Digital Predistorter Design Using B-Spline Neural Network and Inverse of De Boor Algorithm

Sheng Chen, *Fellow, IEEE*, Xia Hong, *Senior Member, IEEE*, Yu Gong, and Chris J. Harris

Abstract—This contribution introduces a new digital pre-distorter to compensate serious distortions caused by memory high power amplifiers (HPAs) which exhibit output saturation characteristics. The proposed design is based on direct learning using a data-driven B-spline Wiener system modeling approach. The nonlinear HPA with memory is first identified based on the B-spline neural network model using the Gauss-Newton algorithm, which incorporates the efficient De Boor algorithm with both B-spline curve and first derivative recursions. The estimated Wiener HPA model is then used to design the Hammerstein predistorter. In particular, the inverse of the amplitude distortion of the HPA's static nonlinearity can be calculated effectively using the Newton-Raphson formula based on the inverse of De Boor algorithm. A major advantage of this approach is that both the Wiener HPA identification and the Hammerstein predistorter inverse can be achieved very efficiently and accurately. Simulation results obtained are presented to demonstrate the effectiveness of this novel digital predistorter design.

Index Terms—B-spline neural network, De Boor algorithm, Hammerstein model, memory high power amplifier, output saturation, predistorter, Wiener model.

I. INTRODUCTION

HIGH POWER amplifier (HPA) as an indispensable component can be found in any wireless communication system. The operation of HPAs in modern wireless systems may introduce serious nonlinear distortions, causing adjacent channel interference and degrading the system's achievable bit error rate (BER) performance. The problem becomes particularly acute, as the recent green-radio initiative [1] places the emphasis on the energy-efficiency aspect of communication. To achieve high energy efficiency, HPAs should operate at their output saturation regions but this operational mode could not accommodate high bandwidth-efficiency single-carrier high-order quadrature amplitude modulation (QAM) signals

S. Chen is with Electronics and Computer Science, University of Southampton, Southampton SO17 1BJ, U.K., and also with King Abdulaziz University, Jeddah 21589, Saudi Arabia (e-mail: sqc@ecs.soton.ac.uk).

X. Hong is with the School of Systems Engineering, University of Reading, Reading RG6 6AY, U.K. (e-mail: x.hong@reading.ac.uk).

Y. Gong is with the School of Electronic, Electrical and Systems Engineering, Loughborough University, Loughborough LE11 3TU, U.K. (e-mail: y.gong@lboro.ac.uk).

C. J. Harris is with Electronics and Computer Science, University of Southampton, Southampton SO17 1BJ, U.K. (e-mail: cjh@ecs.soton.ac.uk).

[2] as well as multi-carrier orthogonal frequency division multiplexing (OFDM) signals [3], which are essential modern transmission technologies. It is therefore critical to compensate the nonlinearity of the HPA in the design of a wireless system. Early researches often considered HPAs to be memoryless. However, for high-rate broadband signals, the influence of the HPAs' memory effects can no longer be ignored. The memory effects are caused by the electrical and electrothermal factors explained in [4]. An accurate linearized compensation technique therefore needs to consider not only the nonlinearities caused by the current input signals but also the distortion induced by the memory effects. Digital predistorter (PD) is considered to be a most effective linearization technique, because it offers a modest implementation cost, while achieves a relatively good performance.

Existing predistortion techniques for compensating memory HPAs [5]–[17] can roughly be divided into three categories. The look-up table (LUT) based techniques [5]–[7] realize a PD by representing the inverse characteristic function of the memory HPA in a LUT. The so-called indirect-learning based PD designs [11]–[13] first identify a post-inverse polynomial filter for the memory HPA to be compensated and then copy the post-inverse polynomial filter to form the PD. By contrast, the direct-learning based PD designs [14]–[16] first identify the input-output relation of the memory HPA using a polynomial model and then adapt a polynomial PD directly to invert the resulting polynomial HPA model. A recent work [17] uses a neural-fuzzy based PD, instead of a polynomial based PD, in the indirect-learning structure. It is well understood that the memory HPA can be modeled by the Wiener model consisting of a linear filter followed by a memoryless nonlinearity [18]. Physically, the memoryless nonlinearity of the HPA is represented by the output amplitude and phase response functions that are the nonlinear functions of the input signal amplitude. The two types of static nonlinearity widely adopted to model the HPA are the traveling-wave tube (TWT) nonlinearity [18], [19] and the solid state power amplifier's nonlinearity [20]. Most of the existing researches dealing with the TWT non-linearity, including [11]–[17], adopt a two-parameters output amplitude response model [19], which peaks at an input saturation amplitude. However, when the input amplitude increases beyond this saturation point, the output amplitude of this model actually starts to fall. This is in contrast to the physical intuition that the output amplitude should not fall off beyond saturation as is supported by the real measurements of HPAs [18].

Against this background, a novel PD design is proposed based on a direct learning structure in this paper. Our contribution is threefold. Firstly, we adopt a more realistic memory HPA

model for the TWT nonlinearity in our design which exhibits true output saturation characteristics. Secondly, we present a data-driven approach to identify the nonlinear memory HPA based on the B-spline Wiener model using the Gauss-Newton algorithm, which naturally exploits the efficiency of both B-spline curve and first derivative recursions in the De Boor algorithm [21]. The Wiener model comprising a linear dynamic model followed by a nonlinear static functional transformation is widely adopted in practice [22]–[31]. The model characterization and representation of the unknown nonlinear static function in the Wiener model is fundamental to its applications, and the B-spline basis function for nonlinear modeling [32]–[34] offers an efficient and accurate means of modeling the nonlinearity of the complex-valued Wiener system [35], in comparison with other methods. Furthermore, the inverse of the B-spline Wiener model can be obtained very efficiently and accurately, and this naturally comes to our third contribution. We design the Hammerstein predistorter based on the estimated B-spline Wiener HPA model. In particular, the inverse of the amplitude distortion of the HPA's static nonlinearity is calculated effectively using the Newton-Raphson formula based on the inverse of De Boor algorithm. Simulation results are used to demonstrate the effectiveness of this novel digital predistorter design.

The rest of this contribution is organized as follows. The Wiener model for memory HPAs is first introduced in Section II, and the identification algorithm is then presented for using the B-spline neural network to model the Wiener HPA. In Section III, the proposed Hammerstein PD solution is derived based on the estimated B-spline Wiener HPA model. Simulation results are presented in Section IV to demonstrate the effectiveness of the proposed PD design approach, while our conclusions are offered in Section V.

II. IDENTIFICATION OF MEMORY HPA MODEL

A widely used model for memory HPAs is the Wiener model [18], which comprises a linear system followed by a static non-linearity. Throughout the rest of this contribution, a complex-valued number $x \in \mathbb{C}$ is represented by the rectangular form $x = x_R + \mathrm{j}x_I$, where $\mathrm{j} = \sqrt{-1}$, while $x_R = \Re[x]$ and $x_I = \Im[x]$ denote the real and imaginary parts of x, or alternatively by the polar form $x = |x| \cdot \exp(\mathrm{j}\angle^x)$ with $|x|$ denoting the amplitude of x and $\angle^x$ its phase.

A. The Wiener Model for Memory HPAs

The linear filter of order L representing the HPA's memory effect on the input signal is defined by its z transfer function

$$H(z) = \sum_{i=0}^{L} h_i z^{-i}, \quad h_0 = 1, \tag{1}$$

with the complex-valued coefficient vector given by $\mathbf{h} = [h_1 \ h_2 \cdots \ h_L]^{\mathrm{T}} \in \mathbb{C}^L$. Without loss of generality, we assume $h_0 = 1$. If this is not the case, h_0 can always be absorbed into the complex-valued static nonlinearity of the HPA, and the filter's coefficients are re-scaled as h_i/h_0 for $0 \le i \le L$. The input signal to the memory HPA, $x(k)$, where the discrete time index k is related to the symbol rate sampling,

takes the values from the complex-valued M-QAM symbol set [2]

$$\left\{ d(2l - \sqrt{M} - 1) + \mathrm{j}d(2q - \sqrt{M} - 1), 1 \le l, q \le \sqrt{M} \right\}, \tag{2}$$

where $2d$ is the minimum distance between symbol points. Although we consider single-carrier QAM systems, the approach is equally applicable to multi-carrier OFDM systems [3]. The unobservable linear filter output

$$w(k) = \sum_{i=0}^{L} h_i x(k - i), \tag{3}$$

which is the input to the static nonlinearity part of the HPA model, can be expressed as

$$w(k) = r(k) \cdot \exp\left(\mathrm{j}\psi(k)\right), \tag{4}$$

with the amplitude $r(k) = |w(k)|$ and phase $\psi(k) = \angle^{w(k)}$.

