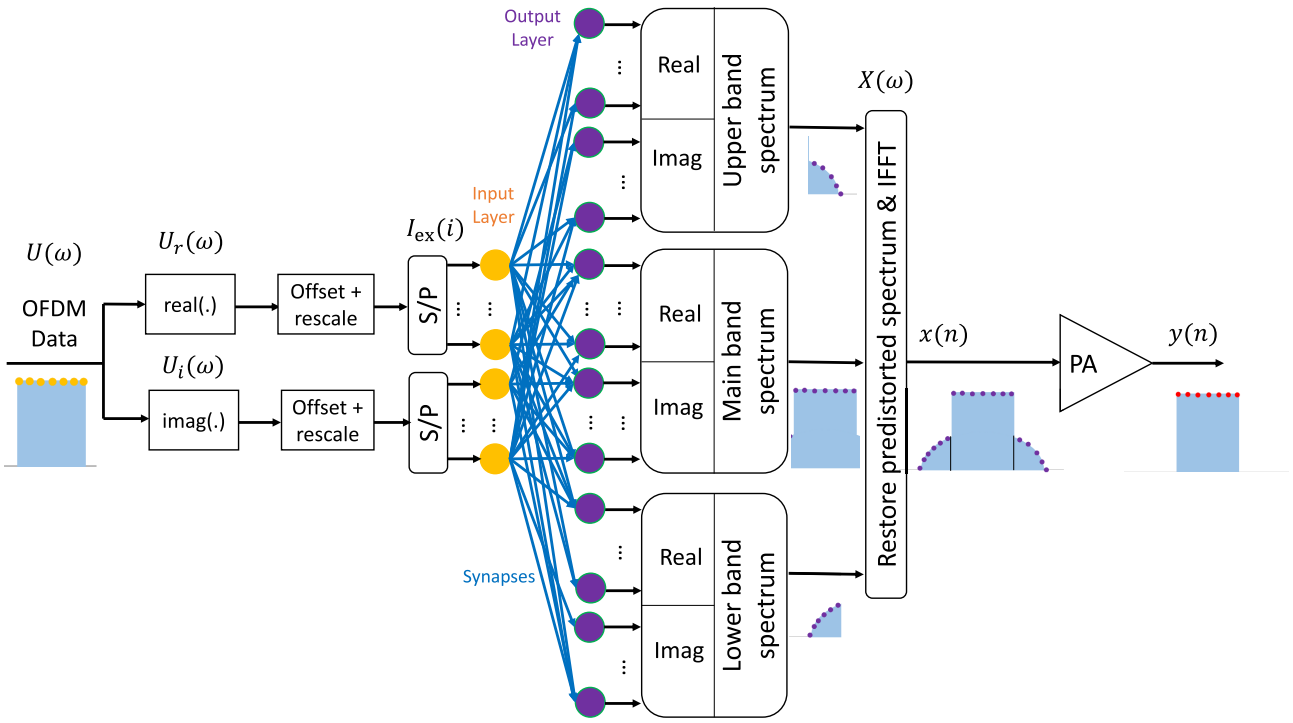

FIGURE 6. PA linearization with SNN-DPD.

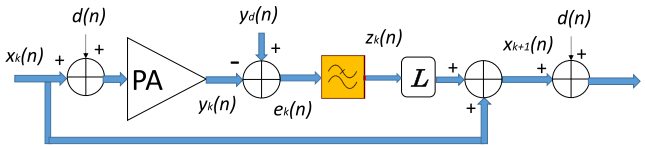

FIGURE 7. AILC scheme at the k-th iteration.

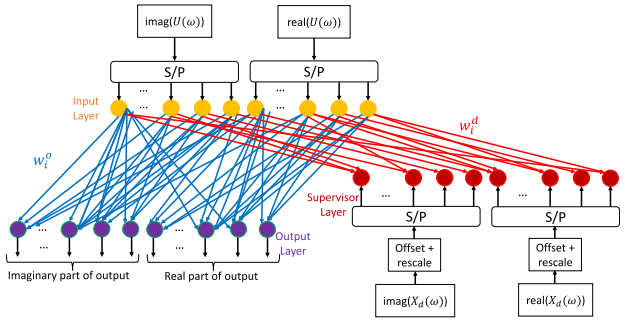

FIGURE 8. Training of SNN-DPD.

Algorithm 1 Training of the i-Th Synapse

Set the maximum loop number $K_{\max}$;
Initialize $w_{i,0}$=0 ;
for k=1:K_{max} **do**
  Feed I_{ex}=U to the pre-neuron in input layer;
  Obtain spiking times $t_{s_{input}}$;
  Run the SNN with $w^o_{i,k-1}$;
  Obtain spiking times of the post-neuron in output layer $t_{s_{output}}$;
  Compute Δw_i^o with $t_{s_{input}}$ and $t_{s_{output}}$ by (3);
  Feed I_{ex}=X_d to the post-neuron in supervisor layer;
  Obtain spiking times $t_{s_{supervisor}}$;
  Compute Δw_i^d with $t_{s_{input}}$ and $t_{s_{supervisor}}$ by (3);
  $w^o_{i,k} = w^o_{i,k-1} + \Delta w_i^o - \Delta w_i^d$;
end

differences $\Delta t_{s_{input},s_{supervisor}}$ between the input and supervisor layers, so that $\Delta w_i^o - \Delta w_i^d \approx 0$ and w_i^o is stable.

E. SNN-DPD FOR BAND-LIMITED AND MULTI-BAND LINEARIZATION

With a modulated OFDM stimulus as the DPD input, the predistorted signal has spectral regrowth which occupies wider bandwidth due to the nonlinearity. From the structure of the proposed SNN in Fig. 6, each subcarrier of predistorted signal can be computed with the entire ensemble of subcarriers of the input signal independently. In other words, the bandwidth or even the frequency bands of the predistorted signal can be customized. This allows band-limited linearization and multi-band linearization with limited number of neurons in the output layer.

