Particle Swarm Optimization Assisted B-spline Neural Network Based Predistorter Design to Enable Transmit Precoding for Nonlinear MIMO Downlink

Sheng Chen[a,b], Soon Xin Ng[a], Emad Khalaf[b], Ali Morfeq[b], Naif Alotaibi[b]

[a] *School of Electronics and Computer Science, University of Southampton, Southampton SO17 1BJ, UK*

[b] *Electrical and Computer Engineering Department, Faculty of Engineering, King Abdulaziz University, Jeddah 21589, Saudi Arabia*

Abstract

For the multiple-input multiple-output (MIMO) downlink employing high-order quadrature amplitude modulation signaling and with nonlinear high power amplifiers (HPAs) at base station transmitter, the existing precoding designs relying on the linear MIMO channel can no longer work. We propose an efficient and accurate predistorter design to enable transmit precoding for nonlinear MIMO downlink. Specifically, we obtain the closed-form least squares estimates of the nonlinear HPA's amplitude and phase response using two B-spline neural networks during training. The estimated HPA's phase response automatically yields the estimate of the predistorter's phase response. Based on the B-spline neural network estimate of the HPA's amplitude response, we construct a B-spline neural network model for the predistorter amplitude response, and we adopt a particle swarm optimization (PSO) algorithm to solve this highly nonlinear optimization problem. Using our accurate predistorter estimate to pre-compensate for the nonlinear distortions of the transmit HPAs, a standard full-digital transmit precoding design can readily be adopted to combat the MIMO channel interference. A simulation study is conducted to demonstrate the effectiveness of our proposed PSO assisted predistorter design.

Email addresses: sqc@ecs.soton.ac.uk (Sheng Chen), sxn@ecs.soton.ac.uk (Soon Xin Ng), ekhalaf@kau.edu.sa (Emad Khalaf), morfeq@kau.edu.sa (Ali Morfeq), ndalotabi@kau.edu.sa (Naif Alotaibi)

Keywords: Particle swarm optimization, B-spline neural network, nonlinear inversion, nonlinear high power amplifier, predistorter, multi-input multi-output, downlink precoding

1. Introduction

Two of the three cornerstones or usage scenarios in the fifth generation (5G) mobile network are enhanced mobile broadband (eMBB) and massive machine type communications (mMTC) [1]. Multiple-input multiple-output (MIMO) technology has been recognized as a promising component for implementing 5G by both academia and industry, owing to its capability of significantly increasing the reliability and/or bandwidth efficiency of communication systems [2, 3, 4, 5, 6, 7, 8, 9, 10, 11]. In particular, the spatial-domain non-orthogonal multiple access (NOMA) with the aid of MIMO technology plays a critical role in supporting the massive increase in connected devices with the limited frequency-time resources.

In the literature, most existing MIMO system designs, including all the best known linear MIMO transceiver designs [12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23] and nonlinear MIMO transceiver designs [24, 25, 26, 27, 28], adopt the linear MIMO channel. However, it is well known that the linear MIMO channel is only valid when the transmitter high power amplifier (HPA) operates within its linear dynamic range. But practical HPAs exhibit nonlinear saturation characteristics [29, 30, 31, 32, 33], and whether the linear channel assumption holds depends on the transmit signal's peak-to-average power ratio (PAPR) as well as the average transmit power. For the modulation constellations with unity PAPR, such as phase shift keying, HPA does not cause amplitude distortion and the phase shift of the HPA's output is constant for all the symbol points. In such scenarios, a linear MIMO channel is valid. In order to meet the demand of massive increase in throughput for supporting eMBB, high-order quadrature amplitude modulation (QAM) signaling [34] has to be adopted, which exhibits high PAPR and imposes high average transmit power. Consequently, the non-

linear distortion of the transmitter HPA becomes serious, and the assumption of linear MIMO channel no longer holds. In such situations, the existing MIMO system designs based on the linear MIMO channel do not work. Moreover, the classical means of avoiding the nonlinearity of transmitter HPA, namely, output back-off (OBO), may not be applicable. This is because for high PAPR signaling, the OBO must be very severe to be effective but such a large OBO cannot meet the required link power budge, that is, it cannot meet the high average transmit power requirement for high-order QAM signaling.

Recently, we have proposed an effective nonlinear multiuser detection design for NOMA multiuser nonlinear MIMO uplink employing high-order QAM signaling and with nonlinear transmit HPAs at mobile users (MUs) [35]. However, there exists no nonlinear MIMO downlink design in the open literature. Against this background, in this work, we focus our attention on nonlinear MIMO downlink employing high-order QAM signaling and with nonlinear transmit HPAs at base station (BS), and we propose a novel and efficient predistorter design to pre-compensate the nonlinear transmit HPAs so that the standard transmit precoding can still be used for nonlinear MIMO downlink.

In downlink, the BS transmitter has sufficient resource to implement a predistorter for pre-compensating the nonlinear distortions of transmit HPAs. In the literature, there exist various predistorter designs [36, 37, 38, 39, 40, 41, 42, 43]. However, none of these predistorters are specifically designed for NOMA multiuser MIMO downlink applications. In this paper, we propose a very efficient and accurate predistorter design based on B-spline neural network for NOMA multiuser nonlinear MIMO downlink. More specifically, we estimate the HPA's amplitude response and phase response using two B-spline neural network models in training, and the two B-spline models' parameter vectors can readily be obtained in the closed-form least squares (LS) solutions. This yields very accurate B-spline neural network based estimates of the HPA's amplitude response and phase response. Since the predistorter's phase response should cancel the HPA's phase response, the estimated predistorter's phase response is the negative of the B-spline HPA phase response estimate. We then

design another B-spline neural network model for the predistorter's amplitude response relying on the B-spline HPA amplitude response estimate already obtained. This is a highly nonlinear optimization problem. Although it can be solved with a gradient based algorithm, we propose to solve this nonlinear design with particle swarm optimization (PSO) [44, 45, 46, 47, 48, 49, 50, 51] in order to obtain a much more accurate predistorter estimate. Unlike the most recent B-spline predistorter design of [42] which requires to find the amplitude of the predistorter output for every transmitted signal point using the iterative root finding procedure during data transmission and hence it is unsuitable for MIMO downlink application, our proposed PSO assisted B-spline neural network based predistorter design constructs an accurate B-spline model of the predistorter prior to data transmission and it offers the first practical B-spline parameterized predistorter for nonlinear MIMO transmitter. With this accurate predistorter estimate to compensate for the HPA's nonlinear distortion, the BS can employ a standard full-digital precoding design, such as zero-forcing (ZF) precoding design, to pre-remove the MIMO downlink channel interference.

The remaining of the paper is organized as follows. Section 2 presents our application background, namely, nonlinear MIMO downlink, while Section 3 details our novel PSO assisted B-spline neural network based predistorter design. An achievable performance of our novel predistorter assisted nonlinear MIMO downlink is extensively evaluated in Section 4. Our conclusions are drawn in Section 5.

2. Nonlinear MIMO Downlink

This section present our application background, specifically, the NOMA multiuser nonlinear MIMO downlink communication system.

2.1. MIMO Downlink Channel

We consider the MIMO downlink shown in Figure 1, where the BS is equipped with L antennas to support M ($\leq L$) single-antenna MUs using the same

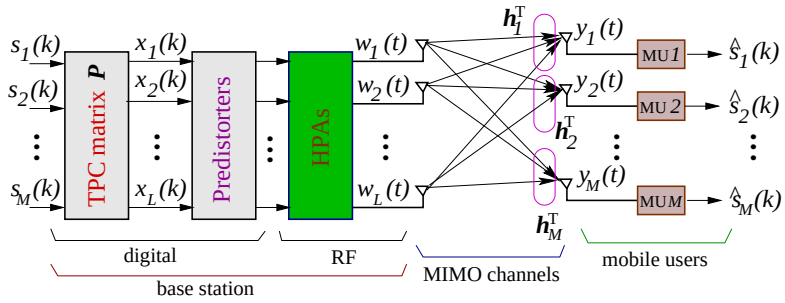


Figure 1: MIMO downlink with nonlinear transmit HPAs where BS employs L antennas to support M single-antenna mobile users based on spatial-domain non-orthogonal multiple access.

frequency-time resource block. Denote the transmit signal vector from the BS's antenna array as $\boldsymbol{w}(t) = \big[w_1(t)\ w_2(t)\cdots w_L(t)\big]^{\mathrm{T}}$ and its baseband equivalent sampled version as $\boldsymbol{w}(k) = \big[w_1(k)\ w_2(k)\cdots w_L(k)\big]^{\mathrm{T}}$. Further collect the received signals at the M MUs as the vector $\boldsymbol{y}(t) = \big[y_1(t)\ y_2(t)\cdots y_M(t)\big]^{\mathrm{T}}$, and denote its baseband equivalent sampled version as $\boldsymbol{y}(k) = \big[y_1(k)\ y_2(k)\cdots y_M(k)\big]^{\mathrm{T}}$. Then the MIMO downlink channel can be represented by the following well-known baseband MIMO channel model

$$\boldsymbol{y}(k) = \big[\boldsymbol{h}_1\ \boldsymbol{h}_2\cdots\boldsymbol{h}_M\big]^{\mathrm{T}}\boldsymbol{w}(k) + \boldsymbol{n}(k) = \boldsymbol{H}^{\mathrm{T}}\boldsymbol{w}(k) + \boldsymbol{n}(k). \tag{1}$$

Here $\boldsymbol{n}(k) \in \mathbb{C}^M \sim \mathcal{CN}\big(\boldsymbol{0}_M, \sigma_n^2\boldsymbol{I}_M\big)$ is the downlink additive white Gaussian noise (AWGN) vector with the M-dimensional zero mean vector $\boldsymbol{0}_M$ and the covariance matrix $\sigma_n^2\boldsymbol{I}_M$ in which $\boldsymbol{I}_M$ is the $M \times M$ identity matrix, and $\boldsymbol{H} = \big[\boldsymbol{h}_1\ \boldsymbol{h}_2\cdots\boldsymbol{h}_M\big]$ in which

$$\boldsymbol{h}_i = \big[h_{1,1}\ h_{1,2}\cdots h_{1,L}\big]^{\mathrm{T}},\ 1 \le i \le M, \tag{2}$$

is the downlink channel vector linking the L BS antennas to the mth MU. The channel coefficient $h_{m,l} \in \mathbb{C}$ for the channel linking the lth BS antenna to the mth MU, where $1 \le l \le L$ and $1 \le m \le M$, is drawn from the complex-value Gaussian distribution $\mathcal{CN}(0, 1)$.