The HPA's static nonlinearity is assumed to be the TWT nonlinearity [18], [19], but the approach is equally applicable to the solid state power amplifier [20]. The input signal is affected by the nonlinear amplitude and phase functions of the HPA, and the output signal $y(k)$ is distorted mainly depending on the input signal amplitude $r(k)$, yielding

$$\begin{aligned} y(k) &= |y(k)| \cdot \exp\left(\mathrm{j}\angle^{y(k)}\right) \\ &= A\left(r(k)\right) \cdot \exp\left(\mathrm{j}\left(\psi(k) + \Phi\left(r(k)\right)\right)\right). \end{aligned} \tag{5}$$

The output amplitude $A(r(k))$ and the phase $\Phi(r(k)) = \angle^{y(k)} - \psi(k)$ of the HPA are specified respectively by

$$A(r) = \begin{cases} \alpha_a r/(1 + \beta_a r^2), & 0 \le r \le r_{\mathrm{sat}}, \\ A_{\max}, & r > r_{\mathrm{sat}}, \end{cases} \tag{6}$$

$$\Phi(r) = \alpha_\phi r^2 / \left(1 + \beta_\phi r^2\right), \tag{7}$$

with the positive real-valued parameter vector that specifies the TWT nonlinearity given by $\mathbf{t} = [\alpha_a \ \beta_a \ \alpha_\phi \ \beta_\phi]^{\mathrm{T}}$, where the saturating input amplitude is defined as

$$r_{\mathrm{sat}} = \frac{1}{\sqrt{\beta_a}}, \tag{8}$$

while the saturation output amplitude is given by

$$A_{\max} = \alpha_a / (2\sqrt{\beta_a}). \tag{9}$$

The underlying physics require that $A_{\max} > r_{\mathrm{sat}}$ and the input amplitude r meets the condition $R_{\min} < r < R_{\max}$, where $R_{\min} = 0$ and $R_{\max}$ is some large positive number.

Note that the work [19] assumes an output amplitude $A(r) = \alpha_a r/(1 + \beta_a r^2)$, which peaks at $r = r_{\mathrm{sat}}$ but falls off from the peak value $A_{\max}$ when $r > r_{\mathrm{sat}}$. This is against the physical intuition that the output amplitude should not fall off beyond saturation. Our output amplitude model (6) is more realistic and is supported by the real measurements of HPAs [18]. The input back-off (IBO) of the HPA is defined as

$$\mathrm{IBO} = 10 \cdot \log_{10}(P_{\mathrm{sat}}/P_{\mathrm{avg}}), \tag{10}$$

where $P_{\mathrm{sat}} = r_{\mathrm{sat}}^2$ is the saturation input power and P_{avg} is the average power of the signal at the input of the TWT nonlinearity.
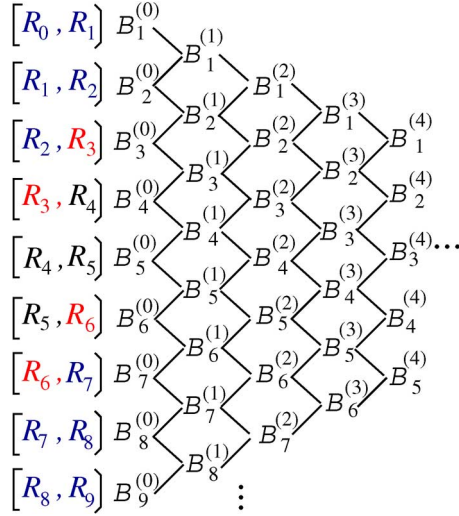
Fig. 1.   Visualization of the De Boor recursion for $P_o = 4$ and $N_b = 5$.

Note that P_{avg} is defined as the average power of $w(k)$, which is equal to the average power of $x(k)$ scaled by the linear filter power gain $1 + \|\mathbf{h}\|^2$. A small IBO value indicates that the HPA operates in the highly nonlinear saturation region.

B. B-Spline Modeling of the HPA's Nonlinearity

Univariate real-valued B-spline basis functions are parametrized by the order $(P_o - 1)$ of a piecewise polynomial and a knot vector which is a set of values defined on the real line that break it up into a number of intervals. Suppose that there are N_b basis functions. Then the knot vector is specified by the $(N_b + P_o + 1)$ knot values, $\{R_0, R_1, \cdots, R_{N_b + P_o}\}$, with

$$R_0 < R_1 \cdots < R_{P_o - 2} < R_{P_o - 1} = R_{\min} < R_{P_o} < \cdots$$

$$< R_{N_b} < R_{N_b + 1} = R_{\max} < R_{N_b + 2} < \cdots < R_{N_b + P_o}.$$
$$(11)$$

At each end, there are $P_o - 1$ external knots that are outside the input region and one boundary knot. As a result, the number of internal knots is $N_b + 1 - P_o$. Given the set of predetermined knots (11), the set of N_b B-spline basis functions can be formed by using the De Boor recursion [21], yielding

$$B_l^{(0)}(r) = \begin{cases} 1, & \text{if } R_{l-1} \le r < R_l, \\ 0, & \text{otherwise,} \end{cases} \quad 1 \le l \le N_b + P_o, \quad (12)$$

$$B_l^{(p)}(r) = \frac{r - R_{l-1}}{R_{p+l-1} - R_{l-1}} B_l^{(p-1)}(r) + \frac{R_{p+l} - r}{R_{p+l} - R_l} B_{l+1}^{(p-1)}(r),$$
$$\text{for } l = 1, \cdots, N_b + P_o - p \text{ and } p = 1, \cdots P_o. \quad (13)$$

The derivative of the B-spline basis function $B_l^{(P_o)}(r)$ can also be computed recursively according to

$$\frac{d\, B_l^{(P_o)}(r)}{d\, r} = \frac{P_o}{R_{P_o+l-1} - R_{l-1}} B_l^{(P_o - 1)}(r)$$
$$- \frac{P_o}{R_{P_o+l} - R_l} B_{l+1}^{(P_o - 1)}(r), \ 1 \le l \le N_b. \quad (14)$$

The De Boor recursion is visualized in Fig. 1.

Given the input amplitude to the static nonlinearity of the HPA, $r(k)$, we model the amplitude and phase of the HPA's static nonlinearity by the following two real-valued univariate B-spline neural networks

$$\hat{A}(r(k)) = \sum_{l=1}^{N_b} B_l^{(P_o)}(r(k)) \omega_l, \quad (15)$$

$$\hat{\Phi}(r(k)) = \sum_{l=1}^{N_b} B_l^{(P_o)}(r(k)) \theta_l, \quad (16)$$

where $\boldsymbol{\omega} = [\omega_1 \ \omega_2 \cdots \omega_{N_b}]^{\mathrm{T}} \in \mathbb{R}^{N_b}$ and $\boldsymbol{\theta} = [\theta_1 \ \theta_2 \ \cdots \ \theta_{N_b}]^{\mathrm{T}} \in \mathbb{R}^{N_b}$ are the parameters to be determined. The derivatives of the two B-spline models are given by

$$\hat{A}'(r(k)) = \sum_{l=1}^{N_b} \frac{d\, B_l^{(P_o)}(r(k))}{d\, r(k)} \omega_l, \quad (17)$$

$$\hat{\Phi}'(r(k)) = \sum_{l=1}^{N_b} \frac{d\, B_l^{(P_o)}(r(k))}{d\, r(k)} \theta_l. \quad (18)$$

Note that, due to the piecewise nature of B-spline functions, there are only $P_o + 1$ basis functions with nonzero functional/derivative values at any point r. Hence, the complexity of the De Boor algorithm is determined by the polynomial order P_o, rather than the number of knots, and this is in the order of $\mathcal{O}(P_o^2)$.

C. Identification Algorithm

Given a block of training data $\{\mathbf{x}(k), y(k)\}_{k=1}^K$, where $\mathbf{x}(k) = [x(k) \ x(k-1) \ \cdots \ x(k-L)]^{\mathrm{T}}$ and $x(k)$ is the input to the HPA, the task is to estimate the parameter vector $\boldsymbol{\vartheta} = [\vartheta_1 \ \vartheta_2 \ \cdots \ \vartheta_{2(N_b+L)}]^{\mathrm{T}}$ of the Wiener model, defined as

$$\boldsymbol{\vartheta} = \left[\boldsymbol{\omega}^{\mathrm{T}} \ \boldsymbol{\theta}^{\mathrm{T}} \ \hat{\mathbf{h}}_R^{\mathrm{T}} \ \hat{\mathbf{h}}_I^{\mathrm{T}}\right]^{\mathrm{T}} \in \mathbb{R}^{2(N_b+L)}, \quad (19)$$

where $\hat{\mathbf{h}} = \hat{\mathbf{h}}_R + \mathrm{j}\hat{\mathbf{h}}_I$ denotes an estimate of $\mathbf{h} = \mathbf{h}_R + \mathrm{j}\mathbf{h}_I$ with $\mathbf{h}_R = [h_{R_1} \ h_{R_2} \ \cdots \ h_{R_L}]^{\mathrm{T}}$ and $\mathbf{h}_I = [h_{I_1} \ h_{I_2} \ \cdots \ h_{I_L}]^{\mathrm{T}}$. Note that the output of the complex-valued B-spline neural network is given by

$$\hat{y}(k) = \hat{A}(\hat{r}(k)) \cdot \exp\left(\mathrm{j}\left(\hat{\Phi}(\hat{r}(k)) + \hat{\psi}(k)\right)\right), \quad (20)$$

with $\hat{w}(k) = \hat{r}(k) \cdot \exp(\mathrm{j}\hat{\psi}(k)) = x(k) + \sum_{i=1}^L \hat{h}_i x(k-i)$. The measured memory HPA's output may be corrupted by a small noise and, therefore, it takes the form

$$y(k) = f_{\mathrm{HPA}}(\mathbf{x}(k); \mathbf{h}, \mathbf{t}) + \xi(k), \quad (21)$$

where the complex-valued *unknown* nonlinear mapping $f_{\mathrm{HPA}}(\bullet; \mathbf{h}, \mathbf{t})$ is specified by (3) to (7), while $\xi(k)$ is the complex-valued Gaussian white noise with $E[|\xi(k)|^2] = 2\sigma_\xi^2$. Define the error between the desired output $y(k)$ and the model output $\hat{y}(k)$ as $e(k) = y(k) - \hat{y}(k)$, yielding the sum of squared errors (SSE) cost function

$$J_{\mathrm{SSE}}(\boldsymbol{\vartheta}) = \sum_{k=1}^K |e(k)|^2 = \sum_{k=1}^K \left(e_R^2(k) + e_I^2(k)\right). \quad (22)$$