We depict the spectra of the DPD input signal $U(\omega)$ and the desired predistorted signal $X_d(\omega)$ acquired by AILC in Fig. 9, where the dots represent the subcarriers. We can see that the SNN-DPD can have a band-limited predistorted signal if we

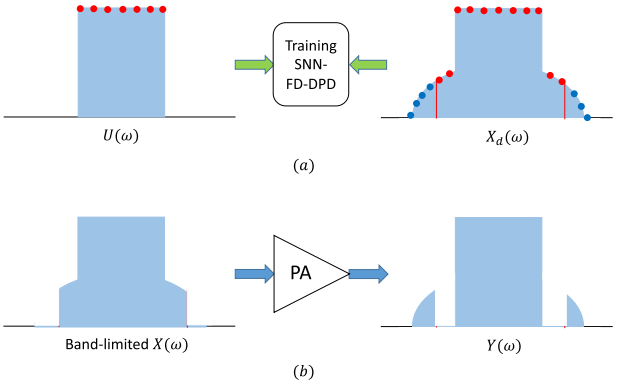

FIGURE 9. Subcarriers of DPD input and output for band-limited PA linearization: (a) selection of subcarriers in the band of desired linearization; (b) the linearization effect within the selected band of predistorted signal.

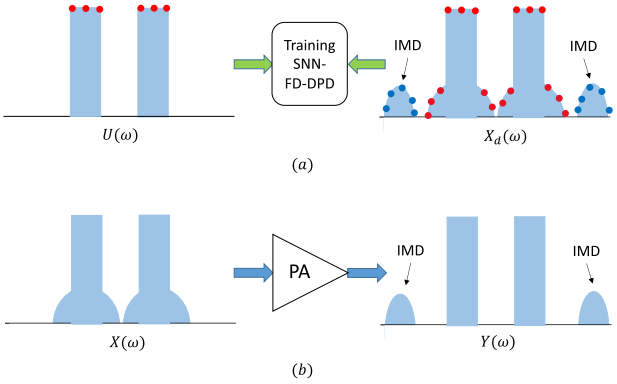

FIGURE 10. Subcarriers of DPD input and output for multi-band linearization: (a) selection of subcarriers in the band of desired linearization; (b) the linearization of a dual-band signal.

train the SNN with limitied number of subcarriers for the supervisor layer as the red dots shown in Fig. 9(a) while the blue dots represented subcarriers excluded from the target training data. With all subcarriers of $U(\omega)$ fed to the input layer of the SNN, we can generate subcarrier of $X(\omega)$ of any wanted frequency at the output layer. If we generate only the predistorted subcarriers in the limited band, we can achieve a linearization of the PA in any specific bands as shown in Fig. 9(b), which results in an equivalent performance as the band-limited DPD in [47]. With lower number of subcarriers for $X(\omega)$, we can reduce the number of neurons in the output layer of the SNN, which reduces the complexity of the network and its power consumption.

With the same principle, the SNN-DPD can be applied for multi-band linearization as illustrated in Fig. 10. The predistorted signal depends only on the subcarriers of the signals in the two bands, and is independent of the frequency separation of the two bands. We train the SNN with the subcarriers of transmitted signals and the subcarriers of desired predistorted signals in the interesting bands, as depicted in Fig. 10(a). The intermodulation (IMD) products of signals in the two bands are not considered in the linearization since they can be easily filtered out as depicted in Fig. 10(b).

TABLE 2. Number of flops for operations.

Operation	flops	Operation	flops
Real Addition	1	Real Multiplication	1
Real Division	4	Complex Addition	2
Complex Multiplication	6	Complex-Real Multiplication	2
Square-root	7		

IV. COMPLEXITY ANALYSIS

Though the advantage of SNN on energy saving is mainly due to the neuromorphic circuits, we still make a comparison against some classical DPD methods implemented on classical digital hardware in this paper. The polynomial based model and classical ANN are taken as comparison references. For a fair comparison, we compute the number of flops according to Table 2 in [48]. Both running complexity and training complexity are discussed in this section.

A. RUNNING COMPLEXITY

According to [48], the complexity of a GMP model is estimated as:

$$F_{\text{gmp}} = 3 + 7 + F_{\text{crm}}(\mathcal{K}_a + \mathcal{K}_b\mathcal{M}_b + \mathcal{K}_c\mathcal{M}_c) + 8N_{\text{coeff}} - 2, \tag{15}$$

where F_{crm} is 2 flops for complex-real multiplication (see Table. 2), and N_{coeff} is the number of model coefficients.

The complexity of a classical ANN with h layers is

$$F_{\text{ann}} = 4N_1 + \sum_{i=1}^{h-1} 2N_iN_{i+1} + 4N_h. \tag{16}$$

where N_i is the number of neurons in the i-th layer.

The complexity of the SNN when implemented on classical digital circuits can be expressed as

$$F_{\text{snn}} = 2F_{\text{rescale}} + N_{\text{in}}F_{\text{synaptic}}. \tag{17}$$

where F_{rescale} is the complexity for bias (real addition) and rescaling (real multiplication) equal to $2\frac{N_{\text{in}}}{N_{\text{out}}}$, F_{synaptic} is the complexity of a synaptic computing which is a real multiplication, N_{in} and N_{out} are number of neurons in the input and output layer respectively.

B. TRAINING COMPLEXITY

The training of polynomial-based model is the linear extraction of the model coefficients through matrix inversion. According to [26], the complexity of N_{coeff} coefficients extraction using QR-decomposition with a dataset of N samples is

$$F_{\text{iden}} = 2NN_{\text{coeff}}^2 + \frac{1}{3}N_{\text{coeff}}^3 + 2NN_{\text{coeff}} - N_{\text{coeff}}. \tag{18}$$

The complexity of training a classical ANN is mainly from the back-propagation computation. For simple estimation, we consider the derivation as a real division of 4 flops. Thus

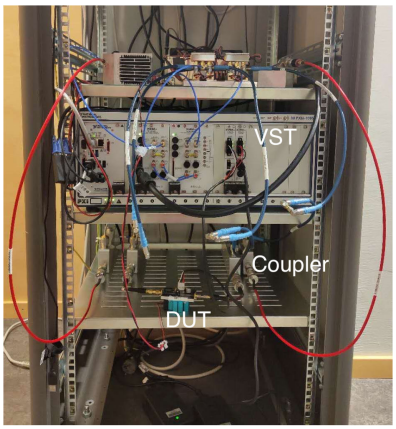

FIGURE 11. Test bench of WebLab [31] for Experimental Implementation.