2.2. Transmit Precoding

The M MUs rely on the BS to perform the transmit precding (TPC) to pre-compensate the MIMO channel interference so that the mth MU's received signal $y_m(k)$ is a sufficient statistic for estimating its data symbol $s_m(k)$. Under the condition that the HPAs are operating within their linear dynamic ranges and hence the predistorters are not required, the baseband MIMO channel model (1) can be equivalently expressed as the following commonly known form

$$\boldsymbol{y}(k) = \boldsymbol{H}^{\mathrm{T}}\boldsymbol{x}(k) + \boldsymbol{n}(k), \tag{3}$$

where $\boldsymbol{x}(k) \in \mathbb{C}^L$ is the digital transmit precoder output vector. Specifically, the BS can employ the standard full digital TPC technique based on the well-known ZF design which is capable of completely removing the MIMO downlink channel interference. Given the MIMO downlink channel matrix $\boldsymbol{H}$, the full-digital ZF TPC is defined as

$$\boldsymbol{x}(k) = \sqrt{\lambda}\boldsymbol{P}_{\mathrm{ZF}}\boldsymbol{s}(k), \tag{4}$$

where the full-digital ZF TPC matrix $\boldsymbol{P}_{\mathrm{ZF}} \in \mathbb{C}^{L\times M}$ is given by

$$\boldsymbol{P}_{\mathrm{ZF}} = \boldsymbol{H}^{*}\left(\boldsymbol{H}^{\mathrm{T}}\boldsymbol{H}^{*}\right)^{-1}, \tag{5}$$

and the normalization factor λ is given by

$$\lambda = \frac{1}{E_s \mathrm{tr}\left\{\boldsymbol{P}_{\mathrm{ZF}}\boldsymbol{P}_{\mathrm{ZF}}^{\mathrm{H}}\right\}}, \tag{6}$$

in which E_s is the average power of each MU's data and $\mathrm{tr}\{\cdot\}$ is the matrix trace operator. We will discuss why needs the normalization factor (6) later.

Obviously, the knowledge of the downlink channel matrix $\boldsymbol{H}^{\mathrm{T}}$ is required at the BS to compute the TPC matrix. In the networks based on frequency division duplexing (FDD) protocol, the BS needs to transmit the training signal to the M MUs for them to acquire their respective channel vectors $\boldsymbol{h}_m$, $1 \le m \le M$.

After obtaining the estimated $\widehat{\boldsymbol{h}}_m$, $1 \leq m \leq M$, the MUs quantize the channel estimates and feed back the quantized channel estimates to the BS. The estimated channel matrix $\widehat{\boldsymbol{H}}^{\mathrm{T}}$ that the BS has is therefore inherently erroneous due to quantization and delay errors. In the so-called time division duplexing (TDD) network, the uplink channel and downlink channel are reciprocal. Hence, the BS can estimate the uplink channel matrix and exploit the reciprocal property to obtain downlink channel estimate $\widehat{\boldsymbol{H}}^{\mathrm{T}}$. Owing to the mismatch in the uplink and downlink radio frequency (RF) chains, the reciprocal property cannot be exact and hence, the downlink channel estimate $\widehat{\boldsymbol{H}}^{\mathrm{T}}$ is also erroneous.

Therefore, the BS does not have the perfect downlink channel matrix. To model the channel estimation error, we express the channel estimate

$$\widehat{h}_{m,l} = h_{m,l} + \varepsilon,\ 1 \leq m \leq M, 1 \leq l \leq L, \tag{7}$$

where both the real and imaginary parts of ε, denoted as $\Re(\varepsilon)$ and $\Im(\varepsilon)$, are the uniformly distributed random variables in $\big[-\sigma_\varepsilon,\ \sigma_\varepsilon\big]$. The case of $\sigma_\varepsilon = 0$ corresponds to the perfect channel knowledge. The BS uses the erroneous channel estimate $\widehat{\boldsymbol{H}}^{\mathrm{T}}$ to calculate the TPC matrix.

2.3. Nonlinear High Power Amplifier

However, for the high PAPR signaling, such as the high-order QAM considered in this paper, the commonly used HPA at transmitter exhibits serious nonlinear saturation distortion. Consequently, the linear MIMO channel model (3) is no longer valid. More specifically, the transmitted signal vector $\boldsymbol{w}(k)$ is no longer linearly proportional to the precoder output vector $\boldsymbol{x}(k)$. Rather, owing to the HPAs' nonlinearity, $\boldsymbol{w}(k)$ is a nonlinear transformation of $\boldsymbol{x}(k)$. As a result, the TPC matrix distorted by the nonlinearity of HPA becomes incapable of compensating for the MIMO channel interference. To model nonlinear HPA, note that a complex-valued number $x \in \mathbb{C}$ can be represented either in rectangular form of $x = \Re(x) + \mathrm{j}\Im(x)$, or in polar form of $x = r_x \exp(\mathrm{j}\phi_x)$, in which r_x is the magnitude of x and ϕ_x the phase of x. Without loss of generality,

omit the antenna index l in the discussion. HPA employed in wireless systems is typically the solid state NEC GaAs power amplifier [32, 33], which exhibits nonlinear saturation characteristics. In the equivalent baseband discrete-time domain, a HPA output signal $w(k)$ from a BS antenna can be expressed as

$$w(k) = \Xi\left(v(k)\right) = A(r_v(k)) \exp\left(\mathrm{j}\big(\Upsilon(r_v(k)) + \phi_v(k)\big)\right), \tag{8}$$

where $\Xi(\cdot)$ denotes the HPA's nonlinear mapping and $v(k)$ is the input to the HPA. Hence the complex-valued HPA's mapping $\Xi(\cdot)$ is defined by its amplitude response $A(r)$ and phase response $\Upsilon(r)$, given respectively by [32, 33]

$$A(r) = \frac{g_a r}{\left(1 + \left(\frac{g_a r}{A_{\text{sat}}}\right)^{2\beta_a}\right)^{\frac{1}{2\beta_a}}}, \tag{9}$$

$$\Upsilon(r) = \frac{\alpha_\phi r^{q_1}}{1 + \left(\frac{r}{\beta_\phi}\right)^{q_2}} \text{ [degree]}, \tag{10}$$

where r denotes the amplitude of the input to the HPA, g_a is the small signal's gain, β_a is the smoothness factor and A_{sat} is the saturation level, while the parameters of the phase response, α_ϕ, β_ϕ, q_1 and q_2, are adjusted to match the specific amplifier's characteristics [32, 33]. The operating status of the HPA is specified by the OBO, which is defined as the ratio of the maximum output power $P_{\max}$ of the HPA to the average output power P_{aop} of the HPA output signal, given by

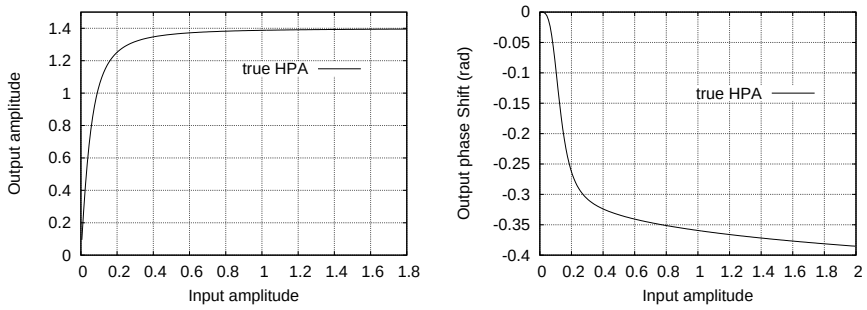
$$\text{OBO} = 10 \cdot \log_{10} \frac{P_{\max}}{P_{\text{aop}}} \text{ [dB]}. \tag{11}$$

The smaller OBO is, the deeper the HPA is into the nonlinear saturation region.

With the HPA's parameters as specified in the standards [32, 33], which are

$$g_a = 19,\ \beta_a = 0.81,\ A_{\text{sat}} = 1.4;\ \alpha_\phi = -48000,\ \beta_\phi = 0.123,\ q_1 = 3.8,\ q_2 = 3.7. \tag{12}$$

Figure 2 depicts its amplitude response and phase response. Clearly, the output saturation amplitude is $A_{\text{sat}} = 1.4$, which occurs theoretically at the saturation



(a) (b)

Figure 2: Nonlinear HPA with the parameters given by (12): (a) amplitude response, and (b) phase response.

input amplitude $r_{\text{sat}} = 1.4$, that is, $A(r) = 1.4$ for $r \geq r_{\text{sat}}$. Consider the case that the HPA is designed to operate in a large OBO value of 5 dB for transmitting 64QAM data. Then the average 64QAM symbol amplitude is approximately 0.06, while the peak 64QAM symbol amplitude is approximately 0.09. The corresponding average amplitude of the HPA's output is approximately 0.8 and the peak amplitude of the HPA's output is approximately 1.0. Even under such a large OBO, the nonlinear distortions of the HPA is noticeable.

For the MIMO downlink, the TPC must be applied to overcome the MIMO channel interference. Consequently, the nonlinear distortion of HPAs is even more serious. This is because the precoded signal $x_l(k)$ is a linear combination of the M high-order QAM data $s_m(k)$ for $1 \leq m \leq M$, and thus the PAPR of $x_l(k)$ is much higher than each MU's data $s_m(k)$. This further amplifies the nonlinear distortion of HPA.

3. Proposed Predistorter Design

3.1. Ideal Predistorter Response

Nonlinearity of the HPA renders the precoding ineffective. It is therefore vital to design the predistorter that can pre-compensate the nonlinear distortions of the HPA. Let the ideal complex-valued predistorter's nonlinear mapping be $\Omega(\cdot)$. Further denote this ideal predistorter's amplitude response and phase response as $B(\cdot)$ and $\Psi(\cdot)$, respectively. Then given the input x, the output of

the predistorter v is given by [41, 42]

$$v = \Omega(x) = B(r_x)\exp\big(\mathrm{j}\big(\Psi(B(r_x)) + \phi_x\big)\big). \tag{13}$$

The ideal or perfect predistorter should satisfy the following conditions [41, 42]:

$$A(B(r_x)) = \begin{cases} r_x, & r_x \le r_{\mathrm{sat}}, \\ r_{\mathrm{sat}}, & r_x > r_{\mathrm{sat}}, \end{cases} \tag{14}$$

$$\Psi(B(r_x)) + \Upsilon(B(r_x)) = 0. \tag{15}$$

Note that the input to the HPA is the output of the predistorter, and the predistorter can only achieve the linearization for $0 \le r_x \le r_{\mathrm{sat}}$.