We apply the Gauss-Newton algorithm to estimate $\boldsymbol{\vartheta}$. First denote $\boldsymbol{\varepsilon} = [\varepsilon_1 \ \varepsilon_2 \ \cdots \ \varepsilon_{2K}]^{\mathrm{T}} \in \mathbb{R}^{2K}$ as

$$\boldsymbol{\varepsilon} = [e_R(1) \, e_R(2) \cdots e_R(K) \, e_I(1) \, e_I(2) \cdots e_I(K)]^{\mathrm{T}}. \quad (23)$$

By denoting the iteration step with the superscript $^{(\tau)}$ and with an initial estimate $\boldsymbol{\vartheta}^{(0)}$, the iteration formula is given by

$$\boldsymbol{\vartheta}^{(\tau)} = \boldsymbol{\vartheta}^{(\tau-1)} - \mu \left(\left(\mathbf{J}^{(\tau)} \right)^{\mathrm{T}} \mathbf{J}^{(\tau)} \right)^{-1} \left(\mathbf{J}^{(\tau)} \right)^{\mathrm{T}} \boldsymbol{\varepsilon} \left(\boldsymbol{\vartheta}^{(\tau-1)} \right), \quad (24)$$

where $\mu > 0$ is the step size, and $\mathbf{J}^{(\tau)}$ denotes the Jacobian of $\boldsymbol{\varepsilon}(\boldsymbol{\vartheta}^{(\tau-1)})$ which is define by

$$\mathbf{J} = \begin{bmatrix} \frac{\partial \varepsilon_1}{\partial \vartheta_1} & \frac{\partial \varepsilon_1}{\partial \vartheta_2} & \cdots & \frac{\partial \varepsilon_1}{\partial \vartheta_{2(N_b+L)}} \\ \frac{\partial \varepsilon_2}{\partial \vartheta_1} & \frac{\partial \varepsilon_2}{\partial \vartheta_2} & \cdots & \frac{\partial \varepsilon_2}{\partial \vartheta_{2(N_b+L)}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial \varepsilon_{2K}}{\partial \vartheta_1} & \frac{\partial \varepsilon_{2K}}{\partial \vartheta_2} & \cdots & \frac{\partial \varepsilon_{2K}}{\partial \vartheta_{2(N_b+L)}} \end{bmatrix}. \quad (25)$$

The partial derivatives in the Jacobian (25) are calculated at the bottom of this page: for $1 \leq k \leq K$, they are given in (26) at the bottom of the page, while for $K + 1 \leq k \leq 2K$ and $t = k - K$, they are given in (27) at the bottom of the page. It can be seen that the De Boor algorithm (12) to (14) are utilized for evaluating (15) to (18), which are required for evaluating the

entries in (26) and (27). In addition, the following derivatives are also needed in (26) and (27)

$$\begin{cases} \frac{\partial \hat{r}(k)}{\partial \hat{h}_{R_i}} = \frac{1}{\hat{r}(k)} \left(\hat{w}_R(k) x_R(k-i) + \hat{w}_I(k) x_I(k-i) \right), \\ \frac{\partial \hat{r}(k)}{\partial \hat{h}_{I_i}} = \frac{1}{\hat{r}(k)} \left(\hat{w}_I(k) x_R(k-i) - \hat{w}_R(k) x_I(k-i) \right), \\ \frac{\partial \hat{\psi}(k)}{\partial \hat{h}_{R_i}} = \frac{1}{\hat{r}^2(k)} \left(\hat{w}_R(k) x_I(k-i) - \hat{w}_I(k) x_R(k-i) \right), \\ \frac{\partial \hat{\psi}(k)}{\partial \hat{h}_{I_i}} = \frac{1}{\hat{r}^2(k)} \left(\hat{w}_R(k) x_R(k-i) + \hat{w}_I(k) x_I(k-i) \right). \end{cases} \quad (28)$$

The iterative procedure (24) can be terminated when $\boldsymbol{\vartheta}^{(\tau)}$ converges or when a predetermined number of iterations has been reached. This Gauss-Newton algorithm has been shown to converge fast with very accurate results in our previous work [35]. Alternatively, the Levenberg-Marquardt algorithm [36]–[38] can be applied with the benefits of faster convergence at the cost of increased computational complexity.

As the cost function (22) is highly nonlinear, the solution of any gradient-based algorithm depends on the initial condition. It is important to properly initialize $\boldsymbol{\vartheta}^{(0)}$ so that it is as close as possible to an optimal solution. Furthermore, it is desirable that the parameter initialization is simple to implement. A simple and effective parameter initialization is presented in Appendix using the least squares algorithm. More detailed discussions on the issue of parameter initialization in nonlinear model identification can be found for example in [39]–[41].

$$\frac{\partial \varepsilon_k}{\partial \vartheta_q} = \begin{cases} \frac{\partial e_R(k)}{\partial \omega_l} = -B_l^{(P_o)}(\hat{r}(k)) \cos\left(\hat{\Phi}(\hat{r}(k)) + \hat{\psi}(k)\right), \ q = 1, 2, \cdots, N_b \text{ and } l = q, \\ \frac{\partial e_R(k)}{\partial \theta_l} = \hat{A}(\hat{r}(k)) \sin\left(\hat{\Phi}(\hat{r}(k)) + \hat{\psi}(k)\right) B_l^{(P_o)}(\hat{r}(k)), \ q = N_b + 1, \cdots, 2N_b \text{ and } l = q - N_b, \\ \frac{\partial e_R(k)}{\partial \hat{h}_{R_l}} = \left(\hat{A}(\hat{r}(k)) \sin\left(\hat{\Phi}(\hat{r}(k)) + \hat{\psi}(k)\right) \hat{\Phi}'(\hat{r}(k)) - \hat{A}'(\hat{r}(k)) \cos\left(\hat{\Phi}(\hat{r}(k)) + \hat{\psi}(k)\right)\right) \frac{\partial \hat{r}(k)}{\partial \hat{h}_{R_l}} \\ \quad + \hat{A}(\hat{r}(k)) \sin\left(\hat{\Phi}(\hat{r}(k)) + \hat{\psi}(k)\right) \frac{\partial \hat{\psi}(k)}{\partial \hat{h}_{R_l}}, \ q = 2N_b + 1, \cdots, 2N_b + L \text{ and } l = q - 2N_b, \\ \frac{\partial e_R(k)}{\partial \hat{h}_{I_l}} = \left(\hat{A}(\hat{r}(k)) \sin\left(\hat{\Phi}(\hat{r}(k)) + \hat{\psi}(k)\right) \hat{\Phi}'(\hat{r}(k)) - \hat{A}'(\hat{r}(k)) \cos\left(\hat{\Phi}(\hat{r}(k)) + \hat{\psi}(k)\right)\right) \frac{\partial \hat{r}(k)}{\partial \hat{h}_{I_l}} \\ \quad + \hat{A}(\hat{r}(k)) \sin\left(\hat{\Phi}(\hat{r}(k)) + \hat{\psi}(k)\right) \frac{\partial \hat{\psi}(k)}{\partial \hat{h}_{I_l}}, \ q = 2N_b + L + 1, \cdots, 2(N_b + L) \text{ and } l = q - 2N_b - L, \end{cases} \quad (26)$$

$$\frac{\partial \varepsilon_k}{\partial \vartheta_q} = \begin{cases} \frac{\partial e_I(t)}{\partial \omega_l} = -B_l^{(P_o)}(\hat{r}(t)) \sin\left(\hat{\Phi}(\hat{r}(t)) + \hat{\psi}(t)\right), \ q = 1, 2, \cdots, N_b \text{ and } l = q, \\ \frac{\partial e_I(t)}{\partial \theta_l} = -\hat{A}(\hat{r}(t)) \cos\left(\hat{\Phi}(\hat{r}(t)) + \hat{\psi}(t)\right) B_l^{(P_o)}(\hat{r}(t)), \ q = N_b + 1, \cdots, 2N_b \text{ and } l = q - N_b, \\ \frac{\partial e_I(t)}{\partial \hat{h}_{R_l}} = -\left(\hat{A}(\hat{r}(t)) \cos\left(\hat{\Phi}(\hat{r}(t)) + \hat{\psi}(t)\right) \hat{\Phi}'(\hat{r}(t)) + \hat{A}'(\hat{r}(t)) \sin\left(\hat{\Phi}(\hat{r}(t)) + \hat{\psi}(t)\right)\right) \frac{\partial \hat{r}(t)}{\partial \hat{h}_{R_l}} \\ \quad - \hat{A}(\hat{r}(t)) \cos\left(\hat{\Phi}(\hat{r}(t)) + \hat{\psi}(t)\right) \frac{\partial \hat{\psi}(t)}{\partial \hat{h}_{R_l}}, \ q = 2N_b + 1, \cdots, 2N_b + L \text{ and } l = q - 2N_b, \\ \frac{\partial e_I(t)}{\partial \hat{h}_{I_l}} = -\left(\hat{A}(\hat{r}(t)) \cos\left(\hat{\Phi}(\hat{r}(t)) + \hat{\psi}(t)\right) \hat{\Phi}'(\hat{r}(t)) + \hat{A}'(\hat{r}(t)) \sin\left(\hat{\Phi}(\hat{r}(t)) + \hat{\psi}(t)\right)\right) \frac{\partial \hat{r}(t)}{\partial \hat{h}_{I_l}} \\ \quad - \hat{A}(\hat{r}(t)) \cos\left(\hat{\Phi}(\hat{r}(t)) + \hat{\psi}(t)\right) \frac{\partial \hat{\psi}(t)}{\partial \hat{h}_{I_l}}, \ q = 2N_b + L + 1, \cdots, 2(N_b + L) \text{ and } l = q - 2N_b - L. \end{cases} \quad (27)$$
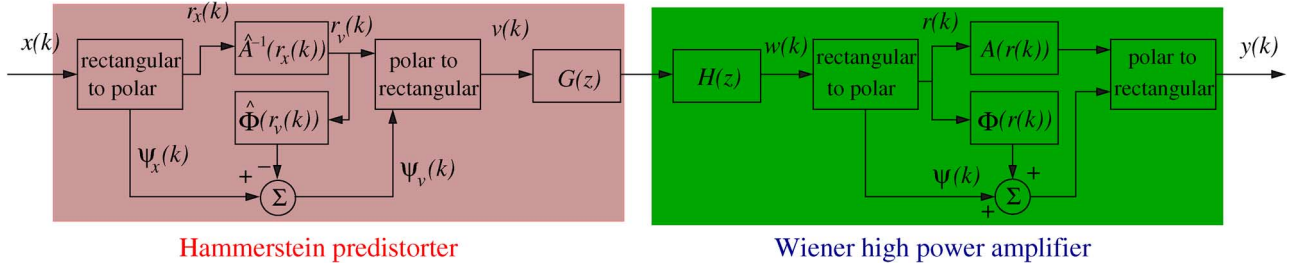
Fig. 2.   The predistorter design using the Hammerstein model.