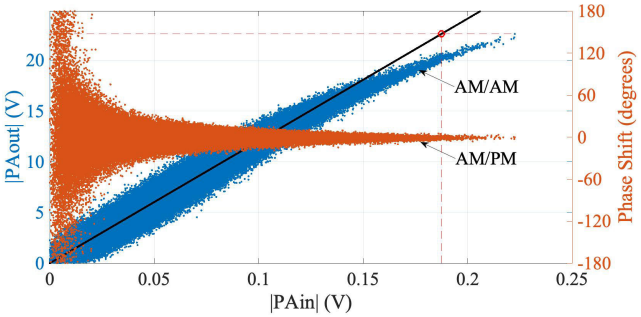

FIGURE 12. PA AM/AM & AM/PM curves from testbench of WebLab.

complexity of training an ANN of h layers in E_{ann} epochs using the dataset of N sample is

$$F_{\text{ann,tr}} = E_{\text{ann}} \sum_{i=1}^{h-1} 4N_i N_{i+1} N. \tag{19}$$

The complexity of training the proposed SNN is mainly from (14) with 2 real additions. Thus complexity of training the proposed SNN in E_{snn} epochs is

$$F_{\text{snn,tr}} = 2E_{\text{snn}} N_{\text{in}} N_{\text{out}}. \tag{20}$$

V. EXPERIMENTAL RESULTS

A. MEASUREMENT SETUP

We use test bench of WebLab [31] for measurements as depicted in Fig. 11. The proposed DPD is tested with different dual-band 5 MHz LTE signal and single-band LTE signals with 20 MHz and 40 MHz bandwidth as the stimulus. The sampling rate of these signals is all 200 MHz with length of 20000 samples. The baseband IQ signal is fed from the PC Workstation to the driver through a Vector Signal Transceiver (PXIe-5646R VST) using a 200 MHz sampling frequency. The VST up-converts the baseband signal to the carrier frequency 2.14 GHz. The signal at the output of the PA is then down-converted to baseband by the VST which provides to the PC workstation the baseband signal digitized with a sampling frequency of 200 MHz. The input and output baseband signals are then synchronized in time to be used by the identification algorithm.

A GaN PA CGH40006P transistor mounted in the manufacturer demo-board fabricated by CREE has been used to validate the proposed low rate DPD. Its nominal gain is 13 dB at 2 GHz and the output power at 1dB gain compression is 40.2 dBm. The nonlinearities and the memory effect of this PA can be seen from the AM/AM & AM/PM (Amplitude Modulation/Amplitude Modulation & Amplitude Modulation/Phase Modulation) curves in Fig. 12.

The SNN is implemented with Python, which is a validation of the functioning and can be further implemented on neuromorphic circuits in the following studies. In this section, we make the estimation of the SNN energy consumption by taking Braindrop [17] as a reference.

B. CHARACTERISTICS OF HH MODEL SNN

The HH model is the most general model of a spiking neuron. One realization of the electronic HH neuron has been proposed in [49]. The Izhikevich model and LIF model can also be used for spiking neurons. In this paper, we use an HH-based SNN simulator thanks to Brian 2 library [50]. The I_{ex}-f_{spike} curve of an HH-model SNN is depicted in Fig. 13 with a certain configuration of electronic parameters (C_m, G_{Na}, G_{K}, G_L, E_{Na}, E_{K}, E_L). In the simulator, we set the area of neuron at 20000 μm^2, so that C_m=200 pF, G_{Na}=20 nS, G_{K}=6 nS, G_L=10 nS. And we set E_{Na}=50 mV, E_{K}=−90 mV, E_L=−65 mV. The variables (α_m, α_n, α_h, β_m, β_n, β_h) in (1) are defined as

$$\begin{aligned}
\alpha_n &= \frac{0.032(15 - V_m + V_T)}{e^{\frac{15-V_m+V_T}{5}} - 1} \\
\beta_n &= 0.5e^{\frac{10-V_m+V_T}{40}} \\
\alpha_m &= \frac{0.32(13 - V_m + V_T)}{e^{\frac{13-V_m+V_T}{4}} - 1} \\
\beta_m &= \frac{0.28(V_m - V_T - 40)}{e^{\frac{V_m-V_T-40}{5}} - 1} \\
\alpha_h &= 0.128e^{\frac{17-V_m+V_T}{18}} \\
\beta_h &= \frac{4}{1 + e^{\frac{40-V_m+V_T}{5}}},
\end{aligned} \tag{21}$$

and the threshold voltage is set at V_T=−63 mV. We rescale the signals sent to neurons as I_{ex} data between 0 and 40 nA to avoid the divergence.

In Fig. 14, from the top to bottom, we show the spike trains of the 4-th neuron in the input layer, the output layer after training, and the supervised layer, respectively. The time scale is dependent on the configuration of parameters, such as area of chip, capacitance C_m, etc. Even though the neuron of the output layer does not have the same waveform of V_m as the neuron of the supervisor layer, their spiking rates are very close. The waveform of V_m of the neuron in the output layer is dominated by the spikes of the neurons in the input layer, which determines it spiking or not by comparison with the threshold potential V_T. The information of spiking rate f_{spike}

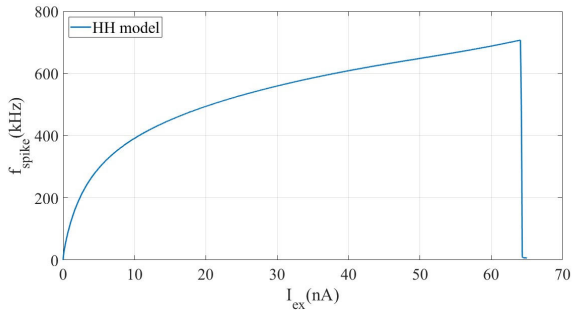

FIGURE 13. Characteristics of the HH model neuron.