With this predistorter to compensate for the HPA's nonlinear distortion, the BS can employ a standard full-digital precoding design, such as the ZF precoding of (4)-(6), to pre-remove the MIMO downlink channel interference. Specifically, denote the output vector of the L idealized predistorters as

$$\boldsymbol{v}(k) = \boldsymbol{\Omega}(\boldsymbol{x}(k)) = \big[\Omega(x_1(k))\ \Omega(x_2(k))\cdots\Omega(x_L(k))\big]^{\mathrm{T}}, \tag{16}$$

and the output vector of the L HPAs as

$$\boldsymbol{w}(k) = \boldsymbol{\Xi}(\boldsymbol{v}(k)) = \big[\Xi(v_1(k))\ \Xi(v_2(k))\cdots\Xi(v_L(k))\big]^{\mathrm{T}}. \tag{17}$$

Then MIMO channel model (1) can be re-expressed as

$$\begin{aligned} \boldsymbol{y}(k) =& \frac{1}{\sqrt{\lambda}}\boldsymbol{H}^{\mathrm{T}}\boldsymbol{\Xi}\Big(\boldsymbol{\Omega}\Big(\sqrt{\lambda}\boldsymbol{P}_{\mathrm{ZF}}\boldsymbol{s}(k)\Big)\Big) + \frac{1}{\sqrt{\lambda}}\boldsymbol{n}(k) \\ =& \frac{1}{\sqrt{\lambda}}\boldsymbol{H}^{\mathrm{T}}\boldsymbol{\Xi}\Big(\boldsymbol{\Omega}\Big(\sqrt{\lambda}\boldsymbol{P}_{\mathrm{ZF}}\boldsymbol{s}(k)\Big)\Big) + \widetilde{\boldsymbol{n}}(k), \end{aligned} \tag{18}$$

where the AWGN $\widetilde{\boldsymbol{n}}(k) \sim \mathcal{CN}\big(\boldsymbol{0}_M, \frac{\sigma_n^2}{\lambda}\boldsymbol{I}_M\big)$. Observing from the ideal amplitude response (14) of the combined HPA and predistorter, it can be concluded that

if the precoded data points $x_l(k)$ have the magnitudes $r_x(k) \leq r_{\text{sat}}$, the predistorter completely linearizes the HPA, and the nonlinear MIMO model (18) is equivalent to the liner MIMO model (3).

Remark 1. *It is necessary to apply the scaling or normalization factor in the precoding operation (4). Otherwise the magnitudes $r_x(k)$ of many precoded data points $x_l(k)$ will become larger than r_{sat}, which leads to high bit error rate (BER) floor even with the idealized predistorters. A consequence of this scaling is that the BS transmitter needs to send $\sqrt{\lambda}$ to the M MU receivers, and each MU needs to 'un-scale' its received signal by $\frac{1}{\sqrt{\lambda}}$ as can be seen in (18).*

3.2. B-spline Neural Network Based Predistorter

The schematic diagram of the proposed predistorter design is depicted in Figure 3, where $\widehat{B}(\cdot)$ is an estimate of the predistorter's true amplitude response $B(\cdot)$, solved from $\widehat{A}(\widehat{B}(r)) = r$, in which $\widehat{A}(\cdot)$ denotes an estimate of the HPA's true amplitude response $A(\cdot)$, while $\widehat{\Upsilon}(\cdot)$ denotes an estimate of the HPA's true phase response $\Upsilon(\cdot)$. Since B-spline neural network is an effective means of nonlinear modeling [42, 43, 52, 53, 54, 55, 56], we adopt the B-spline modeling approach for estimating the HPA's true amplitude response $A(\cdot)$ and true phase response $\Upsilon(\cdot)$ as well as the predistorter's true amplitude response $B(\cdot)$.

3.2.1. B-spline Neural Network

To model a generic real-value nonlinearity $f(r)$ in the univariate of r, we use a B-spline neural network model with piecewise polynomial degree of P_d and N basis functions. This B-spline model is parametrized by the knot sequence

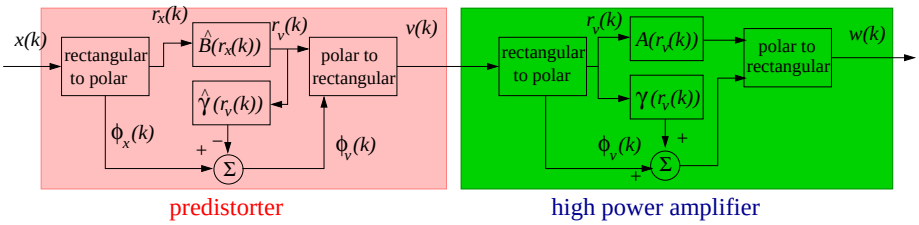


Figure 3: Schematic of proposed predistorter structure.

specified by $(N + P_d + 1)$ knot values, $\{R_0, R_1, \cdots, R_{N+P_d}\}$, with

$$\begin{aligned} R_0 < R_1 < \cdots < R_{P_d-2} < R_{P_d-1} = R_{\min} < R_{P_d} < \cdots \\ < R_N < R_{N+1} = R_{\max} < R_{N+2} < \cdots < R_{N+P_d}. \end{aligned} \tag{19}$$

At each end, there are $P_d - 1$ external knots that are outside the input region $\left[R_{\min},\ R_{\max}\right]$ and one boundary knot. Hence the number of internal knots is given by $N + 1 - P_d$. Given the set of predetermined knots (19), the set of the N B-spline basis functions

$$\mathcal{B}_i(r) = \mathcal{B}_i^{(P_d)}(r),\ 1 \le i \le N, \tag{20}$$

are formed using the De Boor recursion [57], which recursively computes

$$\mathcal{B}_l^{(0)}(r) = \begin{cases} 1, & \text{if } R_{l-1} \le r < R_l, \\ 0, & \text{otherwise,} \end{cases} \tag{21}$$

for $1 \le l \le N + P_d$, as well as

$$\mathcal{B}_l^{(p)}(r) = \frac{r - R_{l-1}}{R_{p+l-1} - R_{l-1}} \mathcal{B}_l^{(p-1)}(r) + \frac{R_{p+l} - r}{R_{p+l} - R_l} \mathcal{B}_{l+1}^{(p-1)}(r), \tag{22}$$

for $l = 1, \cdots, N + P_d - p$ and $p = 1, \cdots, P_d$. The estimate of $f(r)$ is readily expressed as the linear combiner of the N B-spline basis functions

$$\widehat{f}(r) = \sum\nolimits_{i=1}^{N} \mathcal{B}_i(r)\alpha_i, \tag{23}$$

where α_i for $1 \le i \le N$ are the B-spline neural network's weight parameters. An illustration of the De Boor recursion or the B-spline neural network structure is depicted in Figure 4.

Remark 2. *The polynomial degree $P_d = 4$ and the number of B-spline bases $N = 10$ are sufficient for accurately modeling an arbitrary nonlinear function $f(r)$. The computational complexity of the B-spline neural network is on the*

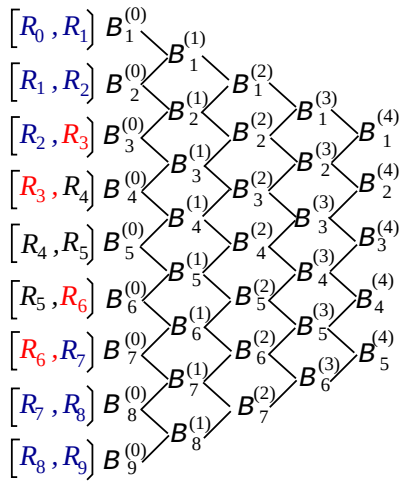


Figure 4: Visualisation of the B-spline neural network architecture for $P_d = 4$ and $N = 5$, where $R_{\min} = R_3$ and $R_{\max} = R_6$.

order of P_d^2, which is the same as the polynomial model with the polynomial degree P_d [43]. The B-spline estimator (23) has the well-known optimal robustness property [52, 53, 54]. Optimality of the B-spline model in terms of numerical stability is due to the convexity of its model bases, i.e., they are all positive and sum up to one. This optimal robustness property ensures that given the same level of structural (computational) complexity, the B-spline estimator will outperform any other non-robust linear-combining-nonlinear-bases estimator, such as the polynomial estimator, in modeling an unknown nonlinear function, in terms of estimation accuracy, particularly when the training input data are noisy.

We now demonstrate this optimality of the B-spline model. Assume that the real-valued true system is represented by the polynomial model of degree P_d as

$$y = \sum_{i=0}^{P_d} a_i x^i,$$

which can also be represented by the following B-spline model exactly

$$y = \sum_{i=1}^{N} b_i \mathcal{B}_i(x),$$

where $y, x \in \mathbb{R}$. Because the identification data are noisy, the estimated model coefficients are perturbed from their true values a_i to $\widehat{a}_i = a_i + \varepsilon_i$ for the polyno-

mial model, and from their true values b_i *to* $\widehat{b}_i = b_i + \varepsilon_i$ *for the B-spline model. Assume that all the estimation noises* ε_i *are bounded, namely,* $|\varepsilon_i| < \varepsilon_{\max}$*. The upper bound of* $|y - \widehat{y}|$ *for the B-spline model can be worked out to be*

$$|y - \widehat{y}| = \Big| \sum\nolimits_{i=1}^{N} b_i \mathcal{B}_i(x) - \sum\nolimits_{i=1}^{N} \widehat{b}_i \mathcal{B}_i(x) \Big| < \varepsilon_{\max} \Big| \sum\nolimits_{i=1}^{N} \mathcal{B}_i(x) \Big| = \varepsilon_{\max}.$$

Observe that the upper bound of the B-spline model output perturbation only depends on the upper bound of the perturbation noise, and it does not depend on the input value x*, the number of basis functions* N *or the polynomial degree* P_d*. This confirms that the B-spline model has the maximum numerical robustness, and this optimality of the B-spline model is due to the convexity of its model bases, that is, they are all positive and sum up to one. By contrast, the upper bound of* $|y - \widehat{y}|$ *for the polynomial model can be worked out to be*

$$|y - \widehat{y}| = \Big| \sum\nolimits_{i=0}^{P_d} a_i x^i - \sum\nolimits_{i=0}^{P_d} \widehat{a}_i x^i \Big| < \varepsilon_{\max} \Big| \sum\nolimits_{i=0}^{P_d} x^i \Big|.$$

Observe that the upper bound of the polynomial model output perturbation depends not only on the upper bound of the perturbation noise but also on the input value x *and the polynomial degree* P_d*. The higher the polynomial degree* P_d*, the more serious the polynomial model may be perturbed, a well-known drawback of using polynomial modeling.*