III.   PREDISTORTER DESIGN WITH THE AID OF INVERSE DE BOOR ALGORITHM

An advantage of adopting the Wiener model for the memory HPA is that the exact inverse of the Wiener model can be represented by a static nonlinearity followed by a linear system, which is known as the Hammerstein model. In particular, our approach of modeling the static nonlinearity of the HPA by B-spline basis functions offers further significant advantages for the Hammerstein PD design. The proposed PD design is illustrated in Fig. 2.

A.   Inverse of the HPA's Static Nonlinearity

First, from the estimated amplitude response function (15) for the HPA's static nonlinearity, the estimates for r_{sat} and A_{max}, denoted by $\hat{r}_{\mathrm{sat}}$ and $\hat{A}_{\mathrm{max}}$, can easily be obtained using numerical search. Note that, for $0 \leq r \leq \hat{r}_{\mathrm{sat}}$, $\hat{A}(r)$ of (15) and $\hat{\Phi}(r)$ of (16) are one-to-one mappings, that is, they are continuous and invertible functions. Referring to Fig. 2, let the input to the static nonlinearity of the PD be $x(k) = r_x(k) \cdot \exp(\mathrm{j}\psi_x(k))$, where $r_x(k)$ and $\psi_x(k)$ are the amplitude and phase of $x(k)$, respectively. Similarly, denote the output of the PD's static nonlinearity by $v(k) = r_v(k) \cdot \exp(\mathrm{j}\psi_v(k))$, with $r_v(k)$ and $\psi_v(k)$ being the amplitude and phase of $v(k)$, respectively. The PD's static nonlinearity is the inverse of the HPA's static nonlinearity. The procedure for calculating the inverse of the HPA's nonlinearity is summarized in the following.

1) Calculate the inverse of the estimated amplitude distortion function for the given $r_x(k)$: if $0 \leq r_x(k) \leq \hat{A}_{\mathrm{max}}$, do 1.a); otherwise, if $r_x(k) > \hat{A}_{\mathrm{max}}$, do 1.b)
   1.a) Using the inverse of De Boor algorithm detailed below to compute $r_v(k) = \hat{A}^{-1}(r_x(k))$
   1.b) $r_v(k) = \hat{r}_{\mathrm{sat}}$
2) Calculate the estimated phase distortion as $\hat{\Phi}(r_v(k))$ using (16). The inverse of the estimated phase distortion is then $-\hat{\Phi}(r_v(k))$, yielding $\psi_v(k) = \psi_x(k) - \hat{\Phi}(r_v(k))$.

We now discuss how to find the inverse $r_v = \hat{A}^{-1}(r_x)$ in Step 1.a). Given that r_x lies in the range between the two points, $\hat{A}(R_{\mathrm{min}})$ and $\hat{A}(R_{\mathrm{max}})$, the problem is to find the root of the polynomial equation of $r_x = \sum_{l=1}^{N_b} B_l^{(P_o)}(r_v)\omega_l$. We propose to solve the problem using the inverse of De Boor algorithm, which effectively utilizes the B-spline curve and first-order derivative recursions in the Newton-Raphson formula that is modified to take into account that r_v is positive. Note that, in the range considered, $\hat{A}(r_v)$ is monotonic, and this means that the inverse of De Boor algorithm converges to the unique solution.

The Inverse of De Boor Algorithm:
1) Initialize $r_v^{(0)}$ as a random number with $R_{\mathrm{min}} < r_v^{(0)} < R_{\mathrm{max}}$.
2) The $(\tau + 1)$-th step is given by

$$\tilde{r}_v^{(\tau+1)} = r_v^{(\tau)} + \eta \Delta r_v^{(\tau)} = r_v^{(\tau)} + \eta \frac{\left(r_x - \hat{A}\left(r_v^{(\tau)}\right)\right)}{\hat{A}'\left(r_v^{(\tau)}\right)}, \quad (29)$$

$$r_v^{(\tau+1)} = \max\left\{\tilde{r}_v^{(\tau+1)}, 0\right\}, \quad (30)$$

where $0 < \eta \ll 1$ is the learning rate, which is preset empirically, while $\hat{A}(r_v^{(\tau)})$ and $\hat{A}'(r_v^{(\tau)})$ are calculated using (15) and (17), in which the De Boor recursions (12) to (14) are utilized.
3) The algorithm is terminated when $|\Delta r_v^{(\tau)}| < \varsigma$, where ς is a preset required precision, e.g. $\varsigma = 10^{-5}$, or τ reaches a predetermined maximum value. Otherwise, set $\tau = \tau + 1$ and repeat Step 2).

Computational cost of the inverse of De Boor algorithm is very low at the order of $\mathcal{O}(P_o^2)$, scaled by the number of iterations. Furthermore, for the M-QAM signal (2), the number of data symbols with distinct amplitudes is much smaller than M. For example, the 64-QAM symbol set only contains 9 distinct amplitude values, as illustrated in Fig. 3. The amplitude distortion values $r_v = \hat{A}^{-1}(r_x)$ and the phase distortion values $\hat{\Phi}(r_v)$ for these distinct amplitude values r_x can be pre-calculated off-line and stored for on-line transmission. Therefore, the on-line computational complexity of our proposed PD solution is extremely low.[1]

B.   Inverse of the HPA's Linear Filter

The identification algorithm presented in the previous section provides the estimate of the HPA's linear filter $\hat{H}(z) = 1 + \sum_{i=1}^{L} \hat{h}_i z^{-i}$. Let the transfer function of the Hammerstein PD's linear filter be

$$G(z) = z^{-\tau} \cdot \sum_{i=0}^{L_g} g_i z^{-i}, \quad (31)$$

---

[1]In this study, we assume the symbol-rate sampling. If the sampling rate is higher, the amplitude of the signal sample no longer takes the few distinct amplitude values of the M-QAM signaling. Similarly, for the OFDM signal, its legitimate set of amplitude values may also be very large. We may opt for the on-line calculation of the predistorter operation for each transmitted signal at an increased on-line computational cost. However, the identification and inverting algorithms presented at this study remain applicable, and our design remains very competitive, in terms of accuracy and efficiency, compared with many existing predistorter designs.
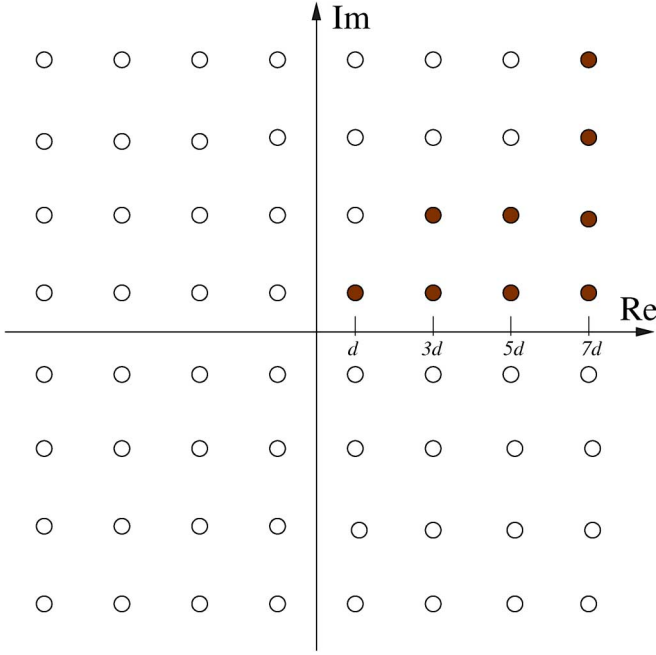
Fig. 3. 64-QAM constellation, which contains only 9 distinct amplitude values as illustrated by the filled symbol points.

where the delay $\tau = 0$ if $H(z)$ is minimum phase. The solution of the PD's linear filter $\mathbf{g} = [g_0 \ g_1 \ \cdots \ g_{L_g}]^{\mathrm{T}}$ can readily be obtained by solving the set of linear equations specified by

$$G(z) \cdot \hat{H}(z) = z^{-\tau}. \tag{32}$$

To guarantee an accurate inverse, the length of $\mathbf{g}$ should be chosen to be three to four times of the length of $\mathbf{h}$. Note that $g_0 = 1$ as $h_0 = 1$.