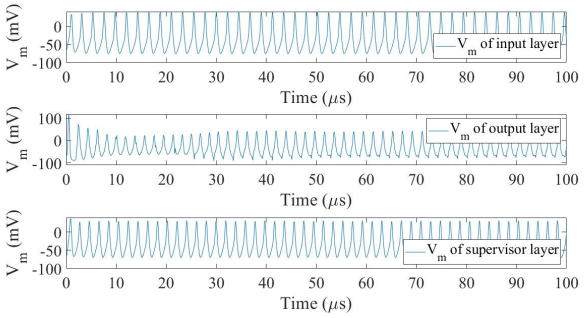

FIGURE 14. Membrane potential V_m of the input layer, output layer, and supervisor layer of the SNN.

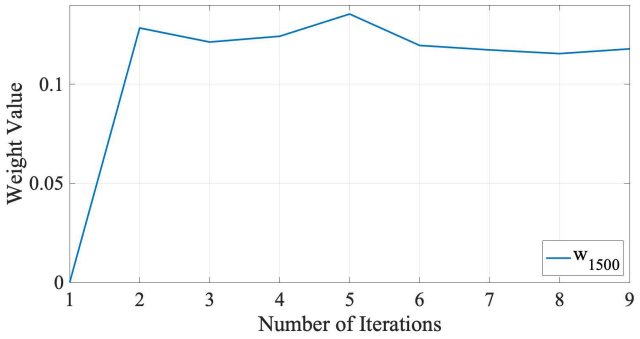

FIGURE 15. Evolution of the 1500-th weight during 9 epochs (iterations) of training.

is thus transmitted from the input layer to the output layer in a nonlinear way. This biological mimicry of the human brain, which processes information in form of spikes, brings a high degrees of freedom in AI system.

In Fig. 15, the evolution of a synapses weight (the 1500-th synapse) is given for the first 9 epochs. The value of w_{1500} becomes stable after the 6-th epoch.

C. LINEARIZATION PERFORMANCE OF SNN-DPD

We evaluate the linearization performance of the proposed SNN-DPD with the testbench of WebLab. The peak-to-average power ratio (PAPR) of the tested OFDM stimulus is about 7.6 dB. The average power of the signal at the input of the driver is around −19.8 dBm. The measured average output power of the PA is 29.7 dBm.

The GMP-DPD is tested under the same circumstance as a reference for the comparison. Thanks to the algorithm in [51], we are allowed to determine a group of optimal parameters for the model structure of the GMP-DPD:

$$\begin{aligned} &\mathcal{K}_a = 6,\ \mathcal{L}_a = 5 \\ &\mathcal{K}_b = 3,\ \mathcal{L}_b = 1,\ \mathcal{M}_b = 2 \\ &\mathcal{K}_c = 1,\ \mathcal{L}_c = 6,\ \mathcal{M}_c = 3. \end{aligned} \tag{22}$$

There are 54 coefficients. The tested ANN has 2 hidden layers with 40 neurons per layer.The number of epochs in the training of ANN is 100.

The PA output spectra with and without linearization are given in Fig. 16. The 1st adjacent channel power ratio (ACPR) and error vector magnitude (EVM) values are listed in Table 3. The yellow green curve with circles presents the theoretical best linearization performance given by the AILC. Since the AILC is only an off-line method to acquire the most appropriate predistorted signal rather than a forward DPD method, it is provided as a reference of the lowest bound. The result of the GMP DPD is shown with green dashed curve. With the strong nonlinearity and memory depth of the PA biased with V_{gs}=−2.8V in the WebLab testbench, the conventional GMP has limited performance especially at near-band region (inside [−20 MHz, 20 MHz]). The blue curve with squares shows the performance of a classical ANN DPD. The result of the proposed SNN-DPD is examined as the red curve shows. We can see that the linearization performance of the proposed SNN-DPD can reach a better level than the classical GMP DPD and a similar level as the classical ANN DPD. The power consumption of the SNN estimated with Braindrop is 4.6 mW which is extremely low compared with classical DPD according to [40].

The generated OFDM signal has 20000 subcarriers spanned in 200 MHz. Since the bandwidth of stimulus is 20 MHz, we only need to process 2000 subcarriers for the input of the SNN. Each subcarrier corresponds to a neuron in the input layer. For the output of the SNN, we consider 6 times of the stimulus bandwidth which demands 12000 subcarriers to process. We make symmetrical zero padding so that we can obtain the 20000-sample time domain signal. In Table 3, we list also the estimation of computational complexity and power consumption of the SNN when it is implemented on classical digital circuits and neuromorphic circuits respectively. We denote the running complexity by F_{run} and the training complexity by F_{train}. The total complexity is computed by multiplying F_{run} with the number of processed samples. According to Table 1, the power consumption of different circuits has a huge variation. We count the energy of synaptic event in Braindrop as a reference and in average 300 events/millisecond for each synapse. The running complexity of the GMP is lower than neural networks since we applied an optimization technique in [51] on pruning its structure. For the ANN and SNN, we use full connection networks without any pruning technique.

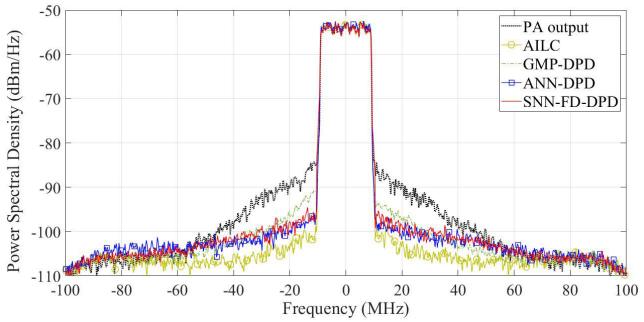

FIGURE 16. PA output spectra tested with 20 MHz LTE.