We further illustrate this optimality of the B-spline model using a simple example. Figure 5 (a) plots a quadratic polynomial function $y = 0.001x^2 - 0.02x + 0.1$ *defined over* $x \in [0,\ 20]$ *in solid line. Based on the knot sequence of* $\{-5, -4, 0, 20, 24, 25\}$*, this function is modeled as a quadratic B-spline model of* $y = 0.14\mathcal{B}_1(x) - 0.10\mathcal{B}_2(x) + 0.14\mathcal{B}_3(x)$*, which is depicted in Figure 5 (b) in solid line. We now draw three noises* ε_i*,* $1 \le i \le 3$*, from a uniformly distributed random number (UDRN) in* $[-0.0001,\ 0.0001]$*, and add them to the three parameters in the two models, respectively, to simulate the effects of the noise in identification. Figure 5 (a) and Figure 5 (b) depict the ten sets of the perturbed functions in dashed line generated by perturbing the two models, respectively. It*

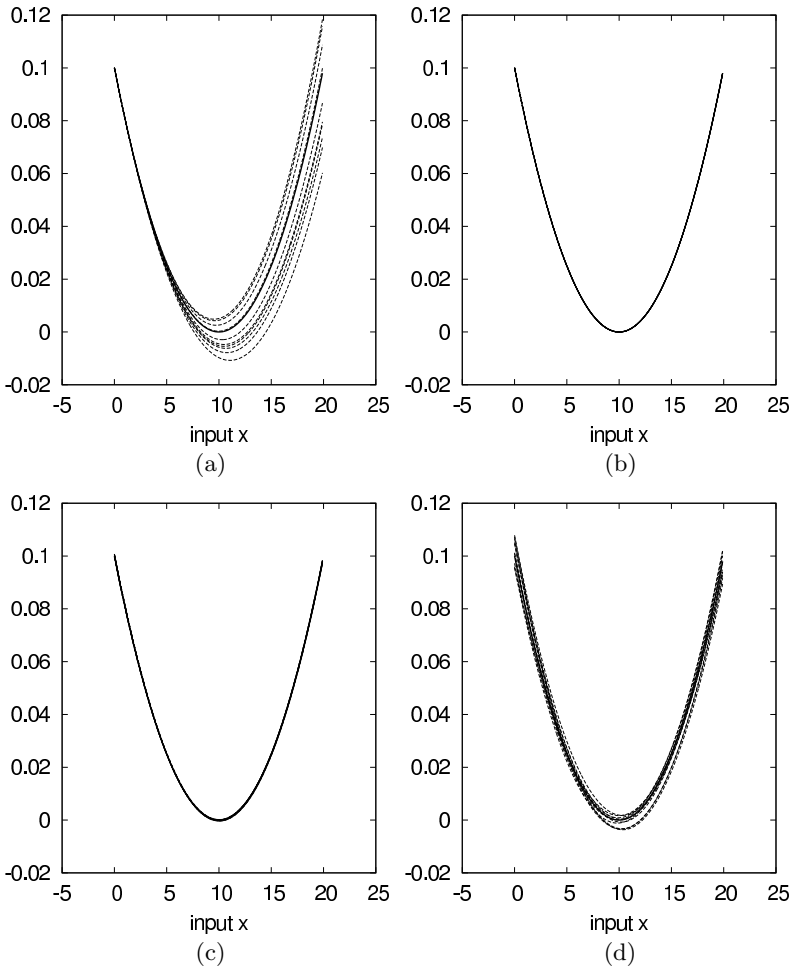


Figure 5: (a) The polynomial model with three perturbation noises drawn from a uniformly distributed random number (UDRN) in $[-0.0001,\ 0.0001]$, (b) the B-spline model with three perturbation noises drawn from a UDRN in $[-0.0001,\ 0.0001]$, (c) the B-spline model with three perturbation noises drawn from a UDRN in $[-0.001,\ 0.001]$, and (d) the B-spline model with three perturbation noises drawn from a UDRN in $[-0.01,\ 0.01]$.

can be clearly seen from Figure 5 (a) that the polynomial model is seriously perturbed, but there is no noticeable change at all in Figure 5 (b) for the quadratic B-spline model. To further demonstrate the maximum robustness of the B-spline model, we next draw three perturbation noises from a UDRN in $[-0.001,\ 0.001]$*, and add them to the three parameters of the B-spline model. Again, the B-spline model is hardly affected, as can be seen from Figure 5 (c). We then draw three perturbation noises from a UDRN in* $[-0.01,\ 0.01]$ *to add to the three B-spline parameters, and the results obtained are shown in Figure 5 (d). By comparing Figure 5 (a) and Figure 5 (d), it can be seen that despite of the fact that the strength of the perturbation noise added to the B-spline model coefficients is 100 times larger than that added to the polynomial model coefficients, the B-spline model is much less seriously perturbed than the polynomial model.*

3.2.2. Estimation of Nonlinear HPA

In order to design a predistorter, we first need to estimate the HPA's nonlinearity, i.e., its amplitude response $A(\cdot)$ and phase response $\Upsilon(\cdot)$. We adopt two B-spline neural networks for this task, one for estimating $A(\cdot)$ and the other for $\Upsilon(\cdot)$. More specifically, we model the HPA's true amplitude response $A(r)$ and true phase response $\Upsilon(r)$ by the following two B-spline neural networks

$$\widehat{A}(r) = \sum\nolimits_{i=1}^{N} \mathcal{B}_i(r)\alpha_i, \tag{24}$$

$$\widehat{\Upsilon}(r) = \sum\nolimits_{i=1}^{N} \mathcal{B}_i(r)\beta_i. \tag{25}$$

Hence, the estimation task is turned into the problem of estimating the two B-spline neural networks' parameter vectors $\boldsymbol{\alpha} = \begin{bmatrix}\alpha_1 & \alpha_2 \cdots \alpha_N\end{bmatrix}^{\mathrm{T}} \in \mathbb{R}^N$ and $\boldsymbol{\beta} = \begin{bmatrix}\beta_1 & \beta_2 \cdots \beta_N\end{bmatrix}^{\mathrm{T}} \in \mathbb{R}^N$.

Given the K training data samples $\{x(k), w(k)\}_{k=1}^{K}$, where $K > N$, $x(k)$ and $w(k)$ are the input and output of the HPA, respectively, we can obtain the closed-form LS estimates of $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$. Specifically, first converting the complex-valued training dataset $\{x(k), w(k)\}_{k=1}^{K}$ into the two real-valued ones, namely, $\{r_x(k), r_w(k)\}_{k=1}^{K}$ and $\{r_x(k), \phi_w(k) - \phi_x(k)\}_{k=1}^{K}$. We introduce the respective

desired output vectors for the models (24) and (25) as

$$\boldsymbol{d}_A = \big[r_w(1)\ r_w(2)\cdots r_w(K)\big]^{\mathrm{T}}, \tag{26}$$

$$\boldsymbol{d}_\Upsilon = \big[(\phi_w(1)-\phi_x(1))\ (\phi_w(2)-\phi_x(2))\cdots(\phi_w(K)-\phi_x(K))\big]^{\mathrm{T}}, \tag{27}$$

as well as the regression matrix

$$\boldsymbol{\mathcal{B}} = \begin{bmatrix} \mathcal{B}_1(r_x(1)) & \mathcal{B}_2(r_x(1)) & \cdots & \mathcal{B}_N(r_x(1)) \\ \mathcal{B}_1(r_x(2)) & \mathcal{B}_2(r_x(2)) & \cdots & \mathcal{B}_N(r_x(2)) \\ \vdots & \vdots & \vdots & \vdots \\ \mathcal{B}_1(r_x(K)) & \mathcal{B}_2(r_x(K)) & \cdots & \mathcal{B}_N(r_x(K)) \end{bmatrix}. \tag{28}$$

Then the LS estimates of $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ are readily given respectively by

$$\widehat{\boldsymbol{\alpha}} = \big(\boldsymbol{\mathcal{B}}^{\mathrm{T}}\boldsymbol{\mathcal{B}}\big)^{-1}\boldsymbol{\mathcal{B}}^{\mathrm{T}}\boldsymbol{d}_A, \tag{29}$$

$$\widehat{\boldsymbol{\beta}} = \big(\boldsymbol{\mathcal{B}}^{\mathrm{T}}\boldsymbol{\mathcal{B}}\big)^{-1}\boldsymbol{\mathcal{B}}^{\mathrm{T}}\boldsymbol{d}_\Upsilon. \tag{30}$$

3.2.3. Designing Predistorter Using Gauss-Newton Algorithm

Given the estimated HPA's phase response $\widehat{\Upsilon}(r)$ of (25), the estimated predistorter's phase response is readily determined according to (15) as $\widehat{\Psi}(r_v) = -\widehat{\Upsilon}(r_v)$, which is also illustrated in Fig. 3. On the other hand, the problem of estimating the predistorter's amplitude response can also be turned into one of estimating the parameter vector $\boldsymbol{\theta} = \big[\theta_1\ \theta_2\cdots\theta_N\big]^{\mathrm{T}} \in \mathbb{R}^N$ for the following B-spline neural network

$$\widehat{B}(r) = \sum_{i=1}^{N} \mathcal{B}_i(r)\theta_i. \tag{31}$$

However, this is a nonlinear estimation problem, and an iterative gradient descent optimization procedure has to be applied. More specifically, given the set of the N input magnitude samples $\big\{r_x(k)\big\}_{k=1}^{K}$ and the estimated HPA's

amplitude response of (24), define the errors $e(k)$

$$\begin{aligned} e(k) =& r_x(k) - \widehat{A}\big(\widehat{B}(r_x(k))\big) = r_x(k) - \sum\nolimits_{i=1}^{N} \mathcal{B}_i\big(\widehat{B}(r_x(k))\big)\widehat{\alpha}_i \\ =& r_x(k) - \sum\nolimits_{i=1}^{N} \mathcal{B}_i\bigg(\sum\nolimits_{l=1}^{N} \mathcal{B}_l(r_x(k))\theta_l\bigg)\widehat{\alpha}_i, \end{aligned} \tag{32}$$

for $1 \le k \le K$. We minimize the following cost function to determine $\boldsymbol{\theta}$

$$J(\boldsymbol{\theta}) = \frac{1}{2}\sum\nolimits_{k=1}^{K} e^2(k). \tag{33}$$

Denote $r_v(k) = \widehat{B}(r_x(k)) = \sum_{l=1}^{N} \mathcal{B}_l(r_x(k))\theta_l$. Clearly, we must have $r_v(k) = \widehat{B}(r_x(k)) \ge 0$, and hence we actually compute $r_v(k)$ as $r_v(k) = \max\left\{\sum\limits_{l=1}^{N} \mathcal{B}_l(r_x(k))\theta_l, 0\right\}$. The gradient of the cost function (33) $\triangledown J(\boldsymbol{\theta}) = \left[\frac{\partial J}{\partial \theta_1}\ \frac{\partial J}{\partial \theta_2} \cdots \frac{\partial J}{\partial \theta_N}\right]^{\mathrm{T}}$ is given by

$$\begin{aligned} \frac{\partial J(\boldsymbol{\theta})}{\partial \theta_i} =& \sum_{k=1}^{K} e(k)\frac{\partial e(k)}{\partial \theta_i} = -\sum_{k=1}^{K} e(k)\frac{\mathrm{d}\widehat{A}(r_v(k))}{\mathrm{d}r_v}\frac{\partial r_v(k)}{\partial \theta_i}, \\ =& -\sum_{k=1}^{K} e(k)\bigg(\sum_{l=1}^{N} \frac{\mathrm{d}\mathcal{B}_l(r_v(k))}{\mathrm{d}r_v}\widehat{\alpha}_l\bigg)\mathcal{B}_i(r_x(k)),\ 1 \le i \le N, \end{aligned} \tag{34}$$

in which the derivatives of the B-spline basis functions can also be computed recursively according to the following De Boor recursion [57]

$$\begin{aligned} \frac{\mathrm{d}\mathcal{B}_l(r)}{\mathrm{d}r} = \frac{\mathrm{d}\mathcal{B}_l^{(P_d)}(r)}{\mathrm{d}r} =& \frac{P_d}{R_{P_d+l-1} - R_{l-1}}\mathcal{B}_l^{(P_d-1)}(r) \\ & - \frac{P_d}{R_{P_d+l} - R_l}\mathcal{B}_{l+1}^{(P_d-1)}(r),\ 1 \le l \le N. \end{aligned} \tag{35}$$