IV. SIMULATION STUDY

We considered the single-carrier 64-QAM system with the static nonlinearity of the memory HPA described by (6) and (7). The parameters of the memory HPA were given as

$$\mathbf{h}^{\mathrm{T}} = [0.7692 + \mathrm{j}0.0 \ 0.1538 + \mathrm{j}0.0 \ 0.0769 + \mathrm{j}0.0],$$
$$\mathbf{t}^{\mathrm{T}} = [2.1587 \ 1.15 \ 4.0 \ 2.1]. \tag{33}$$

The serious nonlinear and memory distortions caused by this memory HPA are illustrated in Fig. 4. Note that, for IBO = 0.5 dB, the HPA is operating well into the saturation region.

A. HPA Model Identification Results

The 64-QAM training sets each containing $K = 2000$ data samples were generated using (21) with the HPA operating at the IBO values of 5 dB and 0.5 dB, respectively, where the power of the complex-valued output measurement noise $\xi(k)$ was $2\sigma_\xi^2$. Note that since the identification is carried out at the transmitter, both the HPA's input $\mathbf{x}(k)$ and the corresponding HPA's output measurement $y(k)$ are available. Furthermore, the measurement $y(k)$ can usually be considered as noise free, i.e. $\sigma_\xi^2 = 0.0$. However, to demonstrate the effectiveness of the proposed B-spline neural network identification approach, we considered both the noise-free and noisy measurement cases with $\sigma_\xi^2 = 0.0$ and $\sigma_\xi^2 = 0.01$, respectively.
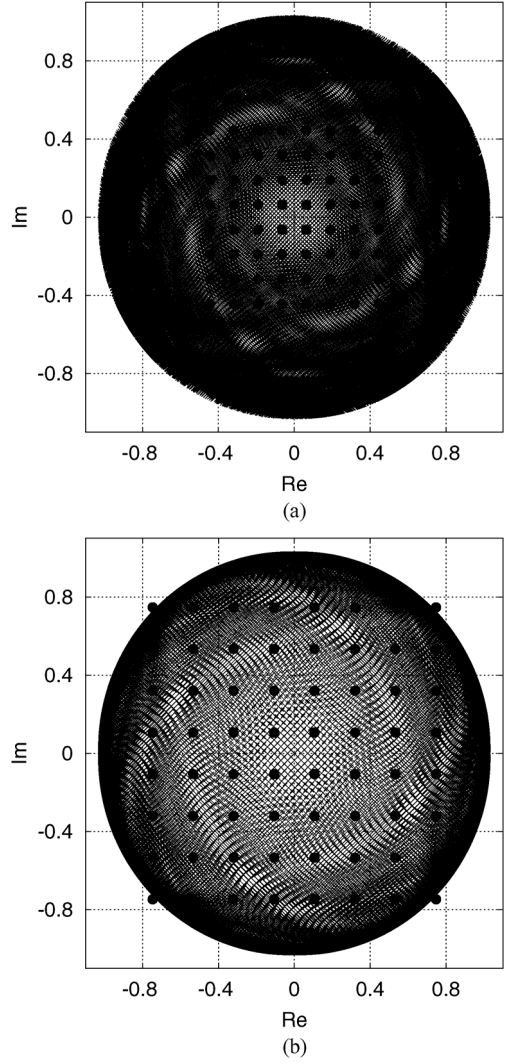

Fig. 4. The HPA's output $y(k)$, marked by $\times$, for the 64-QAM input signal $x(k)$, marked by $\bullet$: (a) the IBO of 5 dB, and (b) the IBO of 0.5 dB.

The piecewise cubic polynomial ($P_o = 4$) was chosen as the B-spline basis function, and the number of B-spline basis functions was set to $N_b = 6$. The predetermined knot sequence $\{-2 \times 10^{-5}, -10^{-5}, -10^{-6}, \mathbf{0.001}, 0.1, 0.3, 0.7, \mathbf{1.1}, 1.5, 5.0, 12.0\}$ was used. Note that the boundary knot values $R_{\min} = 0.001$ and $R_{\max} = 1.1$ were chosen such that $R_{\min} \approx 0$ and the input signal amplitude was less than $R_{\max}$. The number of B-spline basis functions N_b and the polynomial order P_o should be chosen to be sufficiently large to provide accurate approximation capability but not too large as to cause overfitting and to impose unnecessary complexity. From the literature, it is well known that $P_o = 3$ or 4 is often sufficient. It seems that the interval $[R_{\min}, \ R_{\max}] = [0.001, 1.1]$ can be partitioned well by the boundary and internal knot values $\{\mathbf{0.001}, 0.1, 0.3, 0.7, \mathbf{1.1}\}$. However, a uniform-spaced partition is also valid and equally effective. The extrapolation capability of the B-spline model is influenced by the choice of the external knots. For this particular example, we know that there exists no data for $r \gg R_{\max}$ in identification but we need the B-spline model having capability of extrapolating

TABLE I
IDENTIFICATION RESULTS FOR $\mathbf{h}$, r_{sat} AND $A_{\max}$

|		
true parameter values		
$\mathbf{h}^T = [0.7692 + \mathrm{j}0.0 \ 0.1538 + \mathrm{j}0.0 \ 0.0769 + \mathrm{j}0.0]$	$r_{\mathrm{sat}} = 0.9325$	$A_{\max} = 1.0065$
estimated values under IBO= 5 dB and $2\sigma_\xi^2 = 0.02$		
$\hat{\mathbf{h}}^T = [(7.7089e-1) + \mathrm{j}(-5.7923e-4) \ (1.5375e-1) + \mathrm{j}(2.9933e-3) \ (8.6376e-2) + \mathrm{j}(-1.7481e-3)]$	$\hat{r}_{\mathrm{sat}} = 0.94$	$\hat{A}_{\max} = 1.05$
estimated values under IBO= 5 dB and $2\sigma_\xi^2 = 0.0$		
$\hat{\mathbf{h}}^T = [(7.6920e-1) + \mathrm{j}(-1.5465e-6) \ (1.5380e-1) + \mathrm{j}(1.7025e-6) \ (7.6897e-2) + \mathrm{j}(-3.7445e-6)]$	$\hat{r}_{\mathrm{sat}} = 0.93$	$\hat{A}_{\max} = 1.01$
estimated values under IBO= 0.5 dB and $2\sigma_\xi^2 = 0.02$		
$\hat{\mathbf{h}}^T = [(7.7261e-1) + \mathrm{j}(-7.9909e-4) \ (1.5564e-1) + \mathrm{j}(4.5348e-4) \ (8.4417e-2) + \mathrm{j}(-2.2091e-3)]$	$\hat{r}_{\mathrm{sat}} = 1.00$	$\hat{A}_{\max} = 1.01$
estimated values under IBO=0.5 dB and $2\sigma_\xi^2 = 0.0$		
$\hat{\mathbf{h}}^T = [(7.6901e-1) + \mathrm{j}(-8.7790e-6) \ (1.5377e-1) + \mathrm{j}(7.3641e-5) \ (7.6866e-2) + \mathrm{j}(6.0605e-5)]$	$\hat{r}_{\mathrm{sat}} = 0.93$	$\hat{A}_{\max} = 1.01$
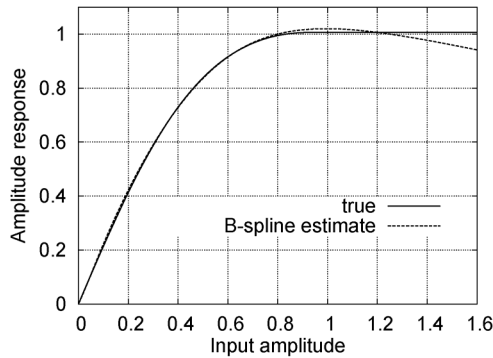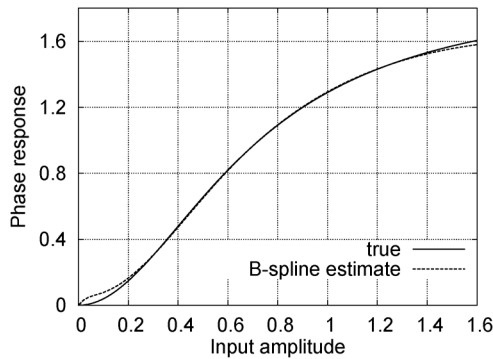
Fig. 5. Comparison of (a) the amplitude response and (b) the phase response between the HPA and the estimated B-spline model, where the HPA operates at the IBO of 5 dB with the measurement noise variance $2\sigma_\xi^2 = 0.02$.

Fig. 6. Comparison of (a) the amplitude response and (b) the phase response between the HPA and the estimated B-spline model, where the HPA operates at the IBO of 5 dB with the measurement noise variance $2\sigma_\xi^2 = 0.0$.

well into the saturation region of $r \gg R_{\max}$. Our experience suggests that by choosing the external knot values well spread into the region of $r \gg R_{\max}$, we can achieve excellent extrapolation capability. The identification algorithm as described in Section II-C with the parameter initialization as described in Appendix was carried out. The results obtained are summarized in Table I as well as illustrated in Figs. 5 to 8, which confirm that an accurate B-spline neural network model can be obtained for the memory HPA even in the cases that the measurements $y(k)$ are corrupted by noise.