TABLE 3. Performance comparison of DPD models with 20 MHz LTE.

	No DPD	GMP-DPD	ANN DPD	SNN-DPD	AILC
ACPR.L (dBc)	-34.2	-40.9	-44.8	-44.6	-48.1
ACPR.U (dBc)	-35.2	-42.3	-45.7	-45.2	-48.1
EVM (%)	7.3	2.2	1.6	1.6	1.3
No. of samples	-	20000	20000	12000	-
F_{run} (flops)	-	1297	2208	2001	-
Total complexity	-	$1.5e^7$	$4.4e^7$	$2.4e^7$	-
F_{train} (flops)	-	$11.8e^7$	$1.7e^{10}$	$4.3e^8$	-

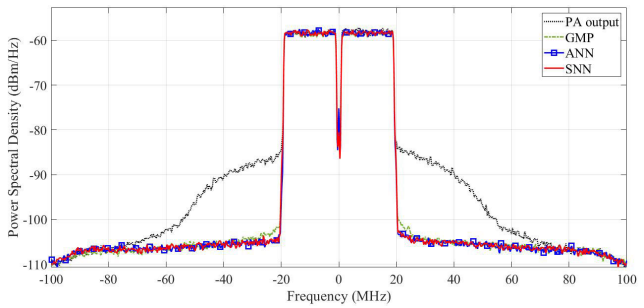

FIGURE 17. PA output spectra tested using 40 MHz LTE.

The PA in WebLab can be biased with a gate-to-source voltage V_{gs} for different behaviors. To better validate the effectiveness of the proposed SNN-DPD, we bias the PA with V_{gs}=−2.1V. We also change the stimulus to 40 MHz LTE. The linearization performance of different DPD techniques is depicted in Fig. 17. The ACPR and EVM values are listed in Table 4. The biased PA in WebLab testbench is easier to linearize than unbiased PA which has been testified in [52]. The nonlinearity orders $\mathcal{K}$ for three branches (a, b and c) are set to 10, and memory depths $\mathcal{L}$ for three branches (a, b and c) and $\mathcal{M}$ for two branches (b and c) are set to 6 and 2 respectively. The GMP DPD has almost the same linearization performance as the ANN and the SNN. However, the proposed SNN-based DPD has always an obvious advantage on energy consumption. Considering the implementation on classical digital circuits, the proposed SNN-DPD needs less flops than GMP-DPD and ANN DPD on both running and training. With a neuromorphic circuit such as Braindrop,

TABLE 4. Performance comparison of DPD models with 40 MHz LTE.

	No DPD	GMP-DPD	ANN DPD	SNN-DPD
ACPR.L (dBc)	-27.2	-45.1	-46.4	-46.6
ACPR.U (dBc)	-26.7	-45.3	-46.8	-46.2
EVM (%)	9.5	1.2	1.0	1.0
No. of samples	-	20000	20000	20000
F_{run} (flops)	-	2986	2208	2001
Total complexity	-	$6.1e^7$	$4.4e^7$	$4.0e^7$
F_{train} (flops)	-	$3.7e^9$	$1.7e^{10}$	$7.2e^8$

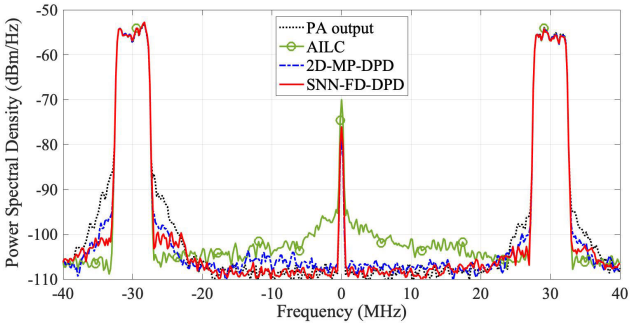

FIGURE 18. Dual-band linearization.

TABLE 5. Performance of dual-band linearization.

		No DPD	2D-DPD	SNN-DPD	AILC
Band 1	ACPR.L (dBc)	-37.9	-45.0	-47.1	-50.9
	ACPR.U (dBc)	-38.6	-45.4	-46.3	-49.7
	EVM (%)	4.6	1.6	1.1	0.6
Band 2	ACPR.L (dBc)	-44.4	-47.1	-48.5	-49.3
	ACPR.U (dBc)	-43.6	-47.0	-48.0	-49.5
	EVM (%)	3.3	1.3	0.7	0.6
No. of samples		-	6000	6000	-
F_{run} (flops)		-	1750	1002	-
F_{train} (flops)		-	$1.2e^8$	$1.1e^8$	-

the SNN consumes a negligible power in front of classical methods.

D. DUAL-BAND LINEARIZATION

We also test the proposed SNN-DPD for multi-band linearization with a dual-band input stimulus. The linearization performances of the proposed technique and traditional 2D-MP model are given in Fig. 18. The ACPR in the 1st adjacent channels and EVM values for each transmission band are listed in Table 5.

The PAPR of the dual-band stimulus is about 9.3 dB. The average power of the signal at the input of the driver is around -23.5 dBm. The measured average output power of the PA is 27.2 dBm. Compared with the single-band case in the previous section, the dual-band signal suffers less nonlinearity distortion since the operating point of the PA is lower.

The proposed SNN-DPD depicted by the red curve outperforms the traditional 2D-MP-DPD which is the blue dashed curve when looking at the spectral regrowth on the adjacent channels. The SNN-DPD has shown its effectiveness and robustness on linearization for different types of signals no matter how its frequency carriers are located. The estimated

consumption is 0.38 mW. The SNN-based technique may not have as obvious advantage on training complexity as on running complexity, especially since some techniques such as in [53] can reduce the training complexity of time domain DPD with a factor of 5. However, since the network does not need to be trained all the time, the running complexity is a more critical factor to be considered for power consumption. Besides, the SNN-based DPD enables its implementation on neuromorphic circuits, which has far more energy-saving than classical DPD techniques.