By denoting the iteration step index with the superscript $^{(\tau)}$ and given the initial estimate $\boldsymbol{\theta}^{(0)}$, the Gauss-Newton algorithm to estimate $\boldsymbol{\theta}$ is given by the iteration formula

$$\boldsymbol{\theta}^{(\tau)} = \boldsymbol{\theta}^{(\tau-1)} - \mu\left(\triangledown J\big(\boldsymbol{\theta}^{(\tau-1)}\big)\big(\triangledown J\big(\boldsymbol{\theta}^{(\tau-1)}\big)\big)^{\mathrm{T}}\right)^{-1}\triangledown J\big(\boldsymbol{\theta}^{(\tau-1)}\big), \tag{36}$$

where $\mu > 0$ is the step size.

3.2.4. Designing Predistorter Using PSO Algorithm

Since the cost function (33) is highly nonlinear, gradient-based estimators, such as the Gauss-Newton algorithm, require good initial parameter estimate to avoid local minima. Therefore, it is preferred to apply an efficient global optimization algorithm to solve this problem. Here we use the PSO [45, 46] to design the predistorter amplitude response. Recall that given the cost function $J(\boldsymbol{\theta})$ of (33) based on a block of training data $\{r_x(k)\}_{k=1}^{K}$, the predistorter design problem is to solve the following optimization problem

$$\boldsymbol{\theta}_{\text{opt}} = \arg\min_{\boldsymbol{\theta}\in\boldsymbol{\Theta}} J(\boldsymbol{\theta}), \tag{37}$$

where the search space is specified by

$$\boldsymbol{\Theta} = \prod_{i=1}^{N} \big[\theta_{i_{\min}},\ \theta_{i_{\max}}\big]. \tag{38}$$

When applying the PSO algorithm [45, 46] to solve this optimization, a swarm of particles $\{\boldsymbol{\theta}_m^{(l)}\}_{m=1}^{S}$ are 'flying' in the search space to find a solution, where S is the size of the swarm and $l \in \{1, 2, \cdots, L_{\max}\}$ denotes the lth movement of the swarm with $L_{\max}$ being the maximum number of searches. Each particle position $\boldsymbol{\theta}_m = \big[\theta_{m,1} \cdots \theta_{m,N}\big]^{\mathrm{T}}$ has a velocity $\boldsymbol{v}_m = \big[v_{m,1} \cdots v_{m,N}\big]^{\mathrm{T}}$ to direct its search, and $\boldsymbol{v}_m \in \boldsymbol{V}$ with the velocity space defined by

$$\boldsymbol{V} = \prod_{i=1}^{N} \big[-v_{i_{\max}},\ v_{i_{\max}}\big], \tag{39}$$

where $v_{i_{\max}} = \frac{1}{2}\big(\theta_{i_{\max}} - \theta_{i_{\min}}\big)$.

The PSO search is started by initializing the particles $\{\boldsymbol{\theta}_m^{(0)}\}_{m=1}^{S}$ randomly within $\boldsymbol{\Theta}$ and setting the velocity for each candidate particle to zero, namely, $\{\boldsymbol{v}_m^{(0)} = \mathbf{0}_N\}_{m=1}^{S}$. The so-called cognitive information $\boldsymbol{pb}_m^{(l)}$ and the social information $\boldsymbol{gb}^{(l)}$ record the best position visited by the particle m and the best position visited by the entire swarm, respectively, during the l movements. The cost function values associated with $\boldsymbol{pb}_m^{(l)}$ and $\boldsymbol{gb}^{(l)}$ are $J\big(\boldsymbol{pb}_m^{(l)}\big)$ and $J\big(\boldsymbol{gb}^{(l)}\big)$,

respectively. The velocities and positions of the swam are updated according to

$$\begin{aligned}\boldsymbol{v}_m^{(l)} =& I_w \cdot \boldsymbol{v}_m^{(l-1)} + rand(0,1) \cdot c_1 \cdot \left(\boldsymbol{pb}_m^{(l-1)} - \boldsymbol{\theta}_m^{(l-1)}\right) \\ &+ rand(0,1) \cdot c_2 \cdot \left(\boldsymbol{gb}^{(l-1)} - \boldsymbol{\theta}_m^{(l-1)}\right), \qquad (40)\\ \boldsymbol{\theta}_m^{(l)} =& \boldsymbol{\theta}_m^{(l-1)} + \boldsymbol{v}_m^{(l)}, \qquad (41)\end{aligned}$$

for $1 \leq m \leq S$, where I_w is the inertia weight, $rand(a, b)$ denotes the random number uniformly distributed in $[a,\ b]$, c_1 and c_2 are the two acceleration coefficients. Experimental results given in [46] show that a better performance can be achieved by using $I_w = rand(0, 1)$ rather than a constant inertia weight. The time varying acceleration coefficients [45] are adopted, in which

$$\begin{cases} c_1 = 2.5 - (2.5 - 0.5)\frac{l}{L_{\max}}, \\ c_2 = 0.5 + (2.5 - 0.5)\frac{l}{L_{\max}}. \end{cases} \qquad (42)$$

The velocity $\boldsymbol{v}_m^{(l)}$ and position $\boldsymbol{\theta}_m^{(l)}$, derived in (40) and (41), respectively, are projected inside the velocity space $\boldsymbol{V}$ and the search space $\boldsymbol{\Theta}$. Furthermore, if $\boldsymbol{v}_m^{(l)} = \mathbf{0}_N$, it is re-initialized to a non-zero value inside $\boldsymbol{V}$. Algorithm 1 summarizes this PSO algorithm.

4. Performance Evaluation

A simulation study is carried out to investigate the achievable performance of the proposed PSO assisted B-spline neural network based predistorter. In the simulated nonlinear MIMO downlink, the BS employs $L = 5$ transmit antennas to support $M = 3$ single-antenna MUs using the same single frequency-time resource block. The 64QAM signaling is adopted, and the transmit HPA's parameters are specified by (12). Both the Gauss-Newton algorithm and the PSO algorithm are compared in designing the B-spline predistorter to demonstrate the superior performance of the latter. The system's signal-to-noise ratio (SNR) is defined as $\text{SNR} = \frac{\sigma_s^2}{\sigma_n^2}$, where σ_s^2 is the average symbol energy of the 64QAM symbols. The ZF TPC matrix is calculated according to the channel estimate

Algorithm 1 Particle swarm optimization algorithm

Swarm Initialization

Give swarm size S and number of search iterations $L_{\max}$.

Randomly initialize $\left\{\boldsymbol{\theta}_m^{(0)}\right\}_{m=1}^{S}$ in $\boldsymbol{\Theta}$, and set $\left\{\boldsymbol{v}_m^{(0)} = \mathbf{0}_N\right\}_{m=1}^{S}$.

Compute $\left\{J\left(\boldsymbol{\theta}_m^{(0)}\right)\right\}_{m=1}^{S}$, set $\left\{\boldsymbol{pb}_m^{(0)} = \boldsymbol{\theta}_m^{(0)}\right\}_{m=1}^{S}$, and determine

$$\boldsymbol{gb}^{(0)} = \arg \min_{1 \le m \le S} J\left(\boldsymbol{\theta}_m^{(0)}\right).$$

Swarm Evolution

For $l = 1, 2, \cdots, L_{\max}$

  For $m = 1, 2, \cdots, S$

    Compute $\boldsymbol{v}_m^{(l)}$ according to (40).

    For $i = 1, 2, \cdots, N$

      If $v_{m,i}^{(l)} = 0$: $v_{m,i}^{(l)} = \pm 0.5 v_{i_{\max}} rand(0,1)$.

      If $v_{m,i}^{(l)} > v_{i_{\max}}$: $v_{m,i}^{(l)} = v_{i_{\max}}$.

      If $v_{m,i}^{(l)} < -v_{i_{\max}}$: $v_{m,i}^{(l)} = -v_{i_{\max}}$.

    End for

    Compute $\boldsymbol{\theta}_m^{(l)}$ according to (41).

    For $i = 1, 2, \cdots, N$

      If $\theta_{m,i}^{(l)} > \theta_{i_{\max}}$: $\theta_{m,i}^{(l)} = \theta_{i_{\max}}$.

      If $\theta_{m,i}^{(l)} < \theta_{i_{\min}}$: $\theta_{m,i}^{(l)} = \theta_{i_{\min}}$

    End For

    Compute $J\left(\boldsymbol{\theta}_m^{(l)}\right)$, and set $\boldsymbol{pb}_m^{(l)} = \boldsymbol{pb}_m^{(l-1)}$.

    If $J\left(\boldsymbol{pb}_m^{(l)}\right) > J\left(\boldsymbol{\theta}_m^{(l)}\right)$: $\boldsymbol{pb}_m^{(l)} = \boldsymbol{\theta}_m^{(l)}$.

    If $J\left(\boldsymbol{gb}^{(l-1)}\right) > J\left(\boldsymbol{pb}_m^{(l)}\right)$: $\boldsymbol{gb}^{(l-1)} = \boldsymbol{pb}_m^{(l)}$.

  End for

  $\boldsymbol{gb}^{(l)} = \boldsymbol{gb}^{(l-1)}$.

End for

Termination

Solution is $\boldsymbol{\theta}_{\text{opt}} = \boldsymbol{gb}^{(L_{\max})}$.

(7), which takes into account the channel estimation error. The BER performance is averaged over 100 MIMO channel realizations.