Note that, under the identification condition of IBO = 5 dB, there were relative few data points which yielded $r_w(k)$ with the values near the saturation value r_{sat}. Consequently, the estimated B-spline amplitude response $\hat{A}(r)$ exhibits noticeable deviation from the HPA's true amplitude response $A(r)$ in the region $r \gg R_{\max}$, as can be seen from Figs. 5 and 6. This of
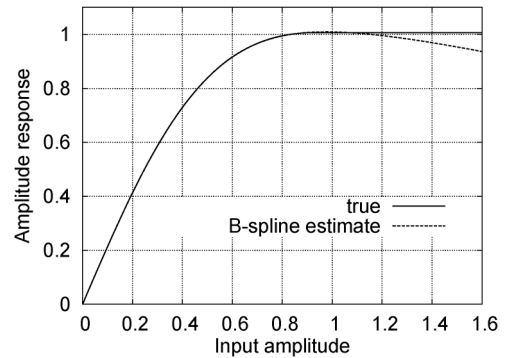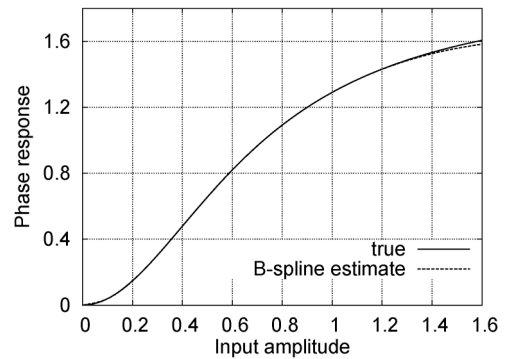
course does not matter, as this region is well beyond the operating region of the HPA. Interestingly, under the operating condition of IBO = 0.5 dB, the deviation between the estimated amplitude response $\hat{A}(r)$ and the true amplitude response $A(r)$ at the region of $r \gg R_{\max}$ is no longer noticeable, but small deviations are observed between the estimated phase response $\hat{\Phi}(r)$ and the true phase response $\Phi(r)$ for the region of r close to 0, as can be seen from Figs. 7 and 8. This is because, under the operating condition of IBO = 0.5 dB, there was sufficient number of input points with $r_w(k)$ close to the value of r_{sat}, but there were very few input points with very small $r_w(k)$.

B. Proposed Predistorter Performance

We employed the estimated B-spline Wiener HPA model obtained under the condition of noise-free measurement ($\sigma_\xi^2 = 0.0$) to design the proposed Hammerstein PD as detailed in
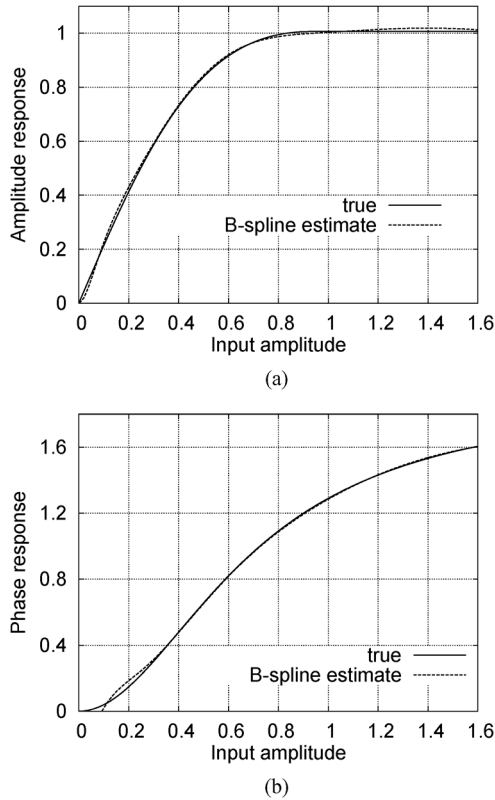
Fig. 7. Comparison of (a) the amplitude response and (b) the phase response between the HPA and the estimated B-spline model, where the HPA operates at the IBO of 0.5 dB with the measurement noise variance $2\sigma_\xi^2 = 0.02$.
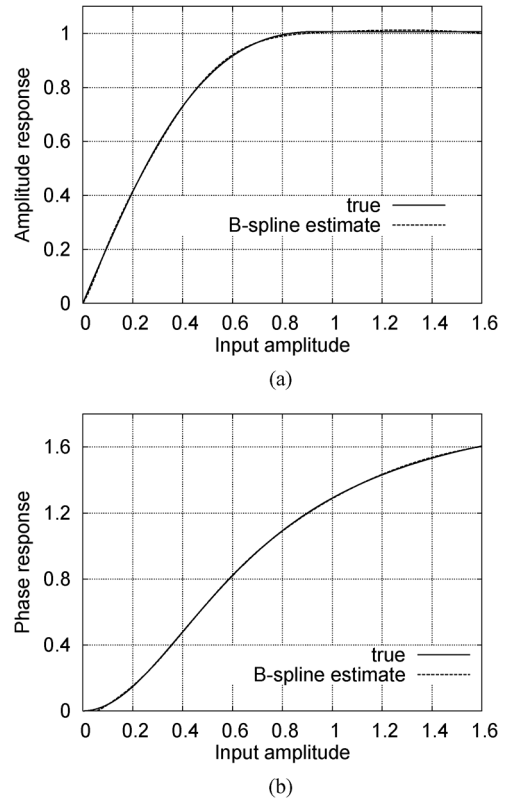

Fig. 8. Comparison of (a) the amplitude response and (b) the phase response between the HPA and the estimated B-spline model, where the HPA operates at the IBO of 0.5 dB with the measurement noise variance $2\sigma_\xi^2 = 0.0$.

Section III. Note that we only needed to calculate the 9 amplitude and phase predistortion values for the 9 distinct amplitude values of the 64-QAM constellation using the inverse of De Boor algorithm as described in Section III-A. The length of the PD's inverse filter was set to $L_g = 12$. The outputs of the combined PD and HPA are depicted in Fig. 9. Compared with the outputs of the HPA as plotted in Fig. 4, it can be seen that the designed PD successfully removes the serious distortions caused by the memory HPA. The achievable performance of the designed PD was further assessed using the mean square error (MSE) metric defined by

$$\text{MSE} = 10 \log_{10} \left(\frac{1}{K_\text{test}} \sum_{k=1}^{K_\text{test}} |x(k) - y(k)|^2 \right), \qquad (34)$$

as well as the system's BER, where K_test was the number of test data, $x(k)$ was the 64-QAM input and $y(k)$ was the output of the combined PD and memory HPA system. The channel signal to noise ratio (SNR) in the simulation was given by

$$\text{SNR} = 10 \log_{10} \left(\frac{\text{E}_\text{b}}{\text{N}_\text{o}} \right), \qquad (35)$$

where E_b was defined as the energy per bit and N_o the power of the channel's additive white Gaussian noise (AWGN).

With $K_\text{test} = 10^5$, 64-QAM data were passed through the combined Hammerstein PD and Wiener HPA system to compute the MSE (34), and the resulting MSE as the function of IBO is plotted in Fig. 10. The output signal after the memory HPA was then transmitted over the AWGN channel, and the

BER was then determined at the receiver. The results so obtained are plotted in Fig. 11, in comparison with the benchmark BER curve of the ideal AWGN channel. It can be seen from Fig. 11 that the BER performance of the combined PD and HPA system is practically indistinguishable from those of the ideal AWGN channel under the operating condition of IBO = 5 dB, which again demonstrates the effectiveness of the proposed PD design. The achievable BER performance of the combined PD and Wiener HPA system are further illustrated in Fig. 12 for the three values of the channel SNR.

V. CONCLUSIONS

An novel digital predistorter design has been proposed to compensate distortions caused by memory high power amplifiers based on the direct learning framework using a data-driven B-spline Wiener system modeling approach. The B-spline neural network model, which naturally incorporates the efficient De Boor algorithm with both B-spline curve and first derivative recursions, has been utilized to identify an accurate memory HPA model, based on which the proposed Hammerstein predistorter can be directly obtained. It has been shown that the inverse of the amplitude distortion function can be calculated efficiently and accurately with the aid of the inverse of De Boor algorithm and, moreover, for the QAM signal, the computation of the nonlinear predistortion is only required for a very small number of the data symbols with distinct amplitude values. The effectiveness of the proposed PD design has been illustrated by simulation results. In particular, it
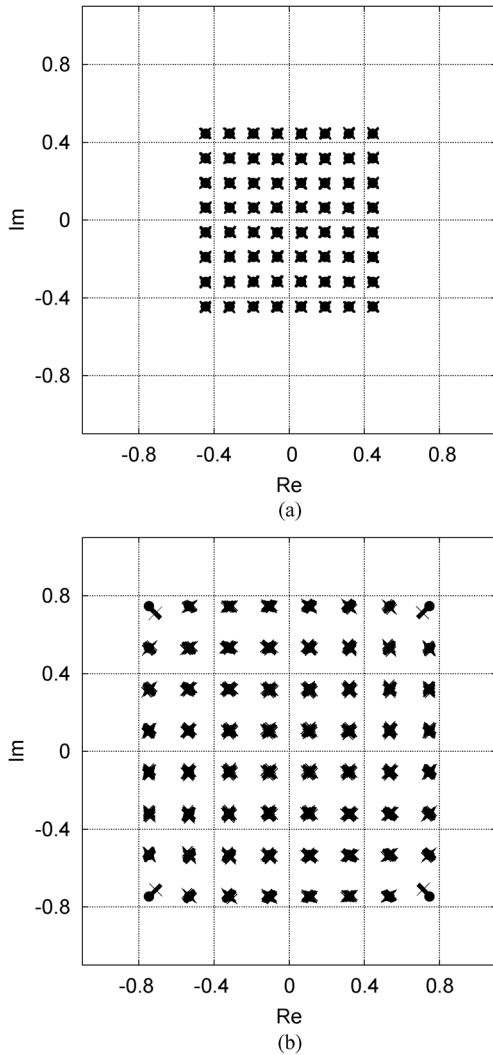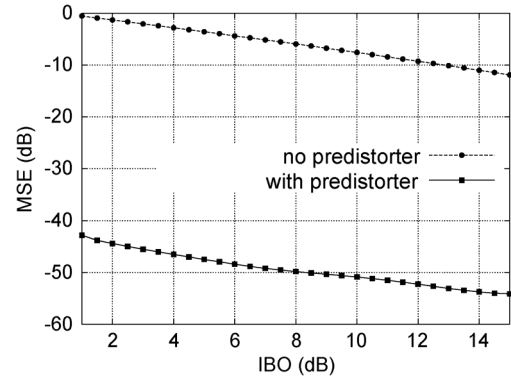
Fig. 10.   The mean square error versus IBO performance.