VI. CONCLUSION

In this paper, we propose an SNN-DPD for PA linearization. To the best of authors' knowledge, this is the first time that an SNN is applied on DPD techniques. The proposed SNN-DPD processes an OFDM signal in frequency domain and generates the predistorted subcarriers at desired frequencies. According to the experimental validation on WebLab testbench for single-band and dual-band cases, the proposed technique gives better trade-off of linearization performance and energy consumption than traditional DPD methods. Besides, it allows implementations on neuromorphic hardware with low-power circuits. Moreover, since the SNN is more brain-like than other conventional neural networks, its application on RF PA linearization will provide more promising development of AI techniques on wireless communication systems.

ACKNOWLEDGMENT

The authors would like to thank Pr. C. Fager, and K. Buisman, Chalmers University of Technology, for their permission to use the RF WebLab measurement setup.

REFERENCES

[1] F. Mkadem and S. Boumaiza, "Physically inspired neural network model for RF power amplifier behavioral modeling and digital predistortion," *IEEE Trans. Microw. Theory Techn.*, vol. 59, no. 4, pp. 913–923, Apr. 2011.

[2] 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 Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 1, pp. 242–254, Jan. 2019.

[3] S. Wang, M. Roger, J. Sarrazin, and C. Lelandais-Perrault, "Augmented iterative learning control for neural-network-based joint crest factor reduction and digital predistortion of power amplifiers," *IEEE Trans. Microw. Theory Techn.*, vol. 68, no. 11, pp. 4835–4845, Nov. 2020.

[4] P. Jaraut, A. Abdelhafiz, H. Chenini, X. Hu, M. Helaoui, M. Rawat, W. Chen, N. Boulejfen, and F. M. Ghannouchi, "Augmented convolutional neural network for behavioral modeling and digital predistortion of concurrent multiband power amplifiers," *IEEE Trans. Microw. Theory Techn.*, vol. 69, no. 9, pp. 4142–4156, Sep. 2021.

[5] J. Cai, C. Yu, L. Sun, S. Chen, and J. B. King, "Dynamic behavioral modeling of RF power amplifier based on time-delay support vector regression," *IEEE Trans. Microw. Theory Techn.*, vol. 67, no. 2, pp. 533–543, Feb. 2019.

[6] C. Oszkinat, S. E. Luczak, and I. G. Rosen, "Uncertainty quantification in estimating blood alcohol concentration from transdermal alcohol level with physics-informed neural networks," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, Jan. 17, 2022, doi: 10.1109/TNNLS.2022.3140726.

[7] Z. Fang, "A high-efficient hybrid physics-informed neural networks based on convolutional neural network," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, no. 10, pp. 5514–5526, Oct. 2022.

[8] W. Maass, "Networks of spiking neurons: The third generation of neural network models," *Neural Netw.*, vol. 10, no. 9, pp. 1659–1671, Dec. 1997.

[9] S. Davidson and S. B. Furber, "Comparison of artificial and spiking neural networks on digital hardware," *Frontiers Neurosci.*, vol. 15, Apr. 2021, Art. no. 651141.

[10] Y. Cao, Y. Chen, and D. Khosla, "Spiking deep convolutional neural networks for energy-efficient object recognition," *Int. J. Comput. Vis.*, vol. 113, no. 1, pp. 54–66, May 2015.

[11] T. Matsubara and H. Torikai, "An asynchronous recurrent network of cellular automaton-based neurons and its reproduction of spiking neural network activities," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 4, pp. 836–852, Apr. 2016.

[12] R. Borwankar, A. Desai, M. R. Haider, R. Ludwig, and Y. Massoud, "An analog implementation of FitzHugh–Nagumo neuron model for spiking neural networks," in *Proc. 16th IEEE Int. New Circuits Syst. Conf. (NEWCAS)*, Jun. 2018, pp. 134–138.

[13] P. M. Ferreira, N. De Carvalho, G. Klisnick, and A. Benlarbi-Delai, "Energy efficient fJ/spike LTS e-Neuron using 55-nm node," in *Proc. 32nd Symp. Integr. Circuits Syst. Design (SBCCI)*, Aug. 2019, pp. 1–6.

[14] C. Loyez, K. Carpentier, I. Sourikopoulos, and F. Danneville, "Subthreshold neuromorphic devices for spiking neural networks applied to embedded A.I," in *Proc. 19th IEEE Int. New Circuits Syst. Conf. (NEWCAS)*, Jun. 2021, pp. 1–4.

[15] P. A. Merolla, J. V. Arthur, R. Alvarez-Icaza, A. S. Cassidy, J. Sawada, F. Akopyan, B. L. Jackson, N. Imam, C. Guo, Y. Nakamura, B. Brezzo, I. Vo, S. K. Esser, R. Appuswamy, B. Taba, A. Amir, M. D. Flickner, W. P. Risk, R. Manohar, and D. S. Modha, "A million spiking-neuron integrated circuit with a scalable communication network and interface," *Science*, vol. 345, no. 6197, pp. 668–673, Aug. 2014.

[16] M. Davies, N. Srinivasa, T. H. Lin, G. Chinya, Y. Cao, S. H. Choday, G. Dimou, P. Joshi, N. Imam, S. Jain, and Y. Liao, "Loihi: A neuromorphic manycore processor with on-chip learning," *IEEE Micro*, vol. 38, no. 1, pp. 82–99, Jan. 2018.

[17] A. Neckar, S. Fok, B. V. Benjamin, T. C. Stewart, N. N. Oza, A. R. Voelker, C. Eliasmith, R. Manohar, and K. Boahen, "Braindrop: A mixed-signal neuromorphic architecture with a dynamical systems-based programming model," *Proc. IEEE*, vol. 107, no. 1, pp. 144–164, Jan. 2019.

[18] P. M. Ferreira, J. Nebhen, G. Klisnick, and A. Benlarbi-Delai, "Neuromorphic analog spiking-modulator for audio signal processing," *Anal. Integr. Circuits Signal Process.*, vol. 106, no. 1, pp. 261–276, Jan. 2021.