4.1. Predistorter Performance

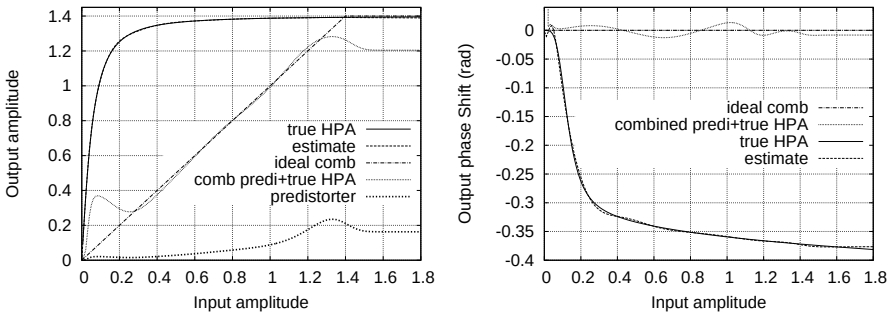
4.1.1. Estimating HPA

Consider the HPA specified by (12), whose amplitude response and phase response are depicted in Figure 2. To identify this HPA, 400 training input and output data samples are generated. The amplitudes of the inputs to the HPA are randomly drawn from [0.01, 1.35]. The B-spline model with polynomial degree $P_d = 4$ and $N = 10$ basis functions is adopted. The knot sequence is specified by

$$-2 \times 10^{-5}, -10^{-5}, -10^{-6}, \mathbf{10^{-5}}, 0.05, 0.1, 0.3, 0.5, 0.7, 1.1, 1.3, \mathbf{1.4}, 1.5, 1.6, 10^{3}.$$

Clearly, the input magnitude $r > 0$ and the HPA saturated at the input magnitude $r_{\text{sat}} = 1.4$. Therefore, we set the two boundary knots to $R_{\min} = 10^{-5}$ and $R_{\max} = 1.4$.

The B-spline estimated amplitude response and phase response model parameter vectors, $\widehat{\alpha}$ and $\widehat{\beta}$, are readily obtained by the closed-form LS solutions (29) and (30). The B-spline estimated HPA amplitude response and phase response are depicted in Figure 6, in comparison with the true HPA amplitude



(a) (b)

Figure 6: Comparison of the true HPA and its B-spline estimate as well as comparison of the ideal combined predistorter and true HPA and the combined estimated predistorter and true HPA: (a) amplitude response, and (b) phase response. The B-spline predistorter estimate is obtained using the Gauss-Newton algorithm.

response and phase response. It can be seen from Figure 6 (a) and Figure 6 (b) that the response of the B-spline neural network estimate closely match the response of the true HPA, except for near zero input amplitude where there is a very small but noticeable phase response error.

4.1.2. Estimating Predistorter by Gauss-Newton Algorithm

Based on the obtained B-spline estimate $\widehat{A}(r)$ of the HPA amplitude response, we can design the B-spline predistorter amplitude response model using the Gauss-Newton algorithm. The initial parameter vector $\boldsymbol{\theta}$ is set to $\mathbf{0}_N$ in this case. The estimated predistorter amplitude response is illustrated in Figure 6 (a). In Figure 6, we also display the amplitude response and phase response of the combined predistorter estimate and the true HPA, in comparison with the ideal response of the combined predistorter and the true HPA, which are specified by (14) and (15). Observe from Figure 6 (a) that for small input amplitude and very large input amplitude, the combined predistorter estimate and the true HPA deviates noticeably from the ideal one. Also there exists a large combined phase response error at very small input amplitude, as can be clearly seen from Figure 6 (b). Clearly, this estimated predistorter's amplitude response by the Gauss-Newton algorithm is insufficiently accurate. Better initial parameter vector are required for the Gauss-Newton algorithm to converge to an accurate solution.

4.1.3. Estimating Predistorter by PSO Algorithm

Next, we apply the PSO algorithm to design the B-spline predistorter amplitude response model based on the obtained B-spline estimate $\widehat{A}(r)$ of the HPA amplitude response. The population size is set to $S = 50$, and the maximum number of swam movements is $L_{\max} = 100$, while all the position lower bounds and upper bounds are set to $\theta_{i_{\min}} = -1.0$ and $\theta_{i_{\max}} = 1.0$, respectively. The amplitude response and phase response of the combined predistorter estimate and the true HPA are shown in Fig. 7 (a) and (b), respectively. Observe that the estimated combined amplitude and phase response closely match to the ideal

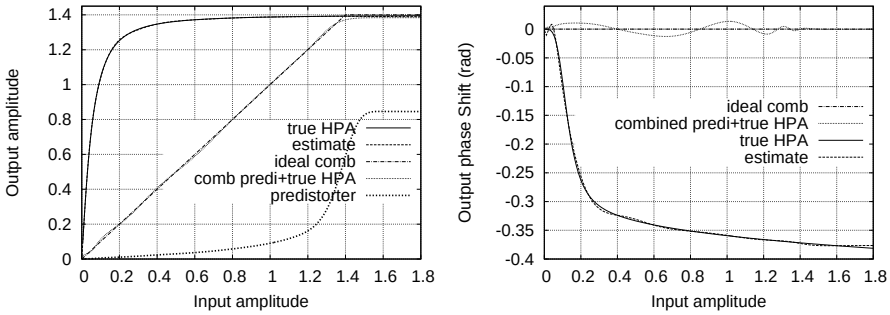


Figure 7: Comparison of the true HPA and its B-spline estimate as well as comparison of the ideal combined predistorter and true HPA and the combined estimated predistorter and true HPA: (a) amplitude response, and (b) phase response. The B-spline predistorter estimate is obtained using the particle swarm optimization algorithm.

ones. Clearly, the PSO algorithm is much better than the Gauss-Newton algorithm for solving this nonlinear optimization problem. More specifically, the estimated B-spline neural network based predistorter by the PSO algorithm is capable of accurately pre-compensating for the HPA's nonlinear distortion.

Remark 3. *The PSO algorithm is ideal for estimating the B-spline predistorter which has only $N = 10$ parameters. Two algorithmic parameters, the population size S and the maximum number of swam movements $L_{\max}$, need to be set. Extensive empirical results suggest that $L_{\max} = 100$ is sufficient and setting S to a few times of N is adequate. The PSO algorithm converges very fast. Although we set $L_{\max} = 100$, the algorithm actually converges in far less than 100 iterations. The predistorter design is an offline problem, since it does not depend on the channel. Specifically, it is designed before communication and fixed. The BS has sufficient computation capacity to implement the PSO algorithm to design the predistorter before performing the downlink TPC transmission.*

4.2. Bit Error Rate Performance

To further demonstrate the effectiveness of the proposed PSO assisted B-spline neural network based predistorter design, we apply the estimated B-spline predistorter obtained in Subsection 4.1.3 in the nonlinear MIMO downlink to

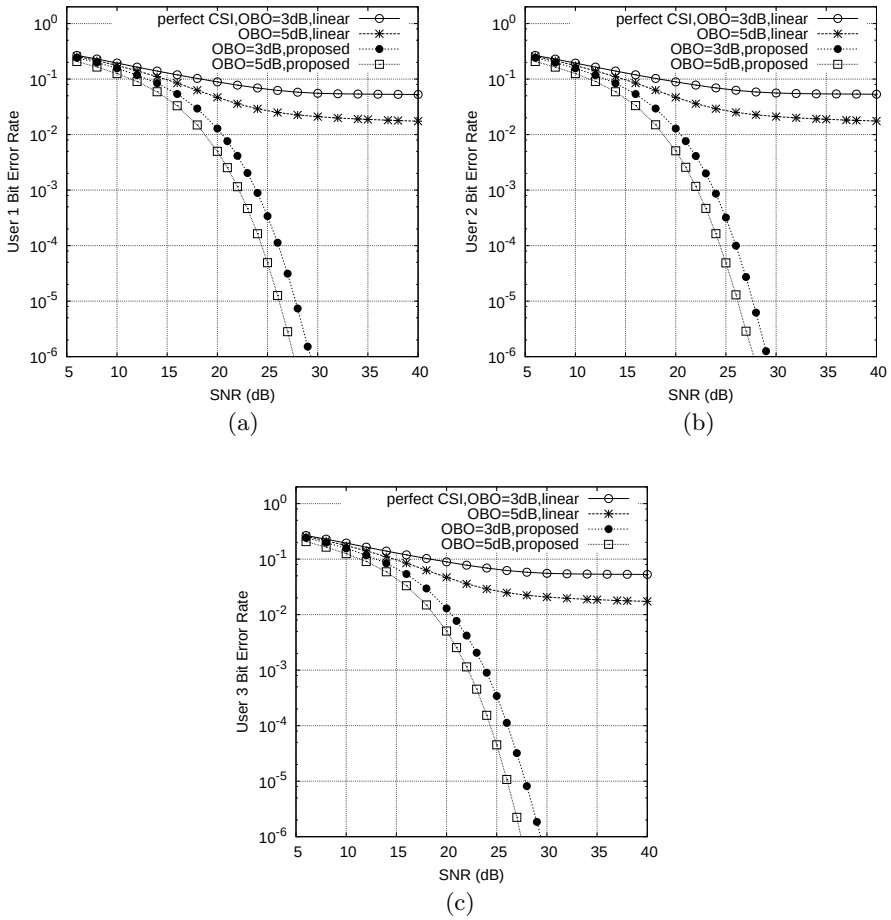


Figure 8: Comparison of the average bit error rate performance over 100 MIMO channel realizations obtained by the nonlinear transmit scheme utilizing the estimated predistorter and the classical linear ZF transmit precoding scheme, respectively, given the two OBO values of 3 dB and 5 dB and with perfect CSI: (a) MU $m = 1$, (b) MU $m = 2$, and (c) MU $m = 3$.

investigate the achievable BER performance. First, we consider the idealistic case of perfect CSI, namely, the channel estimation error bound $\sigma_\varepsilon = 0$. Figure 8 compares the achievable BER performance of our nonlinear transmit design with that of the classical linear transmit precoding scheme. To the best knowledge of the authors, there exists no other nonlinear transmit design in the literature for the NOMA multiuser nonlinear MIMO downlink considered in this research. Therefore, we only compare our proposed nonlinear transmit scheme with the linear transmit scheme. As expect, the linear ZF transmit