Fig. 11.   The bit error rate versus channel SNR performance.

Fig. 9.   The output of the combined PD and HPA $y(k)$, marked by $\times$, for the 64-QAM input signal $x(k)$, marked by $\bullet$: (a) the IBO of 5 dB, and (b) the IBO of 0.5 dB.

has been demonstrated that both the Wiener HPA identification and the Hammerstein predistorter inverse can be achieved very efficiently and accurately, and the proposed novel digital PD is capable of successfully compensating serious nonlinear distortions caused by the memory HPA operating into the output saturation region.

APPENDIX
A SIMPLE LEAST SQUARES PARAMETER INITIALIZATION

The initial estimate $\boldsymbol{\vartheta}^{(0)}$ can be generated based on the training data $\{x(k), y(k)\}_{k=1}^{K}$ as follows.

1) Set $\hat{h}_{R_i}^{(0)} = 0$ and $\hat{h}_{I_i}^{(0)} = 0$ for $1 \leq i \leq L$.
2) Generate the sequence $r^{(0)}(k) = \sqrt{x_R^2(k) + x_I^2(k)}$ for $1 \leq k \leq K$.
3) Generate the sequence $r_d^{(0)}(k) = \sqrt{y_R^2(k) + y_I^2(k)}$ for $1 \leq k \leq K$, and denote $\mathbf{r}_d = [r_d^{(0)}(1) \ r_d^{(0)}(2) \ \cdots \ r_d^{(0)}(K)]^{\mathrm{T}}$.
4) Generate the sequence $\phi^{(0)}(k) = \mathrm{atan2}(y_I(k)/y_R(k)) - \mathrm{atan2}(x_I(k)/x_R(k))$, whose value should be taken in the
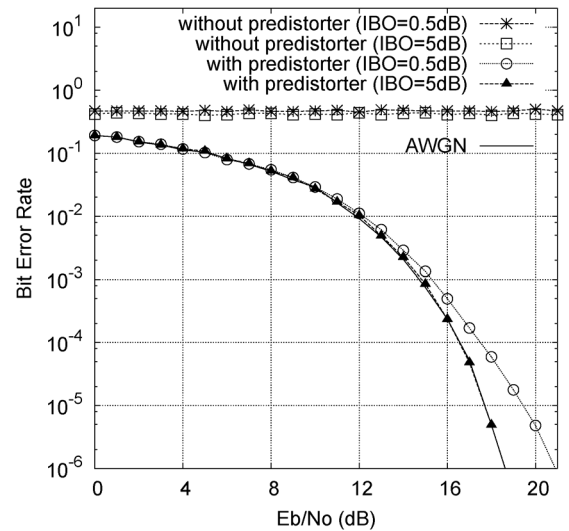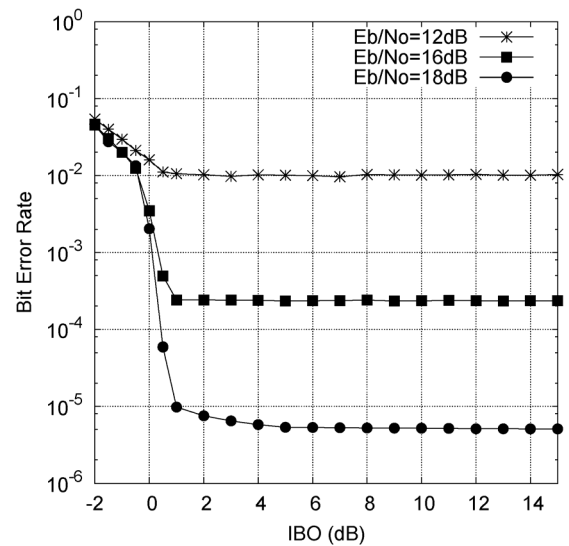

Fig. 12.   The bit error rate versus IBO performance of the combined PD and HPA for three values of the channel SNR.

range $[-\pi, \pi]$, for $1 \leq k \leq K$. Then denote $\boldsymbol{\phi}^{(0)} = [\phi^{(0)}(1) \ \phi^{(0)}(2) \ \cdots \ \phi^{(0)}(K)]^{\mathrm{T}}$.

5) Form the regression matrix

$$
\mathbf{B} = \begin{bmatrix}
B_1^{(P_o)}\left(r^{(0)}(1)\right) & \cdots & B_{N_b}^{(P_o)}\left(r^{(0)}(1)\right) \\
B_1^{(P_o)}\left(r^{(0)}(2)\right) & \cdots & B_{N_b}^{(P_o)}\left(r^{(0)}(2)\right) \\
\vdots & \ddots & \vdots \\
B_1^{(P_o)}\left(r^{(0)}(K)\right) & \cdots & B_{N_b}^{(P_o)}\left(r^{(0)}(K)\right)
\end{bmatrix}.
$$

6) Compute the least squares estimates $\boldsymbol{\omega}^{(0)} = \left(\mathbf{B}^{\mathrm{T}}\mathbf{B}\right)^{-1}\mathbf{B}^{\mathrm{T}}\mathbf{r}_d$ and $\boldsymbol{\theta}^{(0)} = (\mathbf{B}^{\mathrm{T}}\mathbf{B})^{-1}\mathbf{B}^{\mathrm{T}}\boldsymbol{\phi}^{(0)}$.

7) Set

$$
\boldsymbol{\vartheta}^{(0)} = \left[\left(\boldsymbol{\omega}^{(0)}\right)^{\mathrm{T}} \ \left(\boldsymbol{\theta}^{(0)}\right)^{\mathrm{T}} \ \underbrace{0 \ 0 \ \cdots \ 0}_{2L}\right]^{\mathrm{T}}.
$$

REFERENCES

[1] P. M. Grant, S. McLaughlin, H. Aghvami, and S. Fletcher, "Green radio—Towards sustainable wireless networks," in *Mobile VCE Core 5 Programme Presentation*, Apr. 2009 [Online]. Available: http://www.ee.princeton.edu/seminars/iss/Spring2009/slides/grant.pdf

[2] 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, U.K.: Wiley, 2004.

[3] L. Hanzo, M. Münster, B. J. Choi, and T. Keller, *OFDM and MC-CDMA for Broadband Multi-User Communications, WLANs and Broadcasting*. Chichester, U.K.: Wiley, 2003.

[4] J. H. K. Vuolevi, T. Rahkonen, and J. P. A. Manninen, "Measurement technique for characterizing memory effects in RF power amplifiers," *IEEE Trans. Microw. Theory Tech.*, vol. 49, no. 8, pp. 1383–1389, Aug. 2001.

[5] C.-H. Lin, H.-H. Chen, Y.-Y. Wang, and J.-T. Chen, "Dynamically optimum lookup-table spacing for power amplifier predistortion linearization," *IEEE Trans. Microw. Theory Tech.*, vol. 54, no. 5, pp. 2118–2127, May 2006.

[6] B. Ai, Z.-Y. Yang, C.-P. Pan, S.-G. Tang, and T. T. Zhang, "Analysis on LUT based predistortion method for HPA with memory," *IEEE Trans. Broadcast.*, vol. 53, no. 1, pp. 127–131, Mar. 2007.

[7] P. Jardin and G. Baudoin, "Filter lookup table method for power amplifier linearization," *IEEE Trans. Veh. Technol.*, vol. 56, no. 3, pp. 1076–1087, May 2007.

[8] H.-H. Chen, C.-H. Lin, P.-C. Huang, and J.-T. Chen, "Joint polynomial and look-up-table predistortion power amplifier linearization," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 53, no. 8, pp. 612–616, Aug. 2006.

[9] R. Raich, H. Qian, and G. T. Zhou, "Orthogonal polynomials for power amplifier modeling and predistorter design," *IEEE Trans. Veh. Technol.*, vol. 53, no. 5, pp. 1468–1479, Sep. 2004.

[10] D. R. Morgan, Z.-X. 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.

[11] L. Ding, G. T. Zhou, D. R. Morgan, Z. Ma, J. S. Kenney, J. Kim, and C. R. Giardina, "A robust digital baseband predistorter constructed using memory polynomials," *IEEE Trans. Commun.*, vol. 52, no. 1, pp. 159–165, Jan. 2004.

[12] 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. Technol.*, vol. 57, no. 2, pp. 882–892, Mar. 2008.