[19] A. Zhang, X. Li, Y. Gao, and Y. Niu, "Event-driven intrinsic plasticity for spiking convolutional neural networks," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, no. 5, pp. 1986–1995, May 2022.

[20] C. Fager, T. Eriksson, F. Barradas, K. Hausmair, T. Cunha, and J. C. Pedro, "Linearity and efficiency in 5G transmitters: New techniques for analyzing efficiency, linearity, and linearization in a 5G active antenna transmitter context," *IEEE Microw. Mag.*, vol. 20, no. 5, pp. 35–49, May 2019.

[21] A. Katz, J. Wood, and D. Chokola, "The evolution of PA linearization: From classic feedforward and feedback through analog and digital predistortion," *IEEE Microw. Mag.*, vol. 17, no. 2, pp. 32–40, Feb. 2016.

[22] J. Kim and K. Konstantinou, "Digital predistortion of wideband signals based on power amplifier model with memory," *Electron. Lett.*, vol. 37, no. 23, pp. 1417–1418, Nov. 2001.

[23] D. R. Morgan, Z. Ma, J. Kim, M. G. Zierdt, and J. Pastalan, "A generalized memory polynomial model for digital predistortion of RF power amplifiers," *IEEE Trans. Signal Process.*, vol. 54, no. 10, pp. 3852–3860, Oct. 2006.

[24] A. Zhu, J. C. Pedro, and T. J. Brazil, "Dynamic deviation reduction-based Volterra behavioral modeling of RF power amplifiers," *IEEE Trans. Microw. Theory Techn.*, vol. 54, no. 12, pp. 4323–4332, Dec. 2006.

[25] A. Zhu, "Decomposed vector rotation-based behavioral modeling for digital predistortion of RF power amplifiers," *IEEE Trans. Microw. Theory Techn.*, vol. 63, no. 2, pp. 737–744, Feb. 2015.

[26] S. Wang, M. A. Hussein, O. Venard, and G. Baudoin, "Optimal sizing of two-stage cascaded sparse memory polynomial model for high power amplifiers linearization," *IEEE Trans. Microw. Theory Techn.*, vol. 66, no. 9, pp. 3958–3965, Sep. 2018.

[27] A. Fischer-Buhner, A. Brihuega, L. Anttila, M. D. Gomony, and M. Valkama, "Mixture of experts neural network for modeling of power amplifiers," in *IEEE MTT-S Int. Microw. Symp. Dig.*, Jun. 2022, pp. 510–513.

[28] A. Brihuega, M. Abdelaziz, L. Anttila, Y. Li, A. Zhu, and M. Valkama, "Mixture of experts approach for piecewise modeling and linearization of RF power amplifiers," *IEEE Trans. Microw. Theory Techn.*, vol. 70, no. 1, pp. 380–391, Jan. 2022.

[29] A. Brihuega, L. Anttila, and M. Valkama, "Frequency-domain digital predistortion for OFDM," *IEEE Microw. Wireless Compon. Lett.*, vol. 31, no. 6, pp. 816–818, Jun. 2021.
[30] S. Wang, P. M. Ferreira, and A. Benlarbi-Delai, "Behavioral modeling of nonlinear power amplifiers using spiking neural networks," in *Proc. 20th IEEE Interregional NEWCAS Conf. (NEWCAS)*, Jun. 2022, pp. 495–499.
[31] P. N. Landin, S. Gustafsson, C. Fager, and T. Eriksson, "WebLab: A web-based setup for PA digital predistortion and characterization [application notes]," *IEEE Microw. Mag.*, vol. 16, no. 1, pp. 138–140, Feb. 2015.
[32] A. Cappy, *Neuro-Inspired Information Processing* (Electronics Engineering Series). Hoboken, NJ, USA: Wiley, 2020.
[33] A. L. Hodgkin and A. F. Huxley, "A quantitative description of membrane current and its application to conduction and excitation in nerve," *J. Physiol.*, vol. 117, no. 4, pp. 500–544, Aug. 1952.
[34] C. Morris and H. Lecar, "Voltage oscillations in the barnacle giant muscle fiber," *Biophys. J.*, vol. 35, no. 1, pp. 193–213, Jul. 1981.
[35] E. M. Izhikevich, "Simple model of spiking neurons," *IEEE Trans. Neural Netw.*, vol. 14, no. 6, pp. 1569–1572, Nov. 2003.
[36] M. A. Nahmias, B. J. Shastri, A. N. Tait, and P. R. Prucnal, "A leaky integrate-and-fire laser neuron for ultrafast cognitive computing," *IEEE J. Sel. Topics Quantum Electron.*, vol. 19, no. 5, pp. 1–12, Sep. 2013.
[37] T. Soupizet, Z. Jouni, J. F. Sulzbach, A. Benlarbi-Delai, and P. M. Ferreira, "Deep neural network feasibility using analog spiking neurons," in *Proc. 35th SBC/SBMicro/IEEE/ACM Symp. Integr. Circuits Syst. Design (SBCCI)*, Aug. 2022, pp. 1–6.
[38] G.-Q. Bi and M.-M. Poo, "Synaptic modifications in cultured hippocampal neurons: Dependence on spike timing, synaptic strength, and postsynaptic cell type," *J. Neurosci.*, vol. 18, no. 24, pp. 10464–10472, Dec. 1998.
[39] R. Jevtic and C. Carreras, "Power estimation of embedded multiplier blocks in FPGAs," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 18, no. 5, pp. 835–839, May 2010.
[40] J. Wood, *Behavioral Modeling and Linearization of RF Power Amplifiers*. Norwood, MA, USA: Artech House, 2014.
[41] M. A. Hussein, V. A. Bohara, and O. Venard, "Multi-stage digital predistortion based on indirect learning architecture," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, May 2013, pp. 6093–6097.
[42] S. Wang, W. Cao, R. Hou, and T. Eriksson, "A digital predistortion for concurrent dual-band power amplifier linearization based on periodically nonuniform sampling theory," *IEEE Trans. Microw. Theory Techn.*, vol. 70, no. 1, pp. 466–475, Jan. 2022.
[43] S. A. Bassam, M. Helaoui, and F. M. Ghannouchi, "2-D digital predistortion (2-D-DPD) architecture for concurrent dual-band transmitters," *IEEE Trans. Microw. Theory Techn.*, vol. 59, no. 10, pp. 2547–2553, Oct. 2011.
[44] J. Chani-Cahuana, P. N. Landin, C. Fager, and T. Eriksson, "Iterative learning control for RF power amplifier linearization," *IEEE Trans. Microw. Theory Techn.*, vol. 64, no. 9, pp. 2778–2789, Sep. 2016.
[45] S. Wang, M. A. Hussein, O. Venard, and G. Baudoin, "Impact of the normalization gain of digital predistortion on linearization performance and power added efficiency of the linearized power amplifier," in *Proc. 12th Eur. Microw. Integr. Circuits Conf. (EuMIC)*, Oct. 2017, pp. 310–313.
[46] F. Ponulak and A. Kasinski, "Supervised learning in spiking neural networks with ReSuMe: Sequence learning, classification, and spike shifting," *Neural Comput.*, vol. 22, no. 2, pp. 467–510, Feb. 2010.
[47] C. Yu, L. Guan, E. Zhu, and A. Zhu, "Band-limited Volterra series-based digital predistortion for wideband RF power amplifiers," *IEEE Trans. Microw. Theory Techn.*, vol. 60, no. 12, pp. 4198–4208, Dec. 2012.
[48] A. S. Tehrani, H. Cao, S. Afsardoost, T. Eriksson, M. Isaksson, and C. Fager, "A comparative analysis of the complexity/accuracy tradeoff in power amplifier behavioral models," *IEEE Trans. Microw. Theory Techn.*, vol. 58, no. 6, pp. 1510–1520, Jun. 2010.
[49] E. Lazaridis, E. M. Drakakis, and M. Barahona, "Full analogue electronic realisation of the Hodgkin–Huxley neuronal dynamics in weak-inversion CMOS," in *Proc. 29th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc.*, Aug. 2007, pp. 1200–1203.
[50] (2022). *Brian 2 Documentation*. [Online]. Available: https://brian2.readthedocs.io/en/stable/
[51] S. Wang, M. A. Hussein, O. Venard, and G. Baudoin, "A novel algorithm for determining the structure of digital predistortion models," *IEEE Trans. Veh. Technol.*, vol. 67, no. 8, pp. 7326–7340, Aug. 2018.
[52] S. Wang, W. Cao, and T. Eriksson, "Identification methods with different digital predistortion models for power amplifiers with strong nonlinearity and memory effects," in *IEEE MTT-S Int. Microw. Symp. Dig.*, Sep. 2020, pp. 1–3.
[53] M. Abdelaziz, L. Anttila, A. Kiayani, and M. Valkama, "Decorrelation-based concurrent digital predistortion with a single feedback path," *IEEE Trans. Microw. Theory Techn.*, vol. 66, no. 1, pp. 280–293, Jan. 2018.