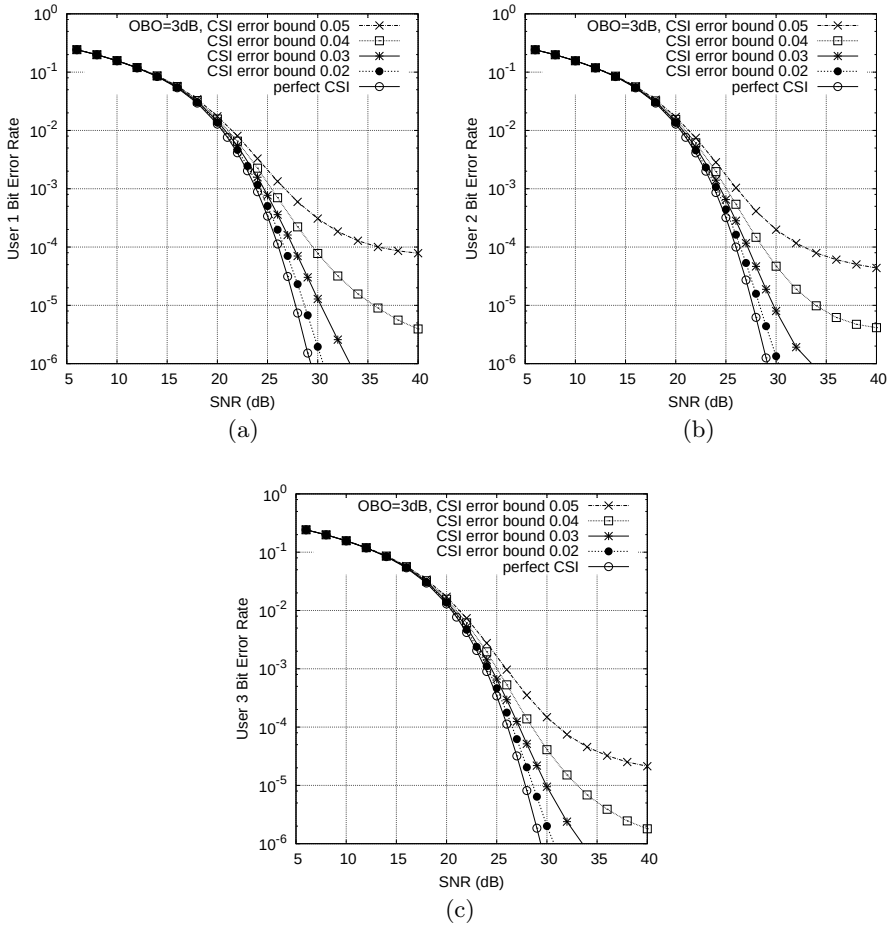


Figure 9: Impact of the channel estimation error bound σ_ε on the achievable average bit error rate performance of the nonlinear transmit scheme utilizing the estimated predistorter over 100 MIMO channel realizations, given OBO = 3 dB: (a) MU $m = 1$, (b) MU $m = 2$, and (c) MU $m = 3$.

precoding scheme cannot compensate for the HPAs' nonlinear distortions and, consequently, a very high BER floor occurs. By effectively compensating for the HPAs' nonlinear distortions with our PSO assisted B-spline neural network based predistorter, our nonlinear transmit design dramatically improving the achievable BER performance.

Next we investigate the impact of the channel estimation error on the achievable BER performance of our proposed nonlinear transmit scheme. Given OBO = 3 dB, Figure 9 depicts the BER performance under different levels

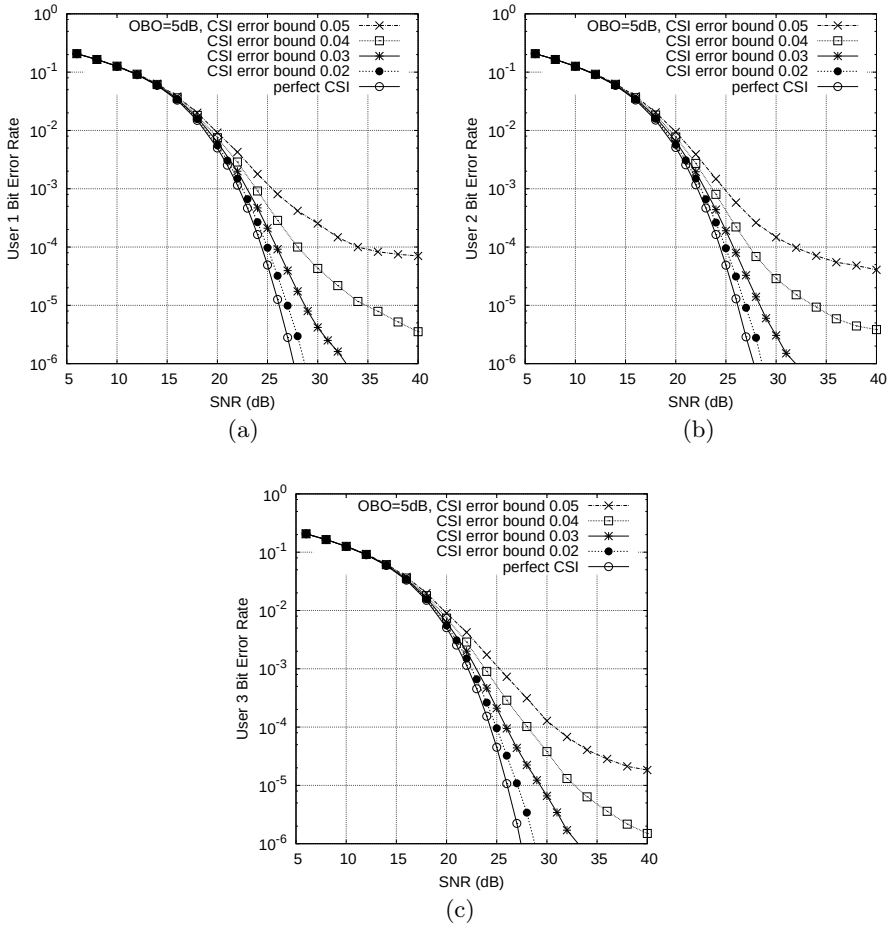


Figure 10: Impact of the channel estimation error bound σ_ε on the achievable average bit error rate performance of the nonlinear transmit scheme utilizing the estimated predistorter over 100 MIMO channel realizations, given OBO = 5 dB: (a) MU $m = 1$, (b) MU $m = 2$, and (c) MU $m = 3$.

of estimation error, ranging from low channel estimation error of $\sigma_\varepsilon = 0.02$ to high channel estimation error of $\sigma_\varepsilon = 0.05$. With very low channel estimation error of $\sigma_\varepsilon = 0.01$, the BER curve is almost indistinguishable form the idealistic case of perfect CSI. On the other hand, with very high channel estimation error of $\sigma_\varepsilon = 0.06$, the BER floor increases to around 10^{-3}. For graphic clarification, we do not plot these two BER curves. Similarly, given OBO = 5 dB, Figure 10 shows the impact of channel estimation error on the achievable BER performance. It can be seen that our proposed nonlinear transmit design rely-

ing on the estimated B-spline predistorter is reasonably robust to the channel estimation error.

5. Conclusions

In this paper, we have proposed an efficient and highly accurate predistorter design to enable a novel nonlinear transmit scheme for the NOMA multiuser nonlinear MIMO downlink with high-order QAM signaling and nonlinear transmit HPAs at the BS. Our main contribution has been to design a novel PSO assisted B-spline neural network based predistorter. This nonlinear predistorter can be pre-constructed in training before communication session, which provides an effective and accurate means of pre-compensating for the nonlinear distortions of transmit HPAs during communication session. With the nonlinear distortions of transmit HPAs been taking care of by this novel predistorter, the standard linear transmit precoding, such as the ZF precoding, can readily been employed to combat the multiuser MIMO channel interference. Our proposed nonlinear transmit design has been the first effective scheme for the NOMA multiuser nonlinear MIMO downlink. A simulation investigation has been conducted to demonstrate its effectiveness. The results obtained have also shown that our proposed nonlinear transmit design is reasonably robust to the channel estimation error.

Acknowledgment

This project was funded by the Deanship of Scientific Research (DSR) at King Abdulaziz University, Jeddah, under grant no.RG-3-135-40. The authors, therefore, acknowledge with thanks DSR technical and financial support.

References

[1] 5G Network Architecture: A High-Level Perspective. Huawei. https://www.huawei.com/minisite/hwmbbf16/insights/5G-Nework-Architecture-Whitepaper-en.pdf

[2] G. J. Foschini, "Layered space-time architecture for wireless communication in a fading environment when using multiple antennas," *Bell Labs Tech. J.*, vol. 1, no. 2, pp. 41-59, 1996.

[3] S. M. Alamouti, "A simple transmit diversity technique for wireless communications," *IEEE J. Sel. Areas Commun.*, vol. 16, no. 8, pp. 1451–1458., Oct. 1998.

[4] I. E. Telatar, "Capacity of multi-antenna Gaussian channels," *European Trans. Commun.*, vol. 10, no. 2, pp. 585–595, Nov./Dec. 1999.

[5] P. Vandenameele, L. van Der Perre, and M. Engels, *Space Division Multiple Access for Wireless Local Area Networks*. Boston, MA: Kluwer, 2001.

[6] A. Paulraj, R. Nabar, and D. Gore, *Introduction to Space-Time Wireless Communications*. Cambridge, U.K.: Cambridge Univ. Press, 2003.

[7] A. J. Paulraj, D. A. Gore, R. U. Nabar, and H. Bolcskei, "An overview of MIMO communications – a key to gigabit wireless," *Proc. IEEE*, vol. 92, no. 2, pp. 198–218, Feb. 2004.

[8] S. Sugiura, S. Chen, and L. Hanzo, "MIMO-aided near-capacity turbo transceivers: Taxonomy and performance versus complexity," *IEEE Commun. Surveys and Tutorials*, vol 14, no. 2, pp. 421–442, Secondquarter, 2012.

[9] F. Rusek, *et al.*, "Scaling up MIMO: Opportunities and challenges with very large arrays," *IEEE Signal Proces. Mag.*, vol. 30, no. 1, pp. 40–60, Jan. 2013.

[10] H. Q. Ngo, E. G. Larsson, and T. L. Marzetta, "Energy and spectral efficiency of very large multiuser MIMO systems," *IEEE Trans. Commun.*, vol. 61, no. 4, pp. 1436–1449, Apr. 2013.

[11] E. G. Larsson, O. Edfors, F. Tufvesson, and T. L. Marzetta, "Massive MIMO for next generation wireless systems," *IEEE Commun. Mag.*, vol. 52, no. 2, pp. 186–195, Feb. 2014.

[12] J. Yang and S. Roy, "On joint transmitter and receiver optimization for multiple-input-multiple-output (MIMO) transmission systems," *IEEE Trans. Commun.*, vol. 42, no. 12, pp. 3221–3231, Dec. 1994.

[13] H. Sampath, P. Stoica, and A. Paulraj, "Generalized linear precoder and decoder design for MIMO channels using the weighted MMSE criterion," *IEEE Trans. Commun.*, vol. 49, no. 12, pp. 2198–2206, Dec. 2001.