[13] L. Xu, X. Wu, M. Zhang, G. Kang, and P. Zhang, "A stable recursive algorithm for memory polynomial predistorter," in *Proc. MILCOM*, Orlando, FL, Oct. 29–31, 2007, pp. 1–5.

[14] Y. Qian and T. Yao, "Structure for adaptive predistortion suitable for efficient adaptive algorithm application," *Electron. Lett.*, vol. 38, no. 21, pp. 1282–1283, Oct. 2002.

[15] 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.

[16] 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.

[17] 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. Broadcast.*, vol. 57, no. 1, pp. 149–158, Mar. 2011.

[18] C. J. Clark, G. Chrisikos, M. S. Muha, A. A. Moulthrop, and C. P. Silva, "Time-domain envelope measurement technique with application to wideband power amplifier modeling," *IEEE Trans. Microw. Theory Tech.*, vol. 46, no. 12, pp. 2531–2540, Dec. 1998.

[19] A. A. M. Saleh, "Frequency-independent and frequency-dependent nonlinear models of TWT amplifiers," *IEEE Trans. Commun.*, vol. COM-29, no. 11, pp. 1715–1720, Nov. 1981.

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

[21] C. De Boor, *A Practical Guide to Splines*. New York: Springer-Verlag, 1978.

[22] I. W. Hunter and M. J. Korenberg, "The identification of nonlinear biological system: Wiener and Hammerstein cascade models," *Biol. Cybern.*, vol. 55, no. 2–3, pp. 135–144, 1986.

[23] W. Greblicki, "Nonparametric identification of Wiener systems," *IEEE Trans. Inf. Theory*, vol. 38, no. 5, pp. 1487–1493, Sep. 1992.

[24] A. Kalafatis, N. Arifin, L. Wang, and W. R. Cluett, "A new approach to the identification of pH processes based on the Wiener model," *Chem. Eng. Sci.*, vol. 50, no. 23, pp. 3693–3701, Dec. 1995.

[25] D. Westwick and M. Verhaegen, "Identifying MIMO Wiener systems using subspace model identification methods," *Signal Process.*, vol. 52, no. 2, pp. 235–258, Jul. 1996.

[26] A. D. Kalafatis, L. Wang, and W. R. Cluett, "Identification of Wiener-type nonlinear systems in a noisy environment," *Int. J. Control*, vol. 66, no. 6, pp. 923–941, 1997.

[27] E. W. Bai, "An optimal two-stage identification algorithm for Hammerstein-Wiener nonlinear systems," *Automatica*, vol. 34, no. 3, pp. 333–338, Mar. 1998.

[28] Y. Zhu, "Distillation column identification for control using Wiener model," in *Proc. 1999 Amer. Control Conf.*, San Diego, CA, Jun. 2–4, 1999, pp. 3462–3466.

[29] J. C. Gómez, A. Jutan, and E. Baeyens, "Wiener model identification and predictive control of a pH neutralisation process," *IEE Proc. Control Theory Appl.*, vol. 151, no. 3, pp. 329–338, May 2004.

[30] I. Skrjanc, S. Blazic, and O. E. Agamennoni, "Interval fuzzy modeling applied to Wiener models with uncertainties," *IEEE Trans. Systems, Man, Cybern. B*, vol. 35, no. 5, pp. 1092–1095, Oct. 2005.

[31] A. Hagenblad, L. Ljung, and A. Wills, "Maximum likelihood identification of Wiener models," *Automatic*, vol. 44, no. 11, pp. 2697–2705, Nov. 2008.

[32] T. Kavli, "ASMOD—An algorithm for adaptive spline modelling of observation data," *Int. J. Control*, vol. 58, no. 4, pp. 947–967, 1993.

[33] M. Brown and C. J. Harris, *Neurofuzzy Adaptive Modelling and Control*. Hemel Hempstead, U.K.: Prentice-Hall, 1994.

[34] C. J. Harris, X. Hong, and Q. Gan, *Adaptive Modelling, Estimation and Fusion from Data: A Neurofuzzy Approach*. Berlin, Germany: Springer-Verlag, 2002.

[35] X. Hong and S. Chen, "Modeling of complex-valued Wiener systems using B-spline neural network," *IEEE Trans. Neural Netw.*, vol. 22, no. 5, pp. 818–825, May 2011.

[36] K. Levenberg, "A method for the solution of certain non-linear problems in least squares," *Quart. J. Appl. Math.*, vol. 2, no. 2, pp. 164–168, Jul. 1944.

[37] D. W. Marquardt, "An algorithm for least-squares estimation of nonlinear parameters," *SIAM J. Appl. Math.*, vol. 11, no. 2, pp. 431–441, Jun. 1963.

[38] A. E. Ruano, C. Cabrita, J. V. Oliveira, and L. T. Kóczy, "Supervised training algorithms for B-spline neural networks and neuro-fuzzy systems," *Int. J. Syst. Sci.*, vol. 33, no. 8, pp. 689–711, 2002.

[39] J. H. Friedman, "Multivariate adaptive regression splines," *Ann. Stat.*, vol. 19, no. 1, pp. 1–141, 1991.

[40] O. Nelles, "Nonlinear systems identification with local linear neuro-fuzzy models," Ph.D. Thesis, TU Darmstadt, Germany, 2000.

[41] J. Botzheim, C. Cabrita, L. T. Kóczy, and A. E. Ruano, "Genetic and bacterial programming for B-spline neural networks design," *J. Adv. Comput. Intell. Intell. Informatics*, vol. 11, no. 2, pp. 220–231, Feb. 2007.

Sheng Chen (M'90–SM'97–F'08) received his B.Eng. degree from the East China Petroleum Institute, Dongying, in 1982, and his Ph.D. degree from the City University, London, U.K., in 1986, both in control engineering. In 2005, he was awarded the D.Sc. degree from the University of Southampton, Southampton, U.K.

From 1986 to 1999, he held research and academic appointments at the Universities of Sheffield, Edinburgh, and Portsmouth, all in the U.K. Since 1999, he has been with Electronics and Computer Science, the University of Southampton, U.K., where he currently holds the post of Professor in Intelligent Systems and Signal Processing. His research interests include adaptive signal processing, wireless communications, modeling and identification of nonlinear systems, neural network and machine learning, intelligent control system design, evolutionary computation methods, and optimization. He has published over 450 research papers.

Dr. Chen is a Fellow of the Institution of Engineering and Technology. He is a Distinguished Adjunct Professor at King Abdulaziz University, Jeddah, Saudi Arabia. In the database of the world's most highly cited researchers in various disciplines, compiled by Institute for Scientific Information (ISI) of the United States, Dr Chen is on the list of the highly cited researchers in the engineering category.

Xia Hong (SM'02) received her B.Sc. and M.Sc. degrees from the National University of Defence Technology, Hunan, China, in 1984 and 1987, respectively, and her Ph.D. degree from the University of Sheffield, Sheffield, U.K., in 1998, all in automatic control.

She worked as a Research Assistant in Beijing Institute of Systems Engineering, Beijing, China from 1987 to 1993. She worked as a research fellow in the School of Electronics and Computer Science at the University of Southampton, U.K., from 1997 to 2001. Since 2001, she has been with the School of Systems Engineering, University of Reading, U.K., where she currently holds a Readership post. She is actively engaged in research into nonlinear systems identification, data modeling, estimation and intelligent control, neural networks, pattern recognition, learning theory, and their applications. She has published over 200 research papers, and coauthored a research book.

Dr. Hong was awarded a Donald Julius Groen Prize by the Institution of Mechanical Engineers in 1999.
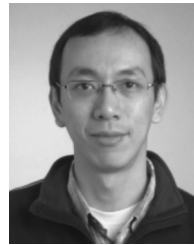
Yu Gong received his B.Eng. and M.Eng. degrees from the University of Electronics and Science Technology of China in 1992 and 1995, respectively, both in electronic engineering. He received his Ph.D. degree in communications from the National University of Singapore in 2002.

After his Ph.D., he took several research positions in the Institute of Infocomm Research in Singapore and Queen's University of Belfast in U.K., respectively. From 2006 to 2012, he was an Academic Staff Member in the School of Systems Engineering, University of Reading, U.K. Since July 2012, he has been with the School of Electronic, Electrical and Systems Engineering, Loughborough University, U.K. His research interests are in the area of signal processing and communications including wireless communications, cooperative networks, non-linear and non-stationary system identification, and adaptive filters.

Chris J. Harris received his B.Sc. and M.A. degrees from the University of Leicester and the University of Oxford in the U.K., respectively, and his Ph.D. degree from the University of Southampton, U.K., in 1972. He was awarded the D.Sc. degree by the University of Southampton in 2001.

He previously held appointments at the University of Hull, the UMIST, the University of Oxford, and the University of Cranfield, all in the U.K., as well as being employed by the U.K. Ministry of Defence. He returned to the University of Southampton as the Lucas Professor of Aerospace Systems Engineering in 1987. Since then, he has been with Electronics and Computer Science, the University of Southampton. His research interests lie in the general area of intelligent and adaptive systems theory and its application to intelligent autonomous systems such as autonomous vehicles, management infrastructures such as command and control, intelligent control, and estimation of dynamic processes, multi-sensor data fusion, and systems integration. He has authored and coauthored 12 research books and over 400 research papers, and he was the associate editor of numerous international journals.

Dr. Harris was elected to the Royal Academy of Engineering in 1996, was awarded the IEE Senior Achievement medal in 1998 for his work in autonomous systems, and the IEE Faraday medal in 2001 for his work in intelligent control and neurofuzzy systems.