SIQI WANG received the B.S. degree from the Huazhong University of Science and Technology, Wuhan, China, in 2012, the M.S. degree from the University of Paris-Sud, Orsay, France, in 2014, and the Ph.D. degree from the University of Paris-Est Marne-La-Vallée, Champs-sur-Marne, France, in 2018. He was a Researcher Fellow with GeePs, Centralesupélec, Gif-sur-Yvette, France, from 2018 to 2019. He was a Research Fellow with the Chalmers University of Technology, Gothenburg, Sweden, from 2019 to 2021. He has been an Associate Professor with Sorbonne University, Paris, France, since 2021. He has authored or coauthored more than 30 journal articles and conference papers. His research interests include neuromorphic circuit modeling, spiking neural networks, the Internet of Things (IoT), federated learning, energy efficiency optimization, digital predistortion, and massive multiple input multiple output (MIMO).

PIETRO MARIS FERREIRA (Senior Member, IEEE) received the B.Eng. degree (cum lauda) in electronics and computer engineering and the M.Sc. degree in microelectronics from the Federal University of Rio de Janeiro (UFRJ), Brazil, in 2006 and 2008, respectively, and the Ph.D. degree in communications and electronics from Télécom Paris, IPP, France. Researching high-performance high-reliability circuits and systems, he joined the IM2NP Laboratory (UMR CNRS 7334) for one year, and IEMN Laboratory (UMR CNRS 8520) for two years during his tenure track. Since 2014, he has been an Associate Professor with Université Paris-Saclay, CentraleSupélec, GeePs (UMR 8507), France. In 2019, he defended his Research Direction Project (HDR) in physics from Université Paris-Saclay. For about 20 years, he has been a member of IEEE Circuits and Systems Society, a volunteer on international conferences organization, the Counselor of IEEE STB Paris-Saclay, and an active member on R8, Section France. His research interests include design methodologies for harsh environments, microwave, and ultra-low power integrated circuits. Recent projects envision the Internet of Things industry considering IA-edge and reliability.

AZIZ BENLARBI-DELAI received the Ph.D. degree in electrical engineering and the Habilitation à Diriger des Recherches (H.D.R.) ès Sciences Physiques from the University of Lille I, in 1992 and 2002, respectively. From 1992 to 2006, he was an Assistant Professor with the University of Lille I and was mainly involved, as a Researcher in microwave and millimetre wave devices and systems for communication and localization with the Institute of Electronic Microelectronic and Nanotechnology (IEMN). He also investigated others fields of research dealing with ultra-fast sampling devices, plasmonic structures, and micro and nano technology. He is currently a full time Professor in electrical engineering with Sorbonne University and the Head of the GeePs Laboratory, Electronics Department. Former, he was Director of the Laboratory of Electronics and Electromagnetism (L2E) and a member of the Strategic Committee of the Labex SMART. His research activities deal with channel modeling for wireless systems and is now involved in neuromorphic design for ultra-low power systems. He is the author of 136 publications and communications and the holder of two patents. He participates to several technical program committee of international conferences.