[14] H. Sampath and A. Paulraj, "Linear precoding for space-time coded systems with known fading correlations," *IEEE Commun. Lett.*, vol. 6, no. 6, pp. 239–241, Jun. 2002.

[15] Y. Jiang, J. Li, and W. W. Hager, "Joint transceiver design for MIMO communications using geometric mean decomposition," *IEEE Trans. Signal Process.*, vol. 53, no. 10, pp. 3791–3803, Oct. 2005.

[16] S. Chen, N. N. Ahmad, and L. Hanzo, "Adaptive minimum bit error rate beamforming," *IEEE Trans. Wireless Commun.*, vol.4, no. 2, pp. 341–348, Mar. 2005.

[17] S. Chen, A. Livingstone, and L. Hanzo, "Minimum bite-error rate design for space-time equalization-based multiuser detection," *IEEE Trans. Commun.*, vol. 54, no. 5, pp. 824–832, May 2006.

[18] S. Chen, A. Livingstone, H.-Q. Du, and L. Hanzo, "Adaptive minimum symbol error rate beamforming assisted detection for quadrature amplitude modulation," *IEEE Trans. Wireless Commun.*, vol. 7, no. 4, pp. 1140–1145, Apr. 2008.

[19] S. Chen, W. Yao, and L. Hanzo, "Semi-blind adaptive spatial equalisation for MIMO systems with high-order QAM signalling," *IEEE Trans. Wireless Commun.*, vol. 7, no. 11, pp. 4486–4491, Nov. 2008.

[20] S. Chen and L. Hanzo, "Fast converging semi-blind space-time equalisation for dispersive QAM MIMO systems," *IEEE Trans. Wireless Commun.*, vol. 8, no. 8, pp. 3969–3974, Aug. 2009.

[21] W. Yao, S. Chen, S. Tan, and L. Hanzo, "Minimum bit error rate multiuser transmission designs using particle swarm optimisation," *IEEE Trans. Wireless Commun.*, vol. 8, no. 10, pp. 5012–5017, Oct. 2009.

[22] S. Sugiura, S. Chen, and L. Hanzo, "Generalized space-time shift keying designed for flexible diversity-, multiplexing- and complexity-tradeoffs," *IEEE Trans. Wireless Commun.*, vol. 10, no. 4, pp. 1144–1153, Apr. 2011.

[23] P. Zhang, S. Chen, and L. Hanzo, "Two-tier channel estimation aided near-capcity MIMO transceivers relaying on norm-based joint transmit and receive antenna selection," *IEEE Trans. Wireless Commun.*, vol. 14, no. 1, pp. 122–137, Jan. 2015.

[24] S. Chen, L. Hanzo, and A. Livingstone, "MBER Space-time decision feedback equalization assisted multiuser detection for multiple antenna aided SDMA systems," *IEEE Trans. Signal Process.*, vol. 54, no. 8, pp. 3090–3098, Aug. 2006.

[25] D. P. Palomar and Y. Jiang, "MIMO transceiver designs via majorization theory," *Foundations and Trends in Commun. and Infor. Theory*, vol. 3, no. 4-5, pp 331–551, Jun. 2007.

[26] W. Yao, S. Chen, and L. Hanzo, "A transceiver design based on uniform channel decomposition and MBER vector perturbation," *IEEE Trans. Veh. Techno.*, vol. 59, no. 6, pp. 3153–3159, Jul. 2010.

[27] W. Yao, S. Chen, and L. Hanzo, "Generalised MBER-based vector precoding design for multiuser transmission," *IEEE Trans. Veh. Techno.*, vol. 60, no. 2, pp. 739–745, Feb. 2011.

[28] S. Gong, *et al.*, "Robust energy efficiency optimization for amplify-and-forward MIMO relaying systems," *IEEE Trans. Wireless Commun.*, vol. 18, no. 9, pp. 4326–4343, Sep. 2019.

[29] A. A. M. Saleh, "Frequency-independent and frequency-dependent nonlinear models of TWT amplifiers," *IEEE Trans. Commun.*, vol. COM-29, no. 11, pp. 17151720, Nov. 1981.

[30] M. Honkanen and S.-G. Häggman, "New aspects on nonlinear power amplifier modeling in radio communication system simulations," in *Proc. PIMRC'97* (Helsinki, Finland), Sep. 1-4, 1997, pp. 844–848.

[31] C. J. Clark, *et al.*, "Time-domain envelope measurement technique with application to wideband power amplifier modeling," *IEEE Trans. Microw. Theory and Tech.*, vol. 46, no. 12, pp. 2531–2540, Dec. 1998.

[32] C.-S. Choi, *et al.*, "RF impairment models 60 GHz band SYS/PHY simulation," Document IEEE 802.15-06-0477-01-003c, Nov. 2006. `https://mentor.ieee.org/802.15/dcn/06/15-06-0477-01-003c-rf-impairment-models-60ghz-band-sysphy-simulation.pdf`

[33] V. Erceg, *et al.*, "60 GHz impairments modeling," Document IEEE 802.11-09/1213r1, Nov. 2009.

[34] L. Hanzo, S. X. Ng, T. Keller, and W. Webb, *Quadrature Amplitude Modulation: From Basics to Adaptive Trellis-Coded, Turbo-Equalised and Space-Time Coded OFDM, CDMA and MC-CDMA Systems*. Chichester, UK: John Wiley, 2004.

[35] S. Chen, S. X. Ng, E. Khalaf, A. Morfeq ,and N. Alotaibi, "Multiuser detection for nonlinear MIMO uplink," *IEEE Trans. Commun.*, vol. 69, no. 1, pp. 207–219, Jan. 2020.

[36] L. Ding, *et al.*, "A robust digital baseband predistorter constructed using memory polynomials," *IEEE Trans. Commun.*, vol. 52, no. 1, pp. 159–165, Jan. 2004.

[37] D. Zhou and V. E. DeBrunner, "Novel adaptive nonlinear predistorters based on the direct learning algorithm," *IEEE Trans. Signal Process.*, vol. 55, no. 1, pp. 120–133, Jan. 2007.

[38] M.-C. Chiu, C.-H. Zeng, and M.-C. Liu, "Predistorter based on frequency domain estimation for compensation of nonlinear distortion in OFDM systems," *IEEE Trans. Veh. Techno.*, vol. 57, no. 2, pp. 882–892, Mar. 2008.

[39] S. Choi, E.-R. Jeong, and Y. H. Lee, "Adaptive predistortion with direct learning based on piecewise linear approximation of amplifier nonlinearity," *IEEE J. Sel. Topics Signal Process.*, vol. 3, no. 3, pp. 397–404, Jun. 2009.

[40] V. P. G. Jiménez, Y. Jabrane, A. G. Armada, and B. Ait Es Said, "High power amplifier pre-distorter based on neural-fuzzy systems for OFDM signals," *IEEE Trans. Broadcasting*, vol.57, no. 1, pp. 149–158, Mar. 2011.

[41] S. Chen, "An efficient predistorter design for compensating nonlinear memory high power amplifier," *IEEE Trans. Broadcasting*, vol. 57, no. 4, pp. 856–865, Dec. 2011.

[42] S. Chen, X. Hong, Y. Gong, and C. J. Harris, "Digital predistorter design using B-spline neural network and inverse of De Boor algorithm," *IEEE Trans. Circuits and Systems I*, vol. 60, no. 6, pp. 1584–1594, Jun. 2013.

[43] S. Chen, X. Hong, E. Khalaf, A. Morfeq, and N. Alotaibi, "Adaptive B-spline neural network based nonlinear equalization for high-order QAM systems with nonlinear transmit high power amplifier," *Digital Signal Processing*, vol. 40, pp. 238–249, 2015.

[44] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proc. 1995 IEEE Int. Conf. Neural Networks* (Perth, Australia), Nov. 27-Dec.1, 1995, pp. 6390–6394.

[45] A. Ratnaweera and S. K. Halgamuge, "Self-organizing hierarchical particle swarm optimizer with time-varying acceleration coefficients," *IEEE Trans. Evolutionary Computation*, vol. 8, no. 3, pp. 240–255, Jun. 2004.

[46] S. Chen, X. Hong, B. L. Luk, and C. J. Harris, "Non-linear system identification using particle swarm optimisation tuned radial basis function models," *Int. J. Bio-Inspired Computation*, vol. 1, no. 4, pp. 246-258, 2009.

[47] S. Chen and B. L. Luk, "Digital IIR filter design using particle swarm optimisation," *Int. J. Modelling, Identification and Control*, vol. 9 no.4, pp. 327–335, 2010.

[48] S. Chen, X. Hong, and C. J. Harris, "Particle swarm optimization aided orthogonal forward regression for unified data modelling," *IEEE Trans. Evolutionary Computation*, vol. 14, no. 4, pp. 477–499, Aug. 2010.

[49] X. Hong, J. B. Gao, S. Chen, and C. J. Harris, "Particle swarm optimisation assisted classification using elastic net prefiltering," *Neurocomputing*, vol. 122, pp. 210–220, 2013.

[50] N. Zeng, Z. Wang, H. Zhang, and F. E. Alsaadi, "A novel switching delayed PSO algorithm for estimating unknown parameters of lateral flow immunoassay," *Cognitive Computation*, vol. 8, pp. 143–152, 2016.

[51] N. Zeng, *et al.*, "A dynamic neighborhood-based switching particle swarm optimization algorithm," *IEEE Trans. Cybernetics*, Early Access, pp. 1–12, DOI:10.1109/TCYB.2020.3029748

[52] J. M. Pena, "B-spline and optimal stability," *Math. Comput.*, vol. 66, no. 220, pp. 1555–1560, Oct. 1997.

[53] T. Lyche and J. M. Pena, "Optimally stable multivariate bases" *Advan. Comput. Math.*, vol. 20, no. 1, pp. 149–159, Jan. 2004.

[54] E. Mainar and J. M. Pena, "Optimal stability of bivariate tensor product B-bases " *J. Numer. Anal. Ind. Appl. Math.*, vol. 6, nos. 34, pp. 95–104, Oct. 2012

[55] X. Hong, S. Iplikci, S. Chen, and K. Warwick, "A model-based PID controller for Hammerstein systems using B-spline neural networks," *Int. J. Adaptive Control and Signal Processing*, vol. 28, nos 3-5, pp. 412–428, 2014.

[56] S. Chen, X. Hong, J. B. Gao, and C. J. Harris, "Complex-valued B-spline neural networks for modeling and inverting Hammerstein systems," *IEEE Trans. Neural Networks and Learning Systems*, vol. 25, no. 9, pp. 1673–1685, Sep. 2014.

[57] C. De Boor, *A Practical Guide to Splines*. New York: Spring Verlag, 1